**LONG COVID PREDICTION USING ENSEMBLE AND AUTOMATED MACHINE LEARNING**

**YONG XIAO JING**

**FACULTY OF COMPUTING AND INFORMATICS**

**UNIVERSITI MALAYSIA SABAH**

**2023**

**LONG COVID PREDICTION USING ENSEMBLE AND AUTOMATED MACHINE LEARNING**

**YONG XIAO JING**

**THESIS SUBMITTED IN PARTIAL FULFILLMENT FOR THE DEGREE OF BACHELOR OF COMPUTER SCIENCE WITH HONOURS**

**(DATA SCIENCE)**

**FACULTY OF COMPUTING AND INFORMATICS UNIVERSITI MALAYSIA SABAH**

**NAME** : YONG XIAO JING

**MATRIC NUMBER** : BI19110271

**TITLE** : LONG COVID PREDICTION USING AUTOMATED MACHINE LEARNING

**DEGREE** : BACHELOR OF COMPUTER WITH HONOURS (DATA SCIENCE)

**VIVA’S DATE** : 30 Jan 2023

**CERTIFIED BY,**

1. **SUPERVISOR** Signature,

PROF. DR. JASON TEO TZE WI

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**DECLARATION**

I hereby declare that the material in this thesis is my own except for quotations, equations, summaries and references, which have been duly acknowledged.

27 JAN 2023 **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

YONG XIAO JING

BI19110271

**ACKNOWLEDGEMENT**

This project was carried out with help and support from various people. First and foremost, I would like to express appreciation to my supervisor, Prof. Dr. Jason Teo Tze Wi, for his helpful advice and suggestions on how to better my final year project. His informative feedback encouraged me to improve my thoughts and raise the quality of my work. Second, I would like to express my gratitude to my parents for their unwavering support. Last but not least, I would like to thank my peers and friends for discussing and exchanging ideas with me to broaden my perspective.

YONG XIAO JING

30 JAN 2023

**ABSTRACT**

Patients have had secondary symptoms, including physical, intensive, and psychological problems, during or after the COVID-19 recovery phase, and these symptoms have lasted for at least four weeks. Long COVID is the name given to this condition but not everyone will suffer in this situation. Current research is mostly focused on predicting the COVID-19 trend, the severity of COVID-19 symptoms, and identifying COVID-19 clusters. Therefore, using Machine Learning to predict the presence of long COVID on a patient is currently understudied. According to statistical results from clinical reports, some variables, such as the patient's medical history and vaccination status, have associations with the presence of long COVID. The NCIPR dataset and the Kenya and Malawi datasets were used, and both provide variables that aid in detecting and forecasting whether a COVID-19-positive patient will have a long COVID. This project intends to utilise Ensemble Machine Learning (EML) and automated machine learning (AutoML) to forecast the occurrence of long COVID due to a lack of research. Auto-sklearn, which generates both single and ensemble machine learning models, and TPOT, which uses a different optimisation strategy from Auto-sklearn to produce machine learning models are used. The AutoML single classifier achieved the best performance with a precision score of 85.56%, recall score of 85.78% and accuracy score of 85.78%. For utilizing the Kenya and Malawi datasets as new test set to investigate the model generalization, the AutoML single classifier also achieved a better performance with a precision score of 99.26%, recall score of 66.35% and accuracy score of 66.35%. Additionally, tests with parameter manipulation and the use of an external oversampling technique and synthetic data generation have been conducted to increase the performance of models. The findings reveal that the synthetic data generation successfully improved the metric scores between 95% and 99% for both the AutoML ensemble model and single classifier, keeping the scores reasonably close to one another.

***ABSTRAK***

***Peramalan ‘Long COVID’ dengan Pembelajaran Mesin Automatik***

Pesakit telah mengalami gejala sekunder, termasuk masalah fizikal, intensif dan psikologi, semasa atau selepas fasa pemulihan COVID-19, dan gejala ini telah berlarutan selama sekurang-kurangnya empat minggu. Long COVID adalah nama yang diberikan kepada keadaan ini tetapi bukan semua orang akan menderita dalam keadaan ini. Penyelidikan semasa kebanyakannya tertumpu pada meramalkan arah aliran COVID-19, tahap keterukan gejala COVID-19 dan mengenal pasti kelompok COVID-19. Oleh itu, menggunakan Pembelajaran Mesin untuk meramalkan kehadiran COVID yang lama pada pesakit pada masa ini kurang dikaji. Menurut keputusan statistik daripada laporan klinikal, beberapa pembolehubah, seperti sejarah perubatan pesakit dan status vaksinasi, mempunyai kaitan dengan kehadiran COVID yang lama. Set data NCIPR dan set data Kenya&Malawi telah digunakan, dan kedua-duanya menyediakan pembolehubah yang membantu dalam mengesan dan meramalkan sama ada pesakit yang positif COVID-19 akan menghidap COVID yang lama. Projek ini bertujuan untuk menggunakan Pembelajaran Mesin Ensemble (EML) dan Pembelajaran Mesin Automatik (AutoML) untuk meramalkan kejadian COVID yang berpanjangan kerana kekurangan penyelidikan. Auto-sklearn, yang menjana kedua-dua model pembelajaran mesin tunggal dan ensemble, dan TPOT, yang menggunakan strategi pengoptimuman berbeza daripada Auto-sklearn untuk menghasilkan model pembelajaran mesin diimplikasikan. Model AutoML daripada TPOT mencapai prestasi terbaik dengan skor ‘precision’ 85.56%, skor ‘recall’ 85.78% dan skor ‘accuracy’ 85.78%. Bagi eksperimen menggunakan set data Kenya dan Malawi sebagai set ujian baharu untuk menyiasat generalisasi model, Model AutoML daripada TPOT juga mencapai prestasi terbaik dengan skor ‘precision’ 99.26%, skor ‘recall’ 66.35% dan skor ‘accuracy’ 66.35%. Selain itu, eksperimen dengan manipulasi parameter dan penggunaan teknik pensampelan dan penjanaan data sintetik telah dijalankan untuk meningkatkan prestasi model. Penemuan mendedahkan bahawa penjanaan data sintetik berjaya meratakan skor metrik antara 95% dan 99% untuk kedua-dua model ensembel AutoML dan model tunggal, memastikan skor hampir sama antara satu sama lain.

**TABLE OF CONTENTS**

|  |  |
| --- | --- |
| **TITLE** | **Page** |
| [**DECLARATION**](#declaration)**-----------------------------------------------------------------------** | ii |
| [**ABSTRACT**](#abstract)**----------------------------------------------------------------------------** | iv |
| **TABLE OF CONTENTS----------------------------------------------------------------** | vi |
| [**LIST OF TABLES**](#lstable)**----------------------------------------------------------------------** | x |
| [**LIST OF FIGURES**](#lsfig)**--------------------------------------------------------------------** | xii |
| [**CHAPTER 1 INTRODUCTION**](#intro) |  |
| * 1. [Introduction](#smallintro) | 1 |
| * 1. [Problem Background](#prbback) | 1-4 |
| * 1. [Problem Statement](#prbstate) | 4 |
| * 1. [Project Objectives](#prjobj) | 4 |
| * 1. [Project Scope](#prjscope) | 4-5 |
| * 1. [Organization of the Report](#orgreport) | 5 |
| [**CHAPTER 2 LITERATURE REVIEW**](#lr) |  |
| * 1. [Introduction](#lrintro) | 6-7 |
| * 1. [Long COVID Analysis](#lcanalysis) | 7-12 |
| * 1. [Ensemble Machine Learning in Health Care](#lreml) | 12-16 |
| * 1. [Automated Machine Learning in Health Care](#lrautoml)   2. [Critical Summary](#crisum)   3. [Conclusion](#lrconclu) | 16-27  27-28  29 |
| [**CHAPTER 3 METHODOLOGY**](#method) |  |
| * 1. [Introduction](#methodintro)   2. [Implementation Environment](#impleenvi)   3. [Data Collection](#datacollect)   4. [Data Pre-processing](#dataprocess)   5. [Feature Engineering](#feaengi)   6. [Model Building](#modelbuild)   7. [Data Splitting](#datasplit)   8. [Model Evaluation](#modeleval)   9. [Overall Flow of Activities](#chartact)   10. [Conclusion](#methodconclu) | 30  30-31  31  31-32  32-37  37-40  41-43  43-44  45  46 |
| [**CHAPTER 4 EXPERIMENTAL DESIGN**](#experimentaldesign) |  |
| * 1. [Introduction](#designintro)   2. [Data Curation](#datacuration)   3. [Baseline Model](#baselinemodel)   4. [Experiment With Manipulated Parameter](#expparam)   5. [Experiment With Feature Reduction](#expfeatr)   6. [Experiment with New Imported Test Set](#expnewtest)   7. [Experiment with New Imported Test Set and Manipulated Parameter](#expnewtestandparam)   8. [Experiment with Oversampling Technique](#expover)   9. [Experiment with Synthetic Data](#expsdg)   10. [Conclusion](#designconclu) | 47  47  48-49  50-51  51-53  53-55  55-57  56-57  57-58  58 |
| [**CHAPTER 5 IMPLEMENTATION**](#imp) |  |
| * 1. [Introduction](#impintro)   2. [Experiment 1: NCIPR Dataset + AutoML Single Classifier](#exp1)   3. [Experiment 2: NCIPR Dataset + Automated Ensemble ML](#exp2)   4. [Experiment 3: NCIPR Dataset + AutoML with Different Generations](#exp3)   5. [Experiment 4: NCIPR Dataset + Automated Ensemble ML with Different Ensemble Size](#exp4)   6. [Experiment 5: NCIPR Dataset + Feature Reduction + AutoML Single Classifier](#sec56)   7. [Experiment 6: NCIPR Dataset + Feature Reduction + Automated Ensemble ML](#sec57)   8. [Experiment 7: NCIPR Dataset + Kenya and Malawi Datasets + AutoML Single Classifier](#sec58)   9. [Experiment 8: NCIPR Dataset + Kenya and Malawi Datasets + Automated Ensemble ML](#sec59)   10. [Experiment 9: NCIPR Dataset + Kenya and Malawi Datasets + AutoML with Different Generations](#sec510)   11. [Experiment 10: NCIPR Dataset + Kenya and Malawi Datasets + Automated Ensemble ML with Different Ensemble Size](#sec511)   12. [Experiment 11: NCIPR Dataset + Kenya and Malawi Datasets + Oversampling Technique + AutoML Single Classifier](#sec512)   13. [Experiment 12: NCIPR Dataset + Kenya and Malawi Datasets + Oversampling Technique + Automated Ensemble ML](#sec513)   14. [Experiment 13: Synthetic Data + AutoML Single Classifier](#sec514)   15. [Experiment 14: Synthetic Data + Automated Ensemble ML](#sec515)   16. [Conclusion](#impconclu) | 59  59-60  60-61  61-62  62-63  63-64  64-65  65-66  66-67  67  68  68-69  69-70  70-71  71-72  72 |
| **C**[**HAPTER 6 PRELIMINARY RESULTS**](#prelimre) |  |
| * 1. [Introduction](#prelimintro)   2. [Comparison between AutoML Single Classifier and Automated Ensemble ML](#automlvsensembleml)   3. [Comparison of AutoML Single Classifiers in Different Generations](#exp3re)   4. [Comparison of Automated Ensemble ML in Different Ensemble Size](#exp4re)   5. [Comparison of AutoML Single Classifiers and Automated Ensemble ML in Feature Reduction](#sec65)   6. [Comparison of AutoML Single Classifiers and Automated Ensemble ML in Unseen Test Set](#sec66)   7. [Comparison of AutoML Single Classifiers and Automated Ensemble ML in Unseen Test Set with Manipulated Parameter](#sec67)   8. [Comparison of AutoML Single Classifiers and Automated Ensemble ML in Unseen Test Set with Oversampling Technique](#sec68)   9. [Comparison of AutoML Single Classifiers and Automated Ensemble ML with Synthetic Data](#sec69)   10. [Overall Experimental Results](#overallre)   11. [Conclusion](#prelimconclu) | 73  73-74  74-75  75-76  77-78  78-79  80-81  81-82  82-83  84-86  86 |
| [**CHAPTER 7 CONCLUSION AND FUTURE WORK**](#conandfuture) |  |
| * 1. [Introduction](#conclufuintro)   2. [Summary](#summary)   3. [Main Findings](#mainf)   4. [Limitations](#limit)   5. [Future Works](#futurework) | 88  88-89  89-90  90  90 |
| [**References**](#ref)  [**Appendix A: Proposal Revision/ Progress Revision**](#appendixA)  [**Appendix B: Meeting Logs**](#appendixB)  [**Appendix C: Turnitin Plagiarism Proposal**](#appendixC) | **xiv**  **xxi**  **xxii**  **xxiii** |

**LIST OF TABLES**

|  |  |
| --- | --- |
|  | **Page** |
| [Table 1: Summary Table of Statistical Study on Long COVID-------------------------------](#table1) | 9-12 |
| [Table 2: Summary Table of Ensemble Models Used in the Healthcare Prediction](#table2)-------- | 13-16 |
| [Table 3: Summary Table of AutoML Used in the Healthcare Prediction](#table3)-------------------- | 22-27 |
| [Table 4: Example of Different Versions Used in Two OS](#table4)------------------------------------- | 31 |
| [Table 5: The Description of Each Selected Feature from the NCIPR dataset](#table5)-------------- | 33-34 |
| [Table 6: The Description of Each Selected Feature from the Kenya&Malawi dataset](#table6)---- | 35-36 |
| [Table 7: Sizes of Train, Validation and Test Sets](#table7)---------------------------------------------- | 36 |
| [Table 8: Description of Calculations Accuracy, Specificity and Sensitivity in Confusion Matrix](#table8)------------------------------------------------------------------------------------------------- | 37-38 |
| [Table 9: Summary of Baseline Experiments](#table9)---------------------------------------------------- | 49 |
| [Table 10: Summary of Experiments with Manipulated Parameter Settings](#table10)---------------- | 50 |
| [Table 11: The Description of Features from the NCIPR dataset after Feature Deduction](#table11)-------------------------------------------------------------------------------------------- | 51-52 |
| [Table 12: Summary of Experiments with Feature Reduction](#table12)-------------------------------- | 52-53 |
| [Table 13: The Renamed Columns in NCIPR, Kenya and Malawi Datasets](#table13)----------------- | 54 |
| [Table 14: Summary of Experiments with New Imported Test Set](#table14)-------------------------- | 54-55 |
| [Table 15: Summary of Experiments with Manipulated Parameter Settings and New Imported Test Set](#table15)---------------------------------------------------------------------------------- | 56 |
| [Table 16: Summary of Experiments with Oversampling Technique](#table16)------------------------- | 57 |
| [Table 17: Summary of Experiments with Synthetic Data](#table17)------------------------------------- | 58 |
| [Table 18: The Experimental Results Generated from Each Loop in Different Generations (Experiment 3)-----------------------------------------------------------------------](#table18) | 62 |
| [Table 19: The Experimental Results Generated from Each Loop in Different Ensemble Size (Experiment 4)](#table19)-------------------------------------------------------------------------------- | 63 |
| [Table 20: The Experimental Results Generated from Each Loop in Different Generations (Experiment 9](#table20))----------------------------------------------------------------------- | 67 |
| [Table 21: The Experimental Results Generated from Each Loop in Different Ensemble Sizes (Experiment 10](#table21))------------------------------------------------------------------------------ | 68 |
| [Table 22: The Experimental Results of AutoML Single Classifiers](#table22)--------------------------- | 75 |
| [Table 23: The Experimental Results of Ensemble ML](#table23)----------------------------------------- | 76 |
| [Table 24: The Experimental Results of AutoML Single Classifiers and Ensemble ML in terms of Feature Deduction](#table24)----------------------------------------------------------------------- | 78 |
| [Table 25: The Experimental Results of AutoML Single Classifiers and Ensemble ML in Unseen Test Set](#table25)------------------------------------------------------------------------------------ | 79 |
| [Table 26: The Experimental Results of AutoML Single Classifiers and Ensemble ML with New Test Set and Manipulated Parameter](#table26)----------------------------------------------- | 80-81 |
| [Table 27: The Experimental Results of AutoML Single Classifiers and Ensemble ML with Oversampling Technique in Unseen Test Set with Oversampling Technique](#table27)-------- | 82 |
| [Table 28: The Experimental Results of AutoML Single Classifiers and Ensemble ML with Synthetic Data](#table28)--------------------------------------------------------------------------------- | 83 |
| [Table 29: Overall Experimental Results](#table29)--------------------------------------------------------- | 85-86 |
| [Table 30: Table Of Objectives and Progress](#table30)--------------------------------------------------- | 89 |

**LIST OF FIGURES**

|  |  |
| --- | --- |
|  | **Page** |
| [Figure 1: Long COVID Prediction](#fig1) --------------------------------------------------------------- | 6 |
| [Figure 2: Taxonomy Tree of AutoML](#fig2) ------------------------------------------------------------ | 27 |
| [Figure 3: Transformation of Multi-class variable to Binary Class Variable](#fig3)------------------ | 37 |
| [Figure 4: Auto-sklearn Framework](#fig4) -------------------------------------------------------------- | 38 |
| [Figure 5: Ensemble Selection Applied in Auto-sklearn](#fig5)---------------------------------------- | 39 |
| [Figure 6: Example Machine Learning Pipeline and the Parts Automated by TPOT](#fig6)------- | 39 |
| [Figure 7: GP Algorithm Applied in TPOT](#fig7)-------------------------------------------------------- | 40 |
| [Figure 8: Process of Data Splitting for Baseline Experiments](#fig8) ------------------------------- | 41 |
| [Figure 9: Process of Data Splitting for Experiments with Feature Reduction](#fig9)-------------- | 42 |
| [Figure 10: Process of Data Splitting for Experiments with Synthetic Data Generation](#fig10)--- | 43 |
| [Figure 11: Confusion Matrix of Long COVID Prediction (Scikit-learn version)](#fig11)------------- | 43 |
| [Figure 12: Overall Flow of Activities](#fig12)------------------------------------------------------------- | 45 |
| [Figure 13: Code Fragment of Transforming Data Type](#fig13)--------------------------------------- | 47 |
| [Figure 14: Code Fragment of AutoML Single Classifier (TPOT)](#fig14)------------------------------ | 48 |
| [Figure 15: Code Fragment of Ensemble ML (Auto-sklearn)](#fig15)---------------------------------- | 49 |
| [Figure 16: Code Fragment of Manipulate parameter ‘generations’ in AutoML Single Classifier (TPOT)](#fig16)------------------------------------------------------------------------------------ | 50 |
| [Figure 17: Code Fragment of Manipulate parameter ‘ensemble\_size’ in Ensemble ML (Auto-sklearn)---------------------------------------------------------------------------------------](#fig17) | 50 |
| [Figure 18: Code Fragment of Mapping Values](#fig18)------------------------------------------------- | 53 |
| [Figure 19: Code Fragment of Synthetic Data Generation with DataSynthesizer](#fig19)---------- | 58 |
| [Figure 20: The Progress Bar and Current Best Score](#fig20) ---------------------------------------- | 59 |
| [Figure 21: The Output Fragment of AutoML Single Classifier (Experiment 1)](#fig21)----------- | 60 |
| [Figure 22: The Metric Scores of AutoML Single Classifier for (Experiment 1)](#fig22)------------- | 60 |
| [Figure 23: The Output Fragment of Automated Ensemble ML for (Experiment 2)](#fig23)------- | 61 |
| [Figure 24: The Metric Scores of Automated Ensemble ML for (Experiment 2)](#fig24)------------ | 61 |
| [Figure 25: The Output Fragment of Ensemble ML in Different Ensemble Sizes (Experiment 4)](#fig25)-------------------------------------------------------------------------------------- | 63 |
| [Figure 26: The Output Fragment of AutoML Single Classifier (Experiment 5](#fig26))------------- | 64 |
| [Figure 27: The Metric Scores of AutoML Single Classifier (Experiment 5)](#fig27)----------------- | 64 |
| [Figure 28: The Output Fragment of Automated Ensemble ML (Experiment 6](#fig28))------------ | 65 |
| [Figure 29: The Metric Scores of Automated Ensemble ML (Experiment 6](#fig29))---------------- | 65 |
| [Figure 30: The Output Fragment of AutoML Single Classifier (Experiment 7)](#fig30)------------- | 66 |
| [Figure 31: The Metric Scores of AutoML Single Classifier (Experiment 7)](#fig31)----------------- | 66 |
| [Figure 32: The Output Fragment of Automated Ensemble ML (Experiment 8)](#fig32)------------ | 66 |
| [Figure 33: The Metric Scores of Automated Ensemble ML (Experiment 8)----------------](#fig33) | 67 |
| [Figure 34: The Output Fragment of AutoML Single Classifier (Experiment 11](#fig34))----------- | 69 |
| [Figure 35: The Metric Scores of AutoML Single Classifier (Experiment 11](#fig35))---------------- | 69 |
| [Figure 36: The Output Fragment of Automated Ensemble ML (Experiment 12](#fig36))---------- | 70 |
| [Figure 37: The Metric Scores of Automated Ensemble ML (Experiment 12](#fig37))--------------- | 70 |
| [Figure 38: The Value Count of Each Category with Synthetic Data](#fig38)------------------------- | 70 |
| [Figure 39: The Output Fragment of AutoML Single Classifier (Experiment 13)](#fig39)----------- | 71 |
| [Figure 40: The Metric Scores of AutoML Single Classifier (Experiment 13](#fig40))---------------- | 71 |
| [Figure 41: The Output Fragment of Automated Ensemble ML (Experiment 14](#fig41))---------- | 71 |
| [Figure 42: The Metric Scores of Automated Ensemble ML (Experiment 14](#fig42))--------------- | 72 |
| [Figure 43: Comparison of AutoML Single Classifier and Ensemble ML in terms of Baseline Models](#fig43)------------------------------------------------------------------------------------- | 74 |
| [Figure 44: Bar Chart of Metrics Scores VS Generations Running-](#fig44)--------------------------- | 75 |
| [Figure 45: Bar Chart of Metrics Scores VS Ensemble Sizes](#fig45)----------------------------------- | 76 |
| [Figure 46: Bar Chart of Metrics Scores VS Type of Model in terms of Feature Deduction](#fig46)-------------------------------------------------------------------------------------------- | 77 |
| [Figure 47: Bar Chart of Metrics Scores VS Type of Model in Unseen Test Set](#fig47)------------ | 79 |
| [Figure 48: Bar Chart of Metrics Scores VS Type of Model in Unseen Test Set with Oversampling Technique](#fig48)--------------------------------------------------------------------------  [Figure 49: Bar Chart of Metrics Scores VS Type of Model with Synthetic Data](#fig49)----------- | 82  83 |

**CHAPTER 1**

**INTRODUCTION**

* 1. **Introduction**

This project is focused on using automated machine learning (AutoML) to predict long COVID symptoms, as post-COVID symptoms are more spontaneous and unpredictable when observed by humans alone.

There are six sections in this chapter. The first section provides an overview of the chapter. The background of the situation and some basic reviews of the long COVID and AutoML are explained in the following section. Section three summarises the existing situation and discusses the problem statement that has been established. Section four outlines the project's goal as a roadmap for resolving the issues that have arisen. The project's scope is stated in section five. The organisation's conclusion is stated in the chapter's concluding section.

* 1. **Problem Background**

The coronavirus disease 2019 (COVID-19) is caused by the SARS-CoV-2 virus that spreads via an infected person's mouth or nose. Covid-19 is continually evolving through mutations; the mutation rate is great and fast, making it impossible for humans to prevent it, despite immunisations being available since 2020. Most persons who get COVID-19 have mild symptoms and recover without any particular treatment as the virus has changed and vaccinations have been developed. There have, however, been a few unusual examples discovered. Patients may endure a wide range of symptoms after recovering from COVID-19, which is referred to by numerous labels, including "long COVID" and "post-COVID disorders," according to reports from around the world [(Mikkelsen, et al., 2021)](#Mikkelsen). The long COVID is differentiated from common COVID symptoms, it is the situations that develop during or after COVID-19 and the symptoms last at least four weeks or a few months. A wide range of symptoms have been described, but none can be explained by any other diagnosis [(Mikkelsen, et al., 2021).](#Mikkelsen)

Two cases of COVID-19-associated organising pneumonia (OP) have been recorded in Malaysia, both occurring more than four weeks after the onset of symptoms [(Ng, et al., 2021)](#ng). The first case was a 58-year-old woman with mild cough and fatigue while case 2 was an 81-year-old man with cough and dyspnoea. The patient in the first example had persistent fatigue and worsening exertional dyspnoea. The second patient had a worsening cough and new-onset pleuritic chest discomfort, as well as radiological consolidation. Both patients responded nicely to steroid treatment for 12 weeks. Another instance is a previously healthy 28-year-old lady who was diagnosed with COVID-19 and suffered from a series of symptoms for three months, including retrosternal pain, shortness of breath, impaired memory, and severe muscle pain [(Taribagil, Creer, & Tahir, 2021).](#Taribagil) Furthermore, psychological symptoms have been recorded in addition to physical and intense post-symptoms. COVID-19 induced severe anxiety and psychotic symptoms in a middle-aged man with no mental history, according to a case report from London [(Kozato, Mishra, & Firdosi, 2021).](#Kozato) After a few weeks of his discharge, he couldn't sleep and became increasingly nervous. The new development of psychosis owing to COVID-19 infection is highlighted in this case report.

ML methods are currently being used to develop studies on COVID-19 cluster prediction, death rate prediction, and ICU period prediction. For example, based on the apparent symptoms, the Support Vector Machine (SVM) is used to analyse and forecast the presence of COVID-19 in a person [(Nayve Villavicencio, Jeng, & Hsieh, 2021)](#Villavicencio). The COVID-19 dataset was obtained from Kaggle by the researchers. The proposed model had a 98.02 per cent accuracy rate and a 0.198 mean misclassification error rate. Additionally, the social network analysis (SNA) tool was used in conjunction with the regression technique and the Long-Short Term Memory (LSTM) model to perform clustering using the acquired Twitter text data to quickly identify the cluster of people who will test positive for COVID-19 by determining whether or not they are covidiots [(Nugroho, et al., 2021)](#Nugroho). The phrase "covidiots" refers to people who refuse to participate in COVID-19 combat operations, such as refusing to wear masks. People in the provinces of Banten, DIY, Central Java, and West Java had a significant rate of unfavourable sentiment and science denial about COVID-19, according to the results. This suggests that more people in the province may be tweeting about the negative features of COVID-19.

From the review of [(Waring, Lindvall, & Umeton, 2020)](#waring), statistical and machine learning technologies could be used to conduct a variety of studies with the deployment of electronic health records (EHR). The digitisation of health information has advanced throughout the COVID-19 pandemic era, allowing data scientists to use these records to translate data into useful insights. On the other hand, obtaining the essential features to illustrate machine learning (ML) models on health data, frequently necessitates expert domain expertise. The feature selection or feature engineering stages take time and must be repeated numerous times to achieve the best results. The emergence of AutoML which includes an automated feature engineering mechanism has solved this difficulty. The selected ML model will be run on a subset of features, and the set of features with the greatest performance will be chosen. Automated hyperparameter optimization is another important feature of AutoML. Every ML model has its unique set of hyperparameters that must be manually specified and some of them have crucial implications on performance, necessitating hyperparameter adjustment. AutoML has a function that allows you to quickly identify an optimal hyperparameter value or a set of hyperparameter values that maximises the performance metric for a machine learning activity. AutoML is popular because it has a pipeline optimization function that may be utilised by a non-expert in a range of jobs. Auto-WEKA, Auto-sklearn, TPOT, and other pipeline optimizers are examples. Each pipeline optimizer helps automate the machine learning process by performing one or more tasks.

There is minimal research on the long COVID prediction due to people now being more concerned about the initially emerging symptoms. However, the US Department of Health and Human Services and the Department of Justice stated in response to an evaluation report [(Mikkelsen, et al., 2021)](#Mikkelsen) confirming that long COVID is a type of disability because its effects interfere with daily activities, thus people should have more concerns on this long-term effect. According to the results of an observational study of patients hospitalised with COVID-19, 40% of patients were unable to return to normal activities after being discharged from the hospital [(Chopra, et al., 2021)](#chopra). The biggest challenge in diagnosing post-COVID-19 symptoms is a paucity of data; self-reporting questionnaires and observational studies can only provide limited information. Health professional now does the majority of the evaluations of patients' persistent complaints, which is time-consuming and labour-intensive. As a result, ML models may aid in identifying and predicting the occurrence of long COVID in a person by identifying and analysing the relevant features that produced long COVID.

* 1. **Problem Statement**

The main cause for someone getting a long COVID is still on explored, however, it could be linked to the COVID-19 variations' characteristics, vaccination status, medical therapy received during the recovery phase, or medical history. Although there have been statistical studies on the relationships between patient demographics and the occurrence of persistent COVID symptoms, more advanced and faster approaches could be employed to address this issue. Hence, several problem statements have been identified. It was discovered that (i) only statistical tools and clinical observations have been used to study the relevant aspects that led to the formation of long COVID. Furthermore, (ii) there is less research on applying ML models to forecast long COVID. Besides, (iii) There is only a little amount of data consisting of the lasting effects on patients that have been published and are available to use.

* 1. **Project Objectives**

1. To curate and modify datasets from the existing long COVID datasets based on the common features in order to identify the most relevant features that develop long COVID by applying data processing tools and automated feature engineering tools.
2. To design a comparison of automated machine learning models and ensemble models in terms of predicting the occurrence of long COVID.
3. To implement machine learning classifiers and ensemble models using Auto-sklearn and TPOT to predict the occurrence of long COVID.
4. To evaluate the accuracy, precision, and sensitivity of machine learning models and ensemble models for predicting the presence of long COVID.
   1. **Project Scope**

The programming language used for this project is Python, which also provides open-source AutoML modules, making it simple to implement codes on datasets. The datasets used originate from the "Novel Coronavirus (COVID) Illness – Patient Report (NCIPR)" [(Thomason, Werchan, & Hendrix, 2021)](#thomason) and Wellcome team survey data in Kenya and Malawi [(Wellcome, 2022)](#wellcome). Both datasets were compiled using responses from confirmed COVID-19 patients ranging in age from 18 to 98, hence this project excludes infants, children, and teenagers. The NCIPR survey included individuals diagnosed with COVID-19 at New York University Langone Medical Center (NYU Langone), whereas the Kenya&Malawi dataset included patients from Kenya and Malawi. Without pre-processing, such as filtering out missing data rows, the data sizes are 1670 and 1665, respectively. These two datasets contain fundamental information such as the individuals' demographics, such as their age and medical history, as well as the self-report, which includes secondary symptoms, vaccination status, and so on. AutoML will choose the classifiers through optimization, but in this project, it will be limited to considering the ML models and will not include neural network (NN) or deep learning models. The two AutoML packages chosen are Auto-sklearn and Tree-based Pipeline Optimization (TPOT). Auto-sklearn was created based on the Scikit-learn (sklearn) package and includes 15 classifiers [(Feurer, et al., 2015)](#Feurer). One of the benefits of Auto-sklearn is that it will automatically build ensemble models. Rather than discarding some marginally good models, it is preferable to reuse them and build an ensemble model to generate a more efficient model. Genetic programming (GP) was used by TPOT as a different optimisation strategy [(Le, Fu, & Moore, 2020)](#le). As a result, the accuracy, precision, and sensitivity of the models developed from Auto- sklearn will be compared to the single classifier developed from TPOT. This initiative is intended to assist the general population, so the results can be used as a model for others.

* 1. **Organization of the Report**

There are four chapters in all in this report. The project overview, objectives, problem background, project scope, and project organisation are all covered in Chapter 1. Chapter 2 provides the literature review conducted for this project, as well as related publications demonstrating AutoML's uses in health care. The methods that will be applied in different stages of this project, such as data collecting, data pre-processing, feature engineering, model development, and model evaluation, are discussed in Chapter 3. The preliminary implementation and system testing are briefly explained in Chapter 4.

**CHAPTER 2**

**LITERATURE REVIEW**

* 1. **Introduction**

The existing clinical investigations on long COVID and the state-of-the-art of using EML and AutoML in the healthcare industry are discussed in this chapter. To cut down on these high development costs and reduce the burden on data scientists, a unique idea of automating the entire ML process, dubbed AutoML, has been developed. Data preparation, feature engineering, model generation, and model validation are typically included in AutoML and all are automated processes. Figure 1 displays the fundamental parts of long COVID prediction, similar to other disease predictions, the relevant features are usually determined by health professionals through statistical analysis, and only then the advanced prediction is attempted. Long COVID prediction using machine learning is currently rare.

Graphical user interface, diagram, text, application, chat or text message

Description automatically generated

Figure 1: Long COVID Prediction

There are four sections in this chapter. The first section provides an overview of the chapter. The second section describes the long COVID analyses, which were conducted to uncover the relevant elements of long COVID prediction. The ensemble learning approaches used in the health care area are discussed in the third part. The fourth section discusses the existing AutoML libraries and platforms provided and applied by the authors in the health care field. The next section is the critical summary of all review. Last section is the short conclusion for this chapter.

* 1. **Long COVID Analysis**

By applying the statistical tools such as test, Kruskal-Wallis rank-sum test on the survey data that has been responded by patients who were positive with COVID-19, the predicting factors of post-acute sequelae of COVID-19 are investigated [(Knight, et al., 2022)](#Knight). 437 adult COVID-19 patients replied to the study, with varying degrees of severity of illness: 77 per cent were between 3 and 6 months after the commencement of infection. 34.9 per cent had prolonged symptoms, and 11.5 per cent were admitted to the hospital. Fatigue was the most prevalent symptom with 75.9 per cent, followed by poor sleep quality 60.3 per cent. Female sex and a poor psychological impact of the condition were both predictive factors for post-acute COVID. The sense of having severe COVID-19 illness was substantially linked with age, hospitalisation, persistent symptoms, psychological impact including anxiety and depression, and time away from work. COVID-19 post-acute effects were not substantially linked to hospitalisation.

Over the course of seven months, an online survey is done by a COVID-19 funded group on a social media platform, including estimates of the prevalence of 203 symptoms in ten organ systems and 66 symptoms [(Davis, et al., 2021)](#Davis). According to the statistics, the majority of respondents almost 91 per cent had a recovery time of more than 35 weeks. Participants had an average of 55.9+/- 25.5 (mean+/-STD) symptoms across an average of 9.1 organ systems during their illness. Fatigue, post-exercise malaise, and cognitive impairment were the most common symptoms after month six. With approximately 88 per cent of responders, cognitive impairment or memory difficulties were common across all age categories.

A cohort study was undertaken with the primary goal of determining the relationship between the female gender and long COVID syndrome, and the secondary goal of identifying predictors of long COVID syndrome using multivariable logistic regression analysis [(Bai, et al., 2022)](#Bai). There were 377 participants in this study, which took place at San Paolo Hospital in Milan. Long COVID syndrome was diagnosed in 260/377 patients, accounting for 69 per cent of the total. The most prevalent reported symptom was weariness, which accounted for 39.5 per cent of the patients (149/377), whereas depression was recorded by 10.6 per cent of the patients (40/377). In multivariable analysis, the female gender was independently associated with long COVID syndrome, with an adjusted odds ratio (AOR) of 3.3 compared to males, a 95 per cent confidence interval (CI) of 1.8-6.2 and a p-value less than 0.0001.

Another study looked into the prevalence of physical, psychological, and sleep disorders, as well as their impact on the quality of life in the general population [(Orrù, et al., 2021).](#Orrù) An online survey was used to collect demographic information, COVID-19 information, sleep problems, and quality of life data from 507 people. The Insomnia Severity Index (ISI) and the EuroQol-5D (EQ-5D) were used to determine the severity of sleep disruptions and quality of life, respectively. Headache, weariness, muscular aches or myalgia, articular pains, cognitive impairment, loss of attention, and loss of smell were the most common symptoms related to "long COVID." In addition, the participants had substantial levels of sleeplessness with a p-value less than 0.05 and a lower overall quality of life with a p-value less than 0.05.

There have been no reliable estimates of the incidence and co-occurrence of long-COVID characteristics, their connection to age, sex, or infection severity, or the amount to which they are specific to COVID-19, so a study was undertaken to investigate this problem [(Taquet, et al., 2021).](#Taquet) For 9 core features of long-COVID (breathing difficulties/breathlessness, fatigue/malaise, chest/throat pain, headache, abdominal symptoms, myalgia, other pain, cognitive symptoms, and anxiety/depression), the incidence and co-occurrence within 6 months and 3 to 6 months after COVID-19 diagnosis were calculated. 57.00 per cent of COVID-19 survivors had one or more long-COVID features reported during the course of the 6-month period and 36.55 per cent between 3 and 6 months. Anxiety/depression (22.82 per cent in the 1- to 180-day period; 15.49 per cent in the 90- to 180-day period), abnormal breathing (18.71 per cent; 7.94 per cent), abdominal symptoms (15.58 per cent; 8.29 per cent), and fatigue/malaise (22.82 per cent in the 1- to 180-day period; 15.49 per cent in the 90- to 180-day period) were the most common features (12.82 per cent; 5.87 per cent).

* + 1. **Summary for Long COVID Analysis**

To date, the majority of studies on long COVID have focused on statistical analysis to identify key factors in predicting long COVID incidence. The majority of the researchers concluded from the research that exhaustion and psychological characteristics play crucial roles because they had a very low p-value in most of the studies. However, a patient's prior acute medical history has no bearing on the likelihood of a long COVID. Some of the research revealed interesting findings, such as the fact that females have long COVID than males. According to statistics, although long COVID symptoms are usually not acute, they have a significant impact on people's daily lives because most patients must deal with psychological issues and stress, causing them to be unable to perform well in their workplace. Table 1 below shows the summary of the statistical study on long Covid.

Table 1: Summary Table of Statistical Study on Long COVID

|  |  |  |  |
| --- | --- | --- | --- |
| **Author(s)** | **Task** | **Dataset** | **Findings** |
| [(Bai, et al., 2022)](#Bai) | * The relationship between the feminine gender and long COVID syndrome was investigated. * Using multivariable logistic regression analysis, find predictors of long COVID syndrome. | San Paolo Hospital in Milan | * Female gender was independently associated with long COVID syndrome, AOR = 3.3, 95 CI of 1.8-6.2, p < 0.0001. * Most prevalent reported symptom = fatigue (39.5%). |
| [(Davis, et al., 2021)](#Davis) | * To characterise the patient experience and recovery process in those who have COVID-19 illness, with a focus on the long COVID experience. | A survey created by Body Politic online COVID-19 support group and the Patient-Led Research Collaborative | * Fatigue, post-exercise malaise and cognitive impairment were the most common symptoms. * 88% of respondents faced cognitive impairment issues. |
| [(Knight, et al., 2022)](#Knight) | * To determine the prevalence, characteristics, and predictors of patient-reported severe coronavirus disease 2019 (COVID-19) infection and COVID-19 post-acute sequelae (PASC). | A survey study with a retrospective assessment of the electronic health record (EHR) was conducted at Mayo Clinic in Jacksonville, Florida. | * Women and patients with self-reported severe disease have a higher prevalence of extended symptoms: fatigue, dyspnoea, anosmia, and poor sleep. |
| [(Goel, et al., 2022)](#goel) | * After therapy with systemic steroids, individuals with long-term COVID were evaluated three months later. | Department of Pulmonary Medicine at Viswanathan Chest Hospital, Vallabhbhai Patel Chest Institute, University of Delhi, Delhi. | * Breathing problems went down from 91.% to 44.89% with p<0.001. * Cough went down from 77.55% to 8.16% with p<0.001. |
| [(Orrù, et al., 2021)](#Orrù) | * To look at the incidence of physical, psychological, and sleep disorders, as well as the general population's quality of life, throughout the continuing pandemic. | Online survey from 507 individuals who has positive with COVID 19. | * Headache, weariness, muscular aches/myalgia, articular pains, cognitive impairment, loss of attention, and loss of smell were the most common symptoms related to "long COVID." * The participants had substantial sleeplessness (p < 0.05) and a lower overall quality of life (p < 0.05). |
| [(Taquet, et al., 2021)](#Taquet) | * To estimate the incidence and co-occurrence of long-COVID characteristics, their connection to age, sex, or infection severity, or the amount to which they are specific to COVID-19. | A linked EHRs recording anonymized data from 59 healthcare organizations | * 57.00% of COVID-19 survivors had one or more long-COVID features reported during the course of the 6-month period. * Anxiety/depression (22.82% in the 1- to 180-day period; 15.49% in the 90- to 180-day period). |
| [(Karaarslan, Güneri, & Kardeş, 2022)](#Karaarslan) | * To document the detailed characteristics including severity, type, and locations of rheumatic and musculoskeletal symptoms along with other COVID-19 persistent symptoms in hospitalized COVID-19 survivors at 3 and 6 months. | Hospitalised patients from Gülhane  Training and Research Hospital. | * At least one rheumatic and musculoskeletal symptom was reported by 74.6 per cent of the participants. * 31.6: weariness * 18.6: joint pain * 15.1%: myalgia |
| [(Sudre, et al., 2021)](#Sudre) | * To create a model that might be used to identify people at risk of long COVID for preventative or treatment trials, as well as to arrange education and rehabilitation services. | COVID Symptom Study app | * Long COVID was associated with fatigue, headache, dyspnoea, and anosmia, and it was more common as people were older, had a higher BMI and were female. |
| [(Blomberg, et al., 2021)](#Blomberg) | * To investigate the spectrum of symptoms in milder cases. | Hospitalized and home-isolated patients in Bergen, Norway. | * 52% young individuals: loss of taste or smell, fatigue, memory issue, dyspnoea, decreased attention. |
| [(Sykes, et al., 2021)](#Sykes) | * At a follow-up patient consultation, the symptom load was measured. | Hull University Teaching Hospitals | * Females were significantly to have anxiety with p = 0.001, fatigue with p = 0.004 and myalgia with p = 0.022 as long COVID symptoms. |

* 1. **Ensemble Machine Learning in Health Care**

An experimental investigation of ensemble deep learning approaches for the processing of time-series data supplied by wearable devices using small datasets was conducted [(Mauldin, et al., 2020)](#Mauldin). The main contribution of this research is the application of ensemble deep learning for fall detection on two independent datasets. Second, investigating the impact of data heterogeneity on classification performance, as well as the extent to which ensemble deep learners can diversify to better capture that heterogeneity, resulting in higher classification accuracy than a single deep learning model. The stacking ensemble strategy of RNN outperforms a single Recurrent Neural Networks (RNN) model trained on the same data samples. The experiment used two types of ensemble techniques: stacking and boosting. The results showed that stacking outperformed boosting, with an f1-score of 0.922 on the first dataset and 0.980 on the second.

A unique self-supervised learning-enabled system was proposed with the goal of COVID-19 cough categorization [(Xue, & Salim, 2021)](#Xue). Three ensemble models were created, with the ensemble model achieving the best outcome which was a ROC-AUC score of 90.03 per cent. The ensemble comprising Transformer-CP (developed by the authors) and Gated Recurrent Units (GRU) was the best ensemble model. A study was undertaken with the goal of using image feature fusion to diagnose COVID-19 in lung window computed tomography (CT) and the ensemble of three RNN with discriminant correlation analysis (DCA), a feature fusion method achieved the best performance with 97.86% of sensitivity, 96.35% of accuracy and 97.38% of precision [(Lu, et al., 2021).](#Lu)

Next, ensemble models like XGBoost (XGB) and Random Forest (RF) are examples of classical ensemble models. These two models were used to forecast the progression of pain alleviation and to generate model explanations [(Costa, et al., 2021).](#Costa) The best model in the model space with the plot of SHapley Additive exPlanations (SHAP) values to show the important features that affect the outputs were the XGB prototype models, providing gains of up to 9.86 per cent.

CoRSAI is a system that was created by combining ensemble learning and deep learning [(Avetisian, et al., 2021)](#Avetisian). For the processing of chest CT scans, the scientists reported a novel ensemble of previously proposed deep convolutional neural networks (CNN) specifically tailored for the COVID-19-induced pneumonia segmentation task. The Dice Similarity Coefficient (DSC) was used to assess model performance and CoRSAI scored 0.768 on the CovidCTSegmentation Dataset and 0.643 on the MosMedData Dataset, both of which were greater than the state-of-the-art from previous studies.

* + 1. **Summary for Ensemble Machine Learning**

Not only have classical methods such as RF and Gradient Boosting been found and refined for use on medical datasets, but more advanced ensemble techniques have also been discovered and enhanced. With the development of EHR, numerous types of datasets like MRI pictures, CT-scan images, and even the most basic format, tabular, have been available. Ensemble approaches can aid in averaging the performance of unique models and combining them to produce a more powerful model. Although there is not much of a case applying ensemble models in predicting Long COVID, there are some studies in predicting or classifying COVID-19 related symptoms. As a result, the Auto-sklearn ensemble approach was chosen to be used on the long COVID datasets in this project. The summary of ensemble models used in the healthcare industry is shown in Table 2.

Table 2: Summary Table of Ensemble Models Used in the Healthcare Prediction

|  |  |  |  |
| --- | --- | --- | --- |
| **Author(s)** | **Task** | **Dataset** | **Findings** |
| [(Mauldin, et al., 2020)](#Mauldin) | * To give an in-depth experimental examination of Ensemble Deep Learning algorithms for the processing of time-series data. | UniMiB Dataset, SmartFall Dataset | * A stacking ensemble strategy paired with an ensemble of RNN models outperforms a single RNN model trained on the same data samples. |
| [(Xue, & Salim, 2021)](#Xue) | * To develop a unique COVID-19 cough classification system based on self-supervised learning. | Coswara Dataset, COVID-19 Sounds | * The ensemble of the Transformer-CP (developed by the authors) and Gated Recurrent Units (GRU) achieved a ROC-AUC score of 90.03%. |
| [(Adiga, et al., 2021)](#adiga) | * To propose a forecasting pipeline that used a Bayesian ensembling methodology to combine probabilistic projections from diverse statistical, ML and mechanical methodologies. | Johns Hopkins University  Center for Systems Science and Engineering (JHU CSSE) | * Under such dynamic situations, a forecasting pipeline based on a trained ensemble can still be useful. * Showed that the proposed Bayesian Model Averaging Ensemble (BMA)'s average performance is comparable to that of the other approach with the lowest Mean Average Error (MAE). |
| [(Lu, et al., 2021)](#Lu) | * To use image feature fusion to diagnose COVID-19 in lung window computed tomography (CT). | Collected from their local hospital. | * The ensemble of three RNNs with discriminant correlation analysis (DCA), a feature fusion method achieved the best performance with 97.86% of sensitivity, 96.35% of accuracy and 97.38% of precision. |
| [(Costa, et al., 2021)](#Costa) | * To forecast the progression of pain alleviation. * To generate model explanations. | Collected by the authors. | * The best model in the model space with the plot of SHapley Additive exPlanations (SHAP) values to show the important features that affect the outputs were the XGB prototype models, providing gains of up to 9.86%. |
| [(Avetisian, et al., 2021)](#Avetisian) | * To propose a novel ensemble of CNNs specifically tailored for the COVID-19-induced pneumonia segmentation task. | CovidCTSegmentation, MosMedData | * CoRSAI scored 0.768 on the CovidCTSegmentation Dataset and 0.643 on the MosMedData Dataset. |
| [(Katsimpras, et al., 2022)](#Katsimpras) | * To create a ML model that can help identify dementia using routine clinical data. | Collected as part of the Oxford Project to Investigate Memory and Ageing  (OPTIMA) | * Showed that combining several ML algorithms can successfully define ways to predict the probability of getting dementia over the next five years with accuracy rates significantly higher than average. * Underbagging-Decision Tree (DT): Precision score = 0.63, Recall = 0.38, F1-score = 0.47. |
| [(Hensley, & Elgazzar, 2022)](#Hensley) | * To investigate how ensemble learning and a variety of other supervised learning systems might be used to classify stroke victims. | Kaggle stroke dataset | * XGB obtained the best performance: 98 per cent for accuracy, recall, precision and f1-scores |
| [(León, et al., 2021)](#León) | * To examine if heart rate variability (HRV) data from preterm and full-term babies might be utilised to identify their functional maturational age using an ensemble machine learning (EML) technique that combines linear regressions (LR) and random forest (RF) (FMA). | Digi-NewB cohort | * Allowing the model to use information from the data's features. * The model's performance was quite accurate, with a low MAE. |
| [(Hooda, Gupta, & Gupta, 2020)](#Hooda) | * To give a case study of the Malwa Belt in India, which has seen an increase in the total mortality rate owing to breast cancer. | Women suffering from breast disease in the Malwa Region of Punjab were surveyed. | * Bagging-based Random Forest and Adaboost name Bagoost ensemble model produced ROC score of 0.98, accuracy score of 98.21, and F-measure 0.98. |

* 1. **Automated Machine Learning in Health Care**
     1. **JADBIO**

Even if someone has no data science background, JADBIO's purpose-built AutoML platform provides cutting-edge AI tools and automation capabilities, allowing life-science professionals to quickly and easily construct and deploy accurate and interpretable predictive models. Linear, Ridge, and Lasso Regression (LR), Decision Trees (DT), Random Forests (RF), and Support Vector Machines (SVMs) with Gaussian and polynomial kernels are all available in JADBIO. Bootstrap corrected cross-validation, also known as hyperparameter optimisation, is used to evaluate the configurations. A built-in function in JADBIO employs stratified cross-validation with varied class weights. In the study of [(Danilatou, et al., 2020](#Danilatou)), they utilised JADBIO to predict early and late mortality in ICU patients, and they compared the ROC-AUC scores using cross-validation and the balancing algorithm SMOTE from the Scikit-learn library. The results were similar: using JADBIO, the AUC score was 0.925, whereas using SMOTE, the AUC score was 0.91. Their findings show that ideas have a high AUC and the same efficiency as the full feature space when it comes to forecasting early death. Besides, JADBIO is able to show the top features selected to do prediction thus users can have a better understanding of the model generated.

* + 1. **TPOT**

TPOT, which was designed to automate biomedical data science but is now capable of handling any machine learning assignment. It is an AutoML system based on genetic programming (GP) and Pareto optimization that will handle feature pre-processing, model selection, and hyperparameter optimisation. Random Forest (RF), K Nearest Neighbors (KNN), and other supervised classification operators are used in the TPOT system. At the start of each TPOT run, a predefined number of pipelines are constructed to form what is known as a population in GP. The set of pipelines that acted on the dataset is evolved using GP, and a fraction of them is kept based on their classification performance. When TPOT reaches convergence or after a user-defined number of runs, the best-performing pipeline is kept. TPOT was used in the PhysioNet 2021 challenge to classify ECG recordings made up of 12-lead, 6-lead, 4-lead, 3-lead, and 2-lead ECGs in order to predict cardiac abnormalities [(Bodini, Rivolta, & Sassi, 2021)](#Bodini). They set the TPOT into two different configurations with the fixed training time: the first was the default configuration, which searched a wide range of pipeline elements but took a long time to run, especially on large datasets; the second was the light configuration, which searched a narrow range of simple and fast-running pipeline elements to find the most optimal and simple model. The light TPOT has achieved a higher testing score, 0.35 and their model was ranked 27th. TPOT includes the function of feature engineering, based on the different models, different methods are implemented for the feature extraction and selection by identifying the correlation between the features. In a comparison of the diabetes prediction AutoML model with customised models [(Kulkarni, et al., 2021](#Kulkarni)), TPOT has AUC scores of 0.81. The results were compared to the results of the other three models, which were created using various pre-processing, sampling, and scaling methods. The manually created models beat the AutoML models because, as the authors point out, AutoML suffers from multi-class scenarios. However, AutoML's capabilities should not be overlooked because they can still be enhanced. TPOT can also be used to categorise photos. TPOT had a training accuracy of 0.83 in the experiment of exploring the sensitivity of CoV-19 detection using the CAD4COVID programme and evaluating the accuracy of the classifier performance using AutoML, and the best pipeline selected was KNN with a precision score of 0.60, recall score of 0.99 and AUC score of 0.98.

* + 1. **Auto-sklearn**

The Auto-sklearn framework is a Python sklearn library-based AutoML system. It uses 15 machine learning classifiers, four data pre-processing methods, and fourteen feature pre-processing methods to generate a large hypothesis space. Auto-sklearn defines a Combined Algorithm Selection and Hyperparameter Optimization (CASH) issue in such a space, and it uses Bayesian Optimization to find the best ML pipeline. We will use the word "pipeline" to refer to the pre-processing and used ML/DNN classifier, as well as its parameters and hyperparameters, to solve the problem. In the PhysioNet 2021 challenge, Auto-sklearn is also [used (Bodini, Rivolta, & Sassi, 2021)](#Bodini). Auto-sklearn was trained with a fixed number of leads but manipulated training time. The test accuracy score gained by the Auto-sklearn model in this contest was 0.30. MedMNIST is a pre-processed collection of medical image datasets. The Auto-sklearn team [(Yang, Shi, & Ni, 2021)](#YangJ) had poor performance on some datasets, such as the PathMNIST dataset, which achieved an AUC score of 0.186 but performed well on the DermaMNIST dataset, which achieved an AUC score of 0.906, and the PneumoniaMNIST dataset, which achieved an AUC score of 0.947. Furthermore, the authors found out that Auto-sklearn is more prone to overfitting.

* + 1. **Auto-Keras**

Auto-Keras is an automatic machine learning approach based on the Keras library in [Python (Jin, Song, & Hu, 2019)](#Jin). This algorithm is inspired by the complex neural architecture search (NAS) which aims to search for the best neural network model to carry out a task. According to the documentation, the Auto-Keras uses Bayesian optimization to navigate the search space. On MNIST, CIFAR10, and FASHIONMNIST benchmark datasets, the suggested methodology is compared to state-of-the-art NAS algorithms [(Ucuzal, YAŞAR, & Çolak, 2019](#Ucuzal)). On all of the datasets, the architectures discovered by their method had the lowest error rates in a short amount of time. The team selected data from Nanfang Hospital and Tianjin Medical University General Hospital to apply Auto-Keras to diagnose brain tumours from MRI images. According to the experimental results, all of the estimated performance metrics for classifying the types of brain tumours on the training dataset are greater than 98 per cent. On the testing dataset, all performance measures are higher than 91 per cent, with the exception of sensitivity and Matthew's Correlation Coefficient (MCC) performance metrics for a meningioma brain tumour. Auto-Keras was also utilised in the same contest, the PhysioNet 2021 challenge [(Bodini, Rivolta, & Sassi, 2021)](#Bodini), AutoKeras obtained 0.32 challenge scores. There were also two configurations set up: training time of 2.5h x #leads for each lead configuration and using the entire set of pipeline elements available, another was hypothesis space was reduced to only consider DNN composed of Convolutional, Dense, ResNet, and Xception layers, and training time was maintained at the same training time as the first configuration. Auto-Keras achieved an accuracy score of 0.864 in the classification test of MedMNIST datasets, which was the highest score in PathMNIST dataset with another manually tuned model [(Yang, Shi, & Ni, 2021)](#YangJ). This experiment tested the AutoML models' performance on all of the datasets in MedMNIST without performing any additional manual tuning, and the AutoML models performed well but might be improved. Wearing a mask is one of the protection methods used during the COVID-19 pandemic. To identify someone wearing a face mask, facial recognition and screening must be adjusted. As a result, a deep learning model based on AutoML is built to detect the presence of a face mask on someone's face [(El Gannour, et al., 2022).](#ElGannour) The acquired data on the face mask photos are given to the Auto-Keras model, which uses a camera to recognise faces in real-time. Other manually modified deep learning models are compared to the Auto-Keras model. In general, all models achieve a high level of precision, ranging from 98.94 per cent to 99.74 per cent, and the accuracy score outperforms earlier studies.

* + 1. **Google AutoML**

Google AutoML is a convenient platform that allows even non-expert developers to train models tailored to their needs. It has a number of features, including the ability to train models on visual, textual, and structured data. The MedMNIST files were sent to the Google AutoML platform in the same experiment, and the results were good. Furthermore, Google AutoML Vision has a good handle on overfitting concerns, whereas Auto-sklearn has this serious issue. This demonstrated the significance of adequate reductive bias in learning algorithms. The performance of a second-generation machine-learning method that incorporates both object detection and image classification is examined, and the results are compared to blinded radiologists [(Tahmasebi, et al., 2020)](#Tahmasebi). The object detection model's internal validation yielded a positive predictive value (PPV) of 68.31 per cent and a sensitivity of 86.81 per cent. The model had a sensitivity of 73.9 per cent, a specificity of 70.8 per cent, a positive predictive value (PPV) of 70.8 per cent, a negative predictive value (NPV) of 73.9 per cent, and overall accuracy of 66.7 per cent in the 47 nodules for risk categorization. In the same dataset, three radiologists are invited to identify the thyroid nodules and their performance showed is lower than the results generated by Google AutoML. Hence, this can be concluded that human error can be corrected by a machine in a short amount of time and energy.

* + 1. **AutoGluon**

AutoGluon is an open-source AutoML toolkit for deep learning that makes it easier for scientists, engineers, and students to work with deep learning models [(Erickson, et al., 2020)](#Erickson). It allows data scientists to achieve state-of-the-art performance with tabular datasets on tasks like image classification, object identification, text classification and supervised learning. The paper describing the use of AutoGluon to classify images can be found here [(Anwar, 2021).](#Anwar) AutoGluon was used on the COVID-19 CT scan images, and the ResNest14 architecture, which is one of the models in AutoGluon, obtained accuracy and F1-score of 89 per cent and 88 per cent, respectively.

* + 1. **H2O AutoML**

To build a large number of models in a short amount of time, the H2O AutoML algorithm depends on the fast training of H2O machine learning algorithms [(LeDell, & Poirier, 2020](#LeDell)). H2O AutoML uses a range of techniques, such as GBMs, Random Forests, Deep Neural Networks, GLMs, and others, to generate a healthy degree of diversity across candidate models, which stacked ensembles can use to create a powerful final model. From the same article by [(Kulkarni, et al., 2021),](#Kulkarni) H2O also be used to predict the occurrence of diabetes and obtained AUC scores of 0.82. It has the same problem as TPOT which is suffering from multi-class conditions. H2O could potentially be used to anticipate the level of anxiety and depression among Parkinson's disease participants through surveys done before and during the epidemic, allowing for timely intervention [(Kaur, et al., 2021](#Kaur)). The stacked ensemble model had a low root mean squared error (RMSE) of 2.841, while a deep learning model from H2O had the highest RMSE of 2.930. The authors pointed out that there is a dearth of studies on predicting and offering early intervention. H2O AutoML is possible to minimise the number of features, such as in their case, from 5,308 to 10, which can alleviate the time-consuming problem. H2O is used in the diagnosis of Parkinson's disease [(Nolazco-Flores, et al., 2021)](#Nolazco). The authors did not just apply H2O to the datasets, but instead allowed them to go through a series of steps that included data augmentation due to the tiny size of the data and feature processing such as applied spectral and cepstral features on Parkinson's disease diagnosis, and more. After that, they used H2O to clean up the data. The addition of spectral and cepstral features to the kinematics and statistics features significantly enhances classification accuracy, attaining a combined classification accuracy of 98.57 per cent in the experiments. This result demonstrates that their proposed model exceeds the best state-of-the-art result of 97.62 per cent.

* + 1. **Self-designed AutoML**

Self-designed AutoML, which involves altering parts of the data pre-processing methods, feature engineering methods, or hyperparameter optimisation algorithms in addition to using the existing AutoML platform or libraries, is a recent trend in the AutoML area. For lesion segmentation in 3D volumes, a new transformer-based AutoML (T-AutoML) is intended to simultaneously search for the optimum neural architecture, as well as the best combination of hyperparameters and data augmentation methodologies [(Yang, et al., 2021)](#Yang). The proposed method forecasts the relationship between different neural network configurations and training configurations. Meanwhile, they developed a predictor-based AutoML method that is both efficient and effective in terms of computing. It can cover almost all CNN framework components. In comparison to other existing methods in the literature, their model can achieve the most advanced performance in large-scale lesion segmentation datasets, according to their experiments. Furthermore, the proposed strategies have been demonstrated to be effective across a variety of datasets.

The second self-designed AutoML is named HOUSES which stands for hyperparameter optimization with surrogate-assisted evolutionary strategy [(Zhang, et al., 2021](#Zhang)). They integrated HOUSES to the multilayer convolutional neural network (ML-CNN), the preferred model for lung nodule classification, which can identify a competitive or even superior hyperparameter configuration than the manual search technique without incurring significant processing costs. On a numerical scenario, the HOUSES was compared to baselines using the trimodel and Branin functions to replicate hyperparameter optimization in DNNs. The results of the experiments on a variety of DNNs and datasets show that the proposed nonstationary kernel-based approach achieves better hyperparameter configuration than other methods like grid search, random search, HORD coordinate search via RBF and Dynamic coordinate search and the stationary kernel-based Gaussian kernel Bayesian optimization. The findings also show that designing a proper network structure is critical for improving performance and that hyperparameter modification may aid in realising the network's full potential.

* + 1. **Summary for AutoML**

In short, the present AutoML libraries are not restricted to the sections covered above; there are other AutoML libraries to investigate and implement. While not all of the AutoML models obtained the highest results, they all performed admirably and cut down on running time. Because diagnosis is a time-consuming process that necessitates caution, the use of AutoML can assist health practitioners and data scientists in reducing their workload. Little study has been done on using AutoML to predict long COVID, similar to ensemble learning, however, earlier research has shown that AutoML can predict diverse diseases independent of dataset type. AutoML may be used to classify images, recognise ECG signals, and analyse tabular data. There has been some study that compares the abilities of multiple AutoML on the same dataset and the results showed differently. This is due to the AutoML's many hyperparameter optimisation techniques or NAS. Some AutoML models use Bayesian optimization, while others use Random Search with Cross-Validation. Table 3 shows the summary of the paper review that has applied AutoML in the health care area while Figure 2 displays the taxonomy tree of AutoML’s hyperparameter optimisation or NAS techniques.

Table 3: Summary Table of AutoML Used in the Healthcare Prediction

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Author** | **Task** | **Dataset** | **AutoML platform/ library** | **Findings** |
| [(Danilatou, et al., 2020)](#Danilatou) | Predict short-term and long-term mortality for ICU patients with Venous thromboembolism (VTE). | Medical Information Mart  for Intensive Care (MIMIC-III) | JADBIO | * The model can find traits and create models to help increase ICU survival rates. * Dimensionality curse, missing data, imbalanced classes and co-dependency between distinct features were all solved. * AUC score for short-term mortality = 0.925. * AUC score for long-term mortality = 0.82. * Best model: Random Forest |
| [(Alnegheimish, et al., 2020)](#Alnegheimish) | * Predict whether a patient will show up to an appointment * Predict the length of stay, Readmission, Diagnosis Prediction, Mortality Prediction | * Kaggle Medical Appointment No-Shows dataset * MIMIC-III | AutoML library | * Accuracy score for Gradient Boosting Classifier (GB) = 0.91. |
| [(Ucuzal, YAŞAR, & Çolak, 2019)](#Ucuzal) | Brain tumours detection (Glioma/Meningioma/Pituitary) | Nanfang Hospital and Tianjin Medical University General Hospital: T1-weighted magnetic resonance images | AutoKeras (CNN) | * All of the estimated performance metrics for training data > 98%. * For testing dataset, all performance measures > 91%. |
| [(Bodini, Rivolta, & Sassi, 2021)](#Bodini) | Cardiac abnormalities (CA) classification from both 12-lead electrocardiograms (ECG). | PhysioNet 2021 challenge | Auto-sklearn, TPOT, Auto-Keras | * TPOT has achieved a higher testing score, 0.35 and their model was ranked 27th. |
| [(Kulkarni, et al., 2021)](#Kulkarni) | Develop diabetes prediction Machine Learning models without involving data scientists. | Pima Indian Diabetes Dataset (PIDD) | TPOT, H2O | * Both TPOT and H2O performed poorly than manual configured models in the multi-class classification scenario. * Accuracy of TPOT = 0.81 * Accuracy of H2O = 0.82 |
| [(Anwar, 2021)](#Anwar) | COVID-19 classification based on CT scan images. | COV19-CT-DB | AutoGluon | * 2D slices: accuracy score = 89%, F1-score = 88% * 3D slices: macro F1-score = 87.77%. |
| [(Kaur, et al., 2021)](#Kaur) | Predict the level of anxiety and depression in Parkinson’s disease participants. | Survey data from the Michael J. Fox Foundation’s online clinical study, Fox Insight. | H2O | * Lowest RMSE = 2.841 * Best model: stacked model |
| [(Izdihar, et al., 2021)](#Izdihar) | Detection of CoV-19 utilising CXR images. | 70 CXR pictures were collected from a tertiary hospital in Kuala Lumpur. | TPOT | * Accuracy score = 0.83 * Precision = 0.60 * Recall = 0.99 * AUC score = 0.98 * Best model: KNN |
| [(Dwivedi, et al., 2021)](#Dwivedi) | Classify Foetus health into normal, suspect or pathological. | Kaggle dataset | PyCaret | * Accuracy score = 94.53% 🡪 95.61% * Best model: Light Gradient Boosting Machine (LGBM) |
| ([Nolazco-Flores, et al., 2021)](#Nolazco) | Spectral and cepstral features were used on the Parkinson’s disease identification | (PaHaW) dataset | H2O | * Accuracy score = 98.57%. * The proposed model exceeds the best current state-of-the-art result of 97.62%. |
| [(Jain, Wadhwa, & Garg, 2019)](#Jain) | Designed and presented the use of AutoML techniques such as Bayesian optimization and Hyper Bands where the dataset is constrained, such as in the medical area. | Breast MRI from NCBI | Bayesian optimization (BO) and Hyperband (HB) (BOHB) were applied to the selected Neural Network model. | * Test accuracy = 89.57%. |
| [(Yang, Shi, & Ni, 2021)](#YangJ) | MedMNIST Classification Decathlon | 10 pre-processed datasets from MedMNIST. | Auto-sklearn, Auto-Keras, Google AutoML Vision | * Overfitting is well controlled by Google AutoML Vision, and significant overfitting is detected for Auto-sklearn. |
| [(Ayachit, et al., 2020)](#Ayachit) | Propose a data-driven machine learning model for predicting a person’s likelihood of receiving H1N1 and seasonal flu vaccinations. | Driven Data from the National 2009 H1N1 Flu Survey (NHFS). | MlBox, TPOT | * MlBox and TPOT performed poorly than the CatBoost which the authors manually tuned themselves. |
| [(El Gannour, et al., 2022)](#ElGannour) | Medical face masks recognition in images of people in public places. | Face mask on people images collected by using a camera to detect faces in real-time. | AutoKeras | * All models achieve a high level of precision, ranging from 98.94% to 99.74%. |
| [(Yang, et al., 2021)](#Yang) | Proposed a new automated machine learning algorithm, T-AutoML. | * LiTS 2017 * Medical Segmentation Decathlon | Self-designed: T-AutoML | * Cover almost all CNN framework components. * In comparison to other existing methods in the literature, their model can achieve the most advanced performance in large-scale lesion segmentation datasets, according to their experiments. |
| [(Tahmasebi, et al., 2020)](#Tahmasebi) | Utilised Thyroid nodule ultrasonography pictures for predicting high and low-risk mutations. | Department Picture Archiving and Communication System (PACS) | Google AutoML | * Sensitivity score = 73.9%, * Specificity score = 70.8% * Positive predictive value (PPV) = 70.8% * Negative predictive value (NPV) = 73.9% * Accuracy = 66.7% * The following is a comparison of the three radiologist’s performance in the same dataset: sensitivity of 53.6±6.6%, specificity of 65.2±6.4%, PPV of 59.7±2.9%, NPV of 59.5±2.1%, and overall accuracy of 59.5±2.2% |
| [(Zhang, et al., 2021)](#Zhang) | Proposed Gaussian surrogate model based on nonstationary kernels | - | HOUSES | * Other strategies fail to produce a better hyperparameter setting than the proposed nonstationary kernel-based approach. |

Diagram

Description automatically generated

Figure 2: Taxonomy Tree of AutoML

* 1. **Critical Summary**

As can be seen from the brief reviews of the state of the art, the electronic health record is feasible and accessible for all types of healthcare research. Non-communicable disease prediction, such as heart disease, lung cancer and COVID-19 prediction, has been extensively researched, both in a larger machine learning context, including ensemble models and automated models, and specifically for neural networks (deep learning). Other difficulties should be examined further because, at the moment, long COVID has only been evaluated until the analytical step but not much in machine learning. All of the studies that have been considered that use machine learning to predict long COVID are interesting information sources for future developments, in my opinion. Long COVID predictions, as well as other disease predictions, are currently a difficulty for machine learning because of the need for results to be reliable, as well as the time limitations required to account for the accuracy of predictions. As a result, while ensemble models and AutoML have been employed in a number of healthcare research to aid in disease prediction, they have not been utilised to predict long COVID. Moreover, many of the used medical-related datasets came from the general platforms such as Kaggle, UCI Machine Learning Repository like the pre-processed datasets from MedMNIST, this is not suited for a disease prediction in real settings. Moreover, relating back to the long COVID, there is a lack of available long COVID datasets, health professionals' datasets for analysis are primarily self-collected from the local hospitals and not for publication. For instance, in the article of [(Knight, et al., 2022](#Knight)), a survey study with an assessment of the electronic health record (EHR) was conducted at Mayo Clinic in Jacksonville, Florida. Indeed, a survey conducted by health professionals in real hospitals could be more appropriate for the application. In this regard, real-world datasets will play a critical role in increasing long COVID prediction performance when compared to tidy datasets, and ML algorithms and architectures will need to be created to deal with heterogeneous data. The research conducted by [(Hooda, Gupta, & Gupta, 2020](#Hooda)) proved their proposed ensemble model which was Bagging-based Random Forest and Adaboost can achieve a ROC score of 0.98 on the real-world surveyed data. Furthermore, AutoML has been examined as a means of reducing data scientists' workloads, thus the optimal settings for an ensemble model can be identified using AutoML. However, data used to train and test long COVID prediction should be given extra consideration: the published and available long COVID datasets contain a number of critical and problematic factors that could render data collected in controlled conditions or from varied contexts unusable.

* 1. **Conclusion**

In conclusion, the current long COVID variables correlations analysis, ensemble models and AutoML libraries implementations in the health field are not limited to the sections discussed above; more research has been done that yields useful results. These previous works have given rise to new ideas and gaps that need to be filled.

**CHAPTER 3**

**METHODOLOGY**

* 1. **Introduction**

This chapter discusses the methodology performed and utilised in this project from data collection to model evaluation. The primary goal of this project is to automate as many stages as possible because AutoML can help to reduce time consumption and is suitable for applying to the tabular dataset.

There are ten sections in this chapter. The first section provides an overview of the chapter. The implementation environment is mentioned in the next section. The sources of the chosen data are explained in the following section. Section four explains the data pre-processing procedure using Pandas. Section five outlines the feature engineering process, including feature selection, feature processing, etc. The model building and selection will be explained in section six. The next section is about the data splitting. The model evaluation metrics applied are stated and shown in section eight. The chart of the flow of activities is presented in the following section of this chapter. The final chapter is the short conclusion of this chapter.

* 1. **Implementation Environment**

The laptop used to run the experiments of this project consists of Intel(R) Core(TM) i5-8265U CPU and 8GB RAM. For implementation, Google Colab is connected to the local runtime, the local operating system (OS) has divided into two: Windows 10 and Ubuntu 18.04 due to the Auto-sklearn must only run on the Linux system. Therefore, the version of applied libraries such as sklearn might be different in the two Colab books according to the running system as shown in Table 4. The applied libraries include Pandas, Numpy, Scikit-learn and Featuretools for data pre-processing and feature processing while TPOT and Auto-sklearn for model building and generation. Sklearn is then applied again to do the evaluations on the models, lastly the Matplotlib and Seaborn are for visualisation of the important features and evaluation results.

Table 4: Example of Different Versions Used in Two OS

|  |  |  |
| --- | --- | --- |
|  | **Windows 10** | **Ubuntu 18.04** |
| **Python** | Python 3.9.7 | Python 3.6.9 |
| **Pandas** | 1.3.4 | 1.1.5 |
| **Sklearn** | 1.0.2 | 0.24.2 |
| **Featuretools** | 1.6.0 | 0.23.3 |

* 1. **Data Collection**

There are 3 sets of data that have been retrieved. First, the "Novel Coronavirus (COVID) Illness – Patient Report (NCIPR)" (Thomason, Werchan, & Hendrix, 2021). The contributors of the NCIPR dataset have provided a file that outlines the survey's processing and filtering steps after it has been gathered. All of the features have been discussed inside the file, and the authors have suggested potential predictors for additional and further research. The second and third datasets are “kenya-long-covid-effects” and “malawi-long-covid“, both datasets were collected by the Wellcome team (Wellcome, 2022). The contributor of the Kenya and Malawi dataset provided a lookup file so that others might get a sense of what they collected.

* 1. **Data Pre-processing**

All the datasets will be pre-processed with the Pandas package, which is included with Python. The NCIPR dataset has 2215 rows and 204 columns in its original form. Following the survey's authors' instructions, the dataset is filtered by only selecting rows with the column 'complete binary' equal to 1.0. The dataset is then filtered once more by removing rows with the field "long symptoms' equal to N/A or missing values. Thus, the final number of rows of the NCIPR dataset that will be applied is 1670. Originally, the Kenya dataset has 806 rows and 1906 columns whereas the Malawi dataset has 885 rows and 1710 columns. They will be combined into one file named ‘Kenya&Malawi’ after feature selection and the final number of rows after the combination is 1665. All the datasets are conducted in the survey form and the contributors convert them into a tabular format. All the datasets are ensured to contain the common features, for example, the fundamental information such as the individuals' demographics, such as their age and medical history, as well as the self-report, which includes secondary symptoms, vaccination status, and so on.

The target column will be extracted and assigned as a new variable for convenient operation. Besides, some of the column input classes will be converted into appropriate classes, such as the date of vaccination. The original class for this column is 'string', thus it will be converted to class 'Date' for ease of use. Furthermore, missing data will be filtered out or filled with mean values, as will noisy data such as incorrect input format because the original input should only be known by dataset contributors, it is not permitted to make changes to the data entered. However, because all columns have missing values, some of the features will be determined to fill with mean values, if not, all rows will be dropped out following the Pandas process. The Kenya&Malawi data has over 1000 columns; however, just the necessary columns will be considered, and similar features will be derived, based on the NCIPR dataset. The columns from the Kenya&Malawi dataset are renamed, and some of the features' values are mapped to match the values in the NCIPR dataset. The columns from the Kenya&Malawi dataset are renamed, and some of the features' values are mapped to match the values in the NCIPR dataset. In the NCIPR dataset, for example, the level of disruption is divided into five categories: "No disruption," "Very light disruption," "Mild to moderate disruption," "Moderate to severe disruption," and "Extreme disruption." Thus, the disruptive level of the Kenya&Malawi dataset which is classified as "Never," "Rarely," "Sometimes," "Always," and "Very often." will be mapped to the values of the NCIPR dataset's disruptive level. After that, the Kenya dataset and Malawi dataset will be combined to form Kenya&Malawi dataset as they are two separate excel files originally.

* 1. **Feature Engineering**

Based on the NCIPR original report, the authors suggested some features to consider such as the stress level of patients, thus 36 features out of 204 features are selected. Furthermore, the Kenya&Malawi dataset feature selection is based on similar features discovered in the NCIPR dataset. The description of each feature of the NCIPR dataset and the Kenya&Malawi dataset is presented in Table 5 and Table 6.

Table 5: The Description of Each Selected Feature from the NCIPR dataset

|  |  |
| --- | --- |
| **Feature** | **Description** |
| age\_calculated | The age that calculated by the report’s authors using the date of birth. |
| ncipr\_gender | Gender of the patient. |
| ncipr\_onset\_date | The first day of tested positive for COVID-19. |
| ncipr\_symptoms\_all | The symptoms experienced by the patient along with the COVID-19 recovery phase. |
| ncipr\_symptoms\_med\_complicat | The medical complications experienced by the patient along with the COVID-19 recovery phase. |
| ncipr\_max\_temp | The maximum recorded body temperature. |
| ncipr\_min\_osat | The lowest recorded oxygen saturation. |
| ncipr\_symptom\_worst | The most serious COVID symptom or medical complication that the patient experienced. |
| ncipr\_symptoms\_events | The events experienced by the patient during the emergence of symptoms. |
| ncipr\_how\_severe\_self | The level of severity of patients with COVID-19. |
| ncipr\_onset\_symptoms | The initially emerged symptoms. |
| ncipr\_employ | The employment status of the patient when developed COVID-19. |
| ncipr\_treat\_home | The home treatments used by the patient. |
| ncipr\_treat\_hospital\_days | The duration of hospitalisation. |
| ncipr\_treat\_med\_therapy | The medical treatments received by the patient during the hospitalisation period. |
| ncipr\_treat\_prescriptions | The medications prescribed to treat COVID-19. |
| ncipr\_lasting\_changes | The lasting symptoms after recovering from COVID-19 and this is the prediction target. |
| ncipr\_lasting\_cognitive | The memory problems experienced by the patients. |
| ncipr\_return\_health\_when | The expected time which patients think will be to recover fully from COVID-19. |
| ncipr\_lasting\_changes\_time | The length of time that patients experienced secondary symptoms. |
| ncipr\_how\_anxious | The level of anxiety that patients experienced when being ill with COVID-19. |
| ncipr\_how\_disrupted | The level of disruption that patients experienced when being ill with COVID-19. |
| ncipr\_quarantine\_kind\_test | The quarantine behaviour when ill with COVID-19. |
| ten\_days\_activity\_01 | The activities were carried out by the patients in the last 10 days. |
| ncipr\_pet | Whether the patient has pets. |
| ncipr\_vaccine\_received | Whether the patient has received the vaccination. |
| ncipr\_vaccine\_received\_date | The date when receiving the first dose of vaccination. |
| ncipr\_vaccine\_had\_side | The side effects when receiving the first dose of vaccination. |
| ncipr\_vaccine\_side\_effects | The types of side effects when receiving the first dose of vaccination. |
| ncipr\_vaccine2\_had\_side | The side effects when receiving the second dose of vaccination. |
| ncipr\_vaccine2\_side\_effects | The types of side effects when receiving the second dose of vaccination. |
| ncipr\_vaccine\_relaxed\_behavior | Has the patient relaxed their COVID-19 safety behaviours after having received the COVID-19 vaccine? |
| db\_24 | The stress level of patients during the COVID-19 outbreak. |
| db\_26 | The energy level of patients during the COVID-19 outbreak. |
| db\_36 | The medical history of the patient. |
| db\_37 | The past medical treatment received by the patient. |

Table 6: The Description of Each Selected Feature from the Kenya&Malawi dataset

|  |  |
| --- | --- |
| **Feature** | **Description** |
| age | The age group which categorised by the authors of the dataset. |
| gender | Gender of the patient. |
| date | \*\*\*only found in the Malawi dataset  The first day of tested positive for COVID-19. |
| Symptoms | The symptoms experienced by the patients when they became unwell. |
| C6 | The long-term Covid-19 symptoms that patients found it difficult to get relief and manage. |
| occupation | The employment status of the patient when developed COVID-19. |
| C10 | The home or self-care treatments that are taken by patients when unwell help them manage their symptoms. |
| C7 | The medical care that patients seek. |
| C4 | The physical lasting symptoms and this is the prediction target. |
| C4b | The mental lasting symptoms and this is the prediction target. |
| C4b\_5 | The memory problems experienced by the patients. |
| C3.1\_1 to C3b.3\_8 | The level of disruption that patients experienced when being ill with COVID-19. |
| C3b.1\_1 | The level of anxiety that patients experienced when being ill with COVID-19. |
| C3b.2\_1 |
| C3b.3\_1 |
| C3b.1\_7 | The stress level of patients during the COVID-19 outbreak. |
| C3b.2\_7 |
| C3b.3\_7 |
| C3.1\_1 | The energy level of patients during the COVID-19 outbreak. |
| C3.2\_1 |
| C3.3\_1 |
| C22 | Whether the patient has received the vaccination. |
| communicable\_disease | The medical history of the patient. |
| non\_communicable\_disease |
| body\_organ\_disease |
| neurological\_condition |
| respiratory\_condition |
| other\_disease |

Featuretools, an automated feature engineering package, will be used to complete the following operation. The most relevant factors for forecasting the occurrence of long COVID will be identified by a ranking of the 36 attributes. The automatic feature selection tool aids in the reduction of implementation time and the delivery of interpretable features with real-world relevance. Furthermore, it prevents incorrect data usage, which can easily occur when data is handled manually. Moreover, due to both Auto-sklearn and TPOT being restricted to numerical format, thus all the non-numeric values will be converted into a numerical format using the built-in function of Featuretools to encode the variables.

Tables 4 and 5 highlight the target variable to be predicted in beige: the lasting effect on COVID-19 positive patients. The columns name in the dataset used will be changed to a common name which is ‘long\_symptoms’. At first, the dependent variable is a multiclass variable, such as a sore throat, weariness, or shortness of breath, among other symptoms. However, the variable is transformed to binary class with No (0) for the 'None of these' and Yes (1) for all other answers, regardless of the type of symptoms. The goal of this project is to forecast if a patient will have long COVID symptoms using the predictors that have been chosen. Figure 3 shows the transformation of long COVID symptoms.

A screenshot of a computer

Description automatically generated with medium confidence

Figure 3: Transformation of Multi-class variable to Binary Class Variable

* 1. **Model Building**

There are some experiments involve NCIPR dataset only and there are involve all the datasets that have been introduced in previous section (Section 3.3). The main dataset for training and validation is the NCIPR dataset while the Kenya&Malawi dataset is for testing purposes. The AutoML libraries will be utilised, as indicated in the previous section. Auto-sklearn and TPOT were chosen as the two packages to implement. The built-in operation of the Auto-sklearn and TPOT is roughly described and shown below. In this project, the comparisons are mainly focus on the differences of single ML model and ensemble models, the AutoML libraries chosen are for reducing the tuning time when suggesting the optimal single classifier and ensemble models. For instance, the question of does single Random Forest is better than ensemble of multiple Random Forest should be answered after experiments done in Chapter 4.

* + 1. **Auto-sklearn**

Diagram

Description automatically generated

Figure 4: Auto-sklearn Framework [(Feurer, et al., 2015)](#Feurer)

Auto-sklearn’s features include the ability to kick-start the Bayesian optimisation process and hence increase efficiency. After that, it will determine and store an ML pipeline with excellent empirical performance for datasets using Bayesian optimisation. Parallelly, the ensemble model will be generated and the results will be stored with the other single ML models.

During the process of optimising the ML framework, the meta-learning technique is complementary to Bayesian optimisation. Although meta-learning has the advantage of searching for and suggesting the instantiation of an ML model that is likely to perform well, it also has the disadvantage of not being able to provide fine-grained performance information, which causes users to be unable to understand and interpret the model generated. The Bayesian optimisation approach, on the other hand, will search hyperparameter spaces as vast as those of complete ML frameworks while also fine-tuning performance over time. A set of meta-features derived from a dataset is evaluated in total. Then, using Bayesian optimisation, an instantiation of the supplied dataset will be determined and saved. As a result, the meta-features are computed for each dataset, and the datasets are ranked in the meta-feature space by the distance. Before proceeding to Bayesian optimisation, the k closest datasets will be picked and saved in the ML framework instantiations for evaluation. Because building a weighted ensemble of the models does not perform well in Auto-sklearn, the ensemble selection is chosen and applied in Auto-sklearn. (Caruana et al., 2004) presented ensemble selection, which is a greedy strategy that starts with an empty ensemble and adds the model that minimises the validation loss iteratively. Figure 5 depicts the ensemble selection techniques in rough.

A picture containing graphical user interface

Description automatically generated

Figure 5: Ensemble Selection Applied in Auto-sklearn

From Figure 4, it is shown that the Bayesian optimiser includes the data pre-processor, feature pre-processor and classifier. The data pre-processor includes rescaling, encoding, libraries and balancing processes. While the feature pre-processor includes the algorithms that can be found in sklearn such as Select percentile, Principal component analysis (PCA) and more. The type of classifier includes AdaBoost, Decision Tree and other classical ML classifiers. The scoring metrics might be a single metric or a list of metrics; in this case, the parameter is passed a list of accuracy, recall, and precision.

* + 1. **TPOT**

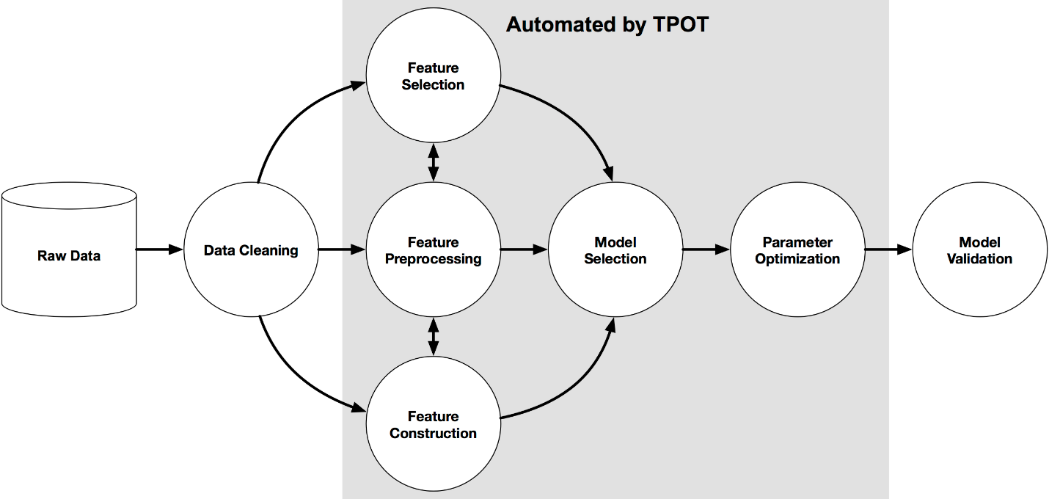


Figure 6: Example Machine Learning Pipeline and the Parts Automated by [TPOT (Le, Fu, & Moore, 2020)](#le)

A screenshot of a video game

Description automatically generated with medium confidence

Figure 7: GP Algorithm Applied in TPOT

TPOT, like Auto-sklearn, has a built-in feature processor and supervised classifiers, but it lacks data pre-processing methods. As demonstrated in Figure 7, TPOT used Genetic Programming (GP) to automate and optimise the tree-based pipelines. TPOT will automatically split the dataset into 75 per cent and 25 per cent for training and validation. After the hundred random tree-based pipelines have been generated, the top 20 pipelines will be chosen based on the default score metric, which is accuracy. In this project, the accuracy metric will be used to choose the optimum pipeline. After that, each pipeline will produce five offspring, of which 5% will use the one-point crossover to cross over with each other and the remainder will be modified at random by a point, insert, or shrink mutation. The approach maintains a Pareto front of non-dominated solutions uncovered at any moment during each generation of the GP cycle.

Despite both Auto-sklearn and TPOT offering data pre-processing and feature pre-processing functions, Pandas, the two encoder tools and Featuretools will still be utilised to reduce the dimensionality of datasets before developing a model. Therefore, as described Auto-sklearn and TPOT have built-in functions for this stage, there will be no manual hyperparameter adjustment in this section. However, the optimised hyperparameters will be printed out and displayed for a better understanding of a model. For example, if Random Forest is chosen as the best model, the settings of parameters such as minimum split can be displayed using the built-in functions of both TPOT and Auto-sklearn.

* 1. **Data Splitting**

For the baseline experiments, which are experiment 1 to 4, the NCIPR dataset with the selected 36 features will be used. The dataset is split into 60% for training and 40% for validation using the traintestsplit() method in the sklearn library. Besides, the 60% train set will be split into train set and validation set automatically when send into AutoML libraries.

The following experiments after baseline, the Kenya&Malawi dataset will be implemented to fulfil the objective 1 which is to curate and modify datasets to investigate the versatility of models generated. In other words, these procedures are designed to see if the classifiers that have been created can be used for a variety of datasets that undergo feature reduction to achieve this project objectives 2 and 3. In those experiment, the number of features which is the 36 features in NCIPR dataset will be reduced to match the features that also coexist in the Kenya&Malawi dataset. The process of data splitting is depicted in [Figure 8](#fig8) and [9](#fig9).

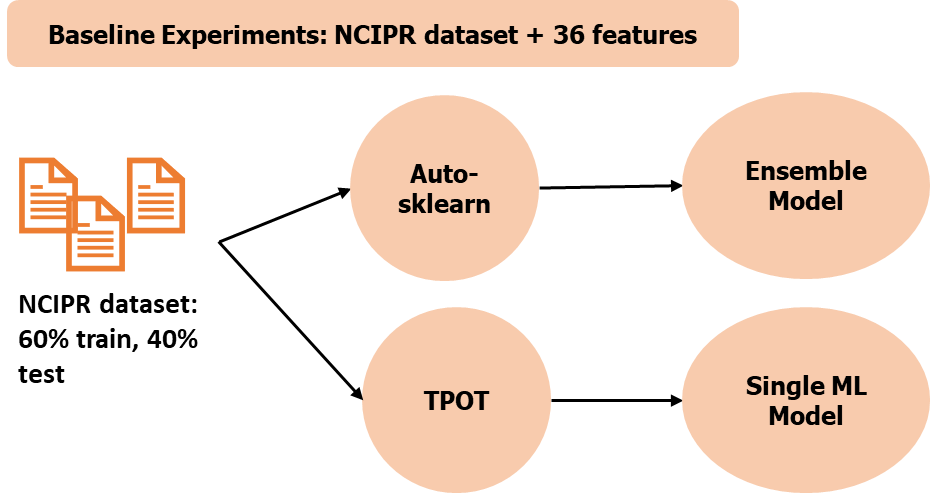


Figure 8: Process of Data Splitting for Baseline Experiments

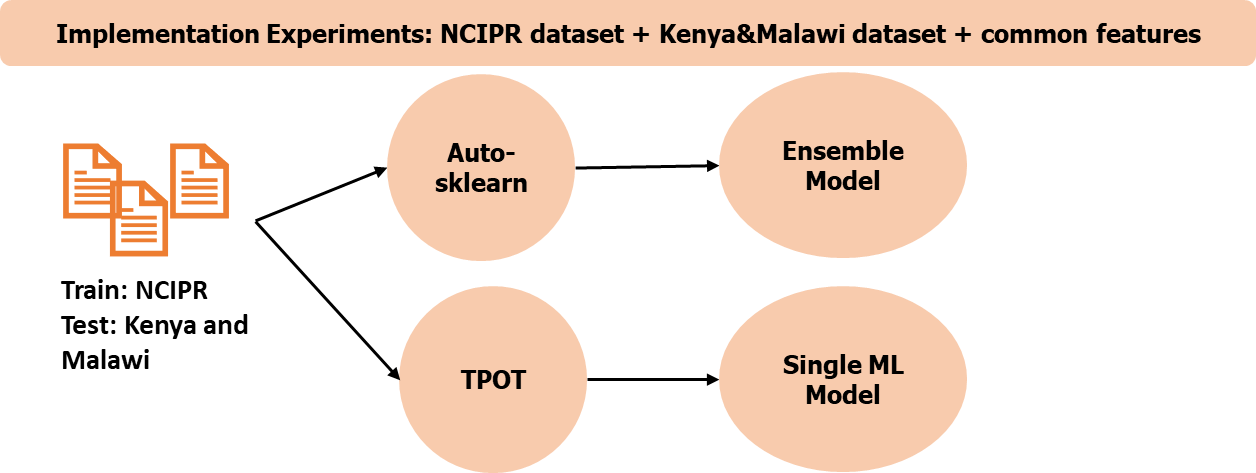


Figure 9: Process of Data Splitting for Experiments with Feature Reduction

* + 1. **Synthetic Data**

A model that has been developed or trained to reproduce real data (RD) based on its distributions and structure produces synthetic data (SD), which is artificial data (Hernandez, et al, 2022). The goals of this concept are to gather and receive a vast amount of data because model training need it. However, data collecting is becoming more difficult nowadays because of the need to preserve data privacy, particularly in the healthcare sector, and the high expense of gathering real-world data. Therefore, the synthetic data can be used in place of the real-world data by creating it using AI or statistical methods depending on the features of the real-world data provided.

In two of the experiments in this study, a synthetic data is generated based on the real-world data which are Kenya and Malawi datasets. The total number of instances is 1691, so the number of instances of synthetic data will be half of the original dataset. A few benefits of this method to solve imbalance problem include the preservation of data quality and the fact that synthetic data are unaffected by resampling techniques, which might alter the distribution or genuine patterns of data. Additionally, because relevant data is hard to come by and more scalable, synthetic data fills in the blanks and supports real-world data to produce a greater scale of inputs. DataSynthesizer which is a synthetic data generation library in Python is utilised. Figure 10 displays the process of data splitting with synthetic data in this project.



Figure 10: Process of Data Splitting for Experiments with Synthetic Data Generation

* 1. **Model Evaluation**

The models will be compared and evaluated by performance metrics including accuracy, precision and sensitivity scores which can be imported from the sklearn library. Because all of the datasets utilised in this study are imbalanced, accuracy alone cannot yield a respectable result, the three metrics were chosen and the parameter ‘average’ of precision and recall is set to ‘weighted’ to indicate the input data is imbalance. Consider the confusion matrix given in [Figure 11](#fig10), and the three performance metrics equations shown in Table 8; however, the confusion matrix which follows the standard of sklearn is not part of the experiment and is just shown here for clarity.

Timeline

Description automatically generated

Figure 11: Confusion Matrix of Long COVID Prediction (Scikit-learn version)

Table 8: Description of Calculations Accuracy, Precision and Recall in Confusion Matrix

|  |  |  |
| --- | --- | --- |
| **Evaluation metric** | **Formula** | **Description** |
| Accuracy |  | The ratio of correct predictions done which prediction is long COVID and actual also long COVID. Also, prediction is non-long COVID and actual is non-long COVID. The prediction is matched the actual class. |
| Precision |  | The ratio of correct positive predictions done which prediction is the long COVID and actual is long COVID. High precision means the patient with long COVID predicted correctly to the class long COVID. So, the probability of predicting the patient to the wrong group is low (Type 1 error is low). |
| Recall |  | The ratio of correct positive predictions done which prediction is long COVID and actual is long COVID. A high recall score means the patient with long COVID predicted correctly to the class long COVID. So, the probability of predicting the patient to the wrong group is low (Type 2 error). The recall score is important because if the patients have long COVID but the prediction is gone to do not have long COVID, this probably will cause the patients to miss the golden time to rescue from long COVID. |

* 1. **Overall Flow of Activities**

Diagram

Description automatically generated

Figure 12: Overall Flow of Activities

* 1. **Conclusion**

In summary, data gathering, data pre-processing, feature engineering, model development, and model evaluation are all part of the whole process. Three similar datasets were collected by various organisations as part of the data collection. They're available on the corresponding web pages. Second, certain basic data pre-processing operations such as renaming columns, eliminating missing data rows, converting data types, and so on are performed. The automated tool Featuretools is then used to handle the feature engineering process. Then, for model building, Auto-sklearn and TPOT are used to generate their best models. Finally, the accuracy, precision, and recall of the models will be assessed.

**CHAPTER 4**

**EXPERIMENTAL DESIGN**

* 1. **Introduction**

This chapter will cover the design of the experiments and how they will be carried out throughout the project. The experiments are designed to solve or accomplish the four objectives outlined in [Section 1.4](#prjobj). The first section covers the chapter's introduction, while the second section covers data curation. The baseline models are described in the following section, and the experiments with parameter changes are described in the fourth section. Furthermore, the experiments involve feature reduction is explained in fifth section. The sixth and seventh sections describe the experiments with new imported test set with the later section includes parameter modifying. The subsequent two sections are about the experiment with applying an oversampling technique on test set and synthetic data generation. Lastly, there is a brief conclusion concludes this chapter.

* 1. **Data Curation**

The NCIPR dataset is imported using Pandas and 35 features from a total of 204 are chosen based on past statistical research and the data authors' suggestions. [Table 5](#table5) in Chapter 3 lists the attributes that were chosen. After that, some of the features will have their data types changed, such as from string to datetime format. [Figure 13](#fig12) depicts the code segment for transforming data types.

Figure 13: Code Fragment of Transforming Data Type

ncipr\_data['age'] = ncipr\_data['age'].str.replace('[+]', '')

ncipr\_data['age'] = pd.to\_numeric(ncipr\_data['age'], errors='ignore') #ignore NaN

ncipr\_data['onset\_date'] = pd.to\_datetime(ncipr\_data['onset\_date'], format='%y-%b', errors='coerce')

* 1. **Baseline Model**

Using the NCIPR dataset, two baseline experiments will be designed for this study. In a 6:4 ratio, the NCIPR dataset will be split into training and testing data. The first experiment uses the AutoML library, TPOT to create a single classifier to predict the occurrence of long COVID. All the parameter are set to default except the 'generations' 'population\_size', ‘scoring’, ‘max\_eval\_time\_mins’ and ‘verbosity’. The 'generations' refers to the number of iterations in the run pipeline optimisation process, while the 'population\_size' refers to the number of individuals to keep in each generation's genetic programming population. Both settings should be set to their maximum values to achieve the best outcomes. However, due to the inevitable reason which the laptop used to execute this experiment can't support the default 'generations' and 'population\_size' of 100, the 'generations' and 'population\_size' are set to 5 and 50, respectively. Besides, the ‘max\_eval\_time\_mins’ is set to 3 (in minutes) to shorten the optimisation running time. The ‘verbosity’ is set to 2 to print the progress bar and some information about the optimisation process. The approach of creating metrics by counting the total true positives, false negatives, and false positives, hence the scoring metric is set to 'f1-micro.' The cross-validation approach used while optimising pipelines is StratifiedKFold, and the default 'cv' is 5. [Figure 14](#fig13) shows the code fragment of running AutoML to produce single classifier.

Figure 14: Code Fragment of AutoML Single Classifier (TPOT)

def tpot\_clf(X\_train, y\_train, gen, X\_test, y\_test):

tpot = TPOTClassifier(generations=5, population\_size=50, verbosity=2, scoring='f1\_micro', max\_eval\_time\_mins=3, random\_state=10)

tpot.fit(X\_train, y\_train)

tpot.export('tpot\_pipeline.py')

The second experiment is to create an ensemble model using AutoML library, AutoSklearn, however the ensemble model is the major focus here. All the parameter are set to default except the 'time\_left\_for\_this\_task', ’ensemble\_size', ‘resampling\_strategy’, ‘resampling\_strategy\_arguments’, ‘scoring\_functions’ and ‘metric’. The 'time\_left\_for\_this\_task' refers to the time limit in seconds for the search for appropriate models. By increasing this value, the automated ensemble ML library has a higher chance of finding better models. The ‘ensemble\_size' refers to the number of models added to the ensemble built by Ensemble selection from libraries of models. Both settings should be set to their default values to achieve the best outcomes. However, due to the reason stated in the previous paragraph, the default 'time\_left\_for\_this\_task' and ‘ensemble\_size' of 3600 and 50 respectively, the 'time\_left\_for\_this\_task' and ‘ensemble\_size' are set to 600 and 3, respectively. Besides, the ‘resampling\_strategy’ and ‘resampling\_strategy\_arguments’ are set to ‘cv’ with 5 folds. The ‘scoring\_functions’ and ‘metric’ are set to f1-micro calculating each pipeline and results. [Figure 15](#fig14) shows the code fragment of running AutoML to produce ensemble model.

def as\_clf(X\_train, y\_train, e\_size, X\_test, y\_test):

clf = AutoSklearnClassifier(time\_left\_for\_this\_task=600, ensemble\_size = e\_size,

scoring\_functions=[f1\_micro], resampling\_strategy='cv', resampling\_strategy\_arguments={'folds': 5},

metric=autosklearn.metrics.f1\_micro)

clf.fit(X = X\_train, y = y\_train)

Figure 15: Code Fragment of Ensemble ML (Auto-sklearn)

Under similar parameter settings, such as resampling technique, the results from the two models will be compared to see which model can perform better. The two baseline experiments are summarised in [Table 9](#table9). Furthermore, the two experiments contribute to the project's objectives 2, 3, and 4.

Table 9: Summary of Baseline Experiments

|  |  |  |
| --- | --- | --- |
| **Experiment** | **Purposes** | **Output** |
| Experiment 1 | 1. Baseline. 2. To implement machine learning classifiers to predict the occurrence of long COVID. 3. To evaluate the accuracy, precision, and sensitivity of machine learning classifier for predicting the presence of long COVID. | Single Classifier |
| Experiment 2 | 1. Baseline. 2. To implement ensemble model to predict the occurrence of long COVID. 3. To evaluate the accuracy, precision, and sensitivity of ensemble model for predicting the presence of long COVID. | Ensemble Model |

* 1. **Experiment with Manipulated Parameter**

The next two experiments are developed using the same NCIPR dataset. The third experiment will modify the parameter 'generations' to 3, 7, and 10 as shown in [Figure 16](#fig15), while the other parameters remain the same as the first experiment. This experiment will be utilised to accomplish objectives 3 and 4 by determining whether the optimisation learning process and the performances of generated single classifier can be improved as the number of generations grows.

generations = [3, 7, 10]

for g in generations:

p, r, a = tpot\_clf(X\_train\_ncipr, y\_train\_ncipr, g, X\_test\_ncipr, y\_test\_ncipr)

Figure 16: Code Fragment of Manipulate parameter ‘generations’ in AutoML Single Classifier (TPOT)

The fourth experiment will modify the parameter 'ensemble\_size' to 5, 7, and 10 as shown in [Figure 17](#fig16), while the other parameters remain the same as the second experiment. This experiment is to test whether increasing the number of 'ensemble\_size' will improve the metric scores and this experiment will be used to satisfy objectives 3 and 4. The two manipulated experiments are summarised in [Table 10](#table10).

en\_size = [5, 7, 10]

for e in en\_size:

as\_clf(X\_train\_ncipr, y\_train\_ncipr, e, X\_test\_ncipr, y\_test\_ncipr)

Figure 17: Code Fragment of Manipulate parameter ‘ensemble\_size’ in Ensemble ML (Auto-sklearn)

Table 10: Summary of Experiments with Manipulated Parameter Settings

|  |  |  |
| --- | --- | --- |
| **Experiment** | **Purposes** | **Outputs** |
| Experiment 3 | 1. To implement machine learning classifiers to predict the occurrence of long COVID. 2. To evaluate the accuracy, precision, and sensitivity of machine learning classifier for predicting the presence of long COVID. | Each loop produced a single classifier |
| Experiment 4 | 1. To implement ensemble model to predict the occurrence of long COVID. 2. To evaluate the accuracy, precision, and sensitivity of ensemble model for predicting the presence of long COVID. | Each loop produced an ensemble model |

* 1. **Experiment with Feature Reduction**

The two experiments that follow were created utilising the NCIPR dataset. In the fifth and sixth experiments, some of the features shown in Table 5 which can be found in Section 3.5, will be manually removed. Due to the fact that the Kenya and Malawi datasets lack some of the features found in the NCIPR dataset and the comparison can be found in Table 6, also in Section 3.5, this step is necessary to match the other two datasets. The experiments designs and the parameter settings are same as the baseline models for both AutoML single classifier and ensemble model. By investigating whether less features will improve or worsen the performance scores of the models, these experiments will be used to achieve objectives 1, 2, 3, and 4. The features that will be used in the ensuing experiments are shown in Table 11 below after some of the features from Table 5 have been removed. Also, Table 12 displays the summaries of the objectives and the outputs of experiments.

Table 11: The Description of Features from the NCIPR dataset after Feature Reduction

|  |  |
| --- | --- |
| **Feature** | **Description** |
| age\_calculated | The age that calculated by the report’s authors using the date of birth. |
| ncipr\_gender | Gender of the patient. |
| ncipr\_onset\_date | The first day of tested positive for COVID-19. |
| ncipr\_symptoms\_all | The symptoms experienced by the patient along with the COVID-19 recovery phase. |
| ncipr\_symptom\_worst | The most serious COVID symptom or medical complication that the patient experienced. |
| ncipr\_employ | The employment status of the patient when developed COVID-19. |
| ncipr\_treat\_hospital | The duration of hospitalization. |
| ncipr\_treat\_prescriptions | The medications prescribed to treat COVID-19. |
| ncipr\_lasting\_changes | The lasting symptoms after recovering from COVID-19 and this is the prediction target. |
| ncipr\_lasting\_cognitive | The memory problems experienced by the patients. |
| ncipr\_how\_anxious | The level of anxiety that patients experienced when being ill with COVID-19. |
| ncipr\_how\_disrupted | The level of disruption that patients experienced when being ill with COVID-19. |
| ncipr\_vaccine\_received | Whether the patient has received the vaccination. |
| db\_24 | The stress level of patients during the COVID-19 outbreak. |
| db\_26 | The energy level of patients during the COVID-19 outbreak. |
| db\_36 | The medical history of the patient. |

Table 12: Summary of Experiments with Feature Reduction

|  |  |  |
| --- | --- | --- |
| **Experiment** | **Purposes** | **Outputs** |
| Experiment 5 | 1. To curate and modify datasets from the existing long COVID datasets based on the common features in order to identify the most relevant features that develop long COVID by applying data processing tools and automated feature engineering tools. 2. To implement machine learning classifiers to predict the occurrence of long COVID. 3. To evaluate the accuracy, precision, and sensitivity of machine learning classifier for predicting the presence of long COVID. | Single classifier |
| Experiment 6 | 1. To curate and modify datasets from the existing long COVID datasets based on the common features. 2. To implement ensemble model to predict the occurrence of long COVID. 3. To evaluate the accuracy, precision, and sensitivity of ensemble model for predicting the presence of long COVID. | Ensemble model |

* 1. **Experiment with** **New Imported Test Set**

The NCIPR, Kenya, and Malawi datasets were used to develop the next two studies. The three datasets are altered by selecting and merging them based on the common features in the seventh and eighth experiment. Table 11 of the NCIPR dataset and Table 6 of the datasets for Kenya and Malawi contain descriptions of each feature. The baseline models, Experiments 1 and 2, serve as the basis for the experiment designs and parameter values. Additionally, Kenya and Malawi are used for testing whereas NCIPR is used for training. The datasets for Kenya and Malawi can serve as the test sets because they were obtained from a different sources from NCIPR and are thus completely unseen by the model developed. These experiments will be utilised to accomplish goals 1, 2, 3, and 4 by examining the generalisation of models created.

As indicated in Table 13, all the columns from the NCIPR, Kenya, and Malawi dataset have been renamed, and some of the feature values have been mapped to guarantee that they match, the highlighted row is the target. The disruptive level of the Kenya and Malawi datasets, which is categorised as "Never," "Rarely," "Sometimes," "Always," and "Very often." will be mapped to the values of the NCIPR dataset's disruptive level. For example, the level of disruption in the NCIPR dataset is divided into five categories: "No disruption," "Very light disruption," "Mild to moderate disruption," "Moderate to severe disruption," and "Extreme disruption." In Figure 18, the mapping code is displayed. Then, since they were initially two independent excel files, the Kenya dataset and the Malawi dataset will be joined. The summary of two experiments is shown in Table 14.

disrupt\_dict = {'Never':'No disruption', 'Rarely':'Very mild disruption', 'Sometimes':'Mild to moderate disruption', 'Always':'Moderate to severe disruption', 'Very often':'Extreme disruption'}

kenya\_data['how\_disrupted'] = kenya\_data['how\_disrupted'].map(disrupt\_dict)

Figure 18: Code Fragment of Mapping Values

Table 13: The Renamed Columns in NCIPR, Kenya and Malawi Datasets

|  |  |  |
| --- | --- | --- |
| **NCIPR** | **Kenya and Malawi** | **Renamed Column** |
| age\_calculated | age | age |
| ncipr\_gender | gender | gender |
| ncipr\_onset\_date | date | Onset\_date |
| ncipr\_symptoms\_all | Symptoms | symptoms |
| ncipr\_symptom\_worst | C6 | Symptom\_worst |
| ncipr\_employ | occupation | Employ\_status |
| ncipr\_treat\_hospital | C7 | treat\_hospital |
| ncipr\_treat\_prescriptions | C10 | treat\_prescriptions |
| ncipr\_lasting\_changes | C4, C4b | long\_symptoms |
| ncipr\_lasting\_cognitive | C4b\_5 | lasting\_cognitive |
| ncipr\_how\_anxious | C3.1\_1, C3.2\_1, C3.3\_1 | how\_anxious |
| ncipr\_how\_disrupted | C3.1\_1 to C3b.3\_8 | how\_disrupted |
| ncipr\_vaccine\_received | C22 | vaccine\_received |
| db\_24 | C3b.1\_1, C3b.2\_1, C3b.3\_1 | stress\_lvl |
| db\_26 | C3b.1\_7, C3b.2\_7, C3b.3\_7 | energy\_lvl |
| db\_36 | communicable\_disease | med\_history |
| non\_communicable\_disease |
| body\_organ\_disease |
| neurological\_condition |
| respiratory\_condition |
| other\_disease |

Table 14: Summary of Experiments with New Imported Test Set

|  |  |  |
| --- | --- | --- |
| **Experiment** | **Purposes** | **Outputs** |
| Experiment 7 | 1. To curate and modify datasets from the existing long COVID datasets based on the common features. 2. To implement machine learning classifiers to predict the occurrence of long COVID. 3. To evaluate the accuracy, precision, and sensitivity of machine learning classifier for predicting the presence of long COVID. | Single classifier |
| Experiment 8 | 1. To curate and modify datasets from the existing long COVID datasets based on the common features. 2. To implement ensemble model to predict the occurrence of long COVID. 3. To evaluate the accuracy, precision, and sensitivity of ensemble model for predicting the presence of long COVID. | Ensemble model |

* 1. **Experiment with New Imported Test Set and Manipulated Parameter**

The next two experiments are developed using the NCIPR, Kenya, and Malawi datasets like the two previous experiments, Kenya and Malawi datasets are for testing purpose. In these experiment, the parameter 'generations' will be modified to 3, 7, and 10 as shown in Figure 15 but with different test set, while the other parameters remain the same. This experiment will be utilised to accomplish objectives 1, 2, 3 and 4 by determining whether the optimisation learning process and the performances of generated single classifier can be improved as the number of generations grows on a new test set.

The tenth experiment will modify the parameter 'ensemble\_size' to 5, 7, and 10 as shown in Figure 1[7](#fig16) but with different test set, while the other parameters remain the same. This experiment is to test whether increasing the number of 'ensemble\_size' will improve the metric scores when testing on a totally new and unseen test dataset and this experiment will be used to satisfy objectives 1, 2, 3 and 4. The two manipulated experiments are summarised in Table 15.

Table 15: Summary of Experiments with Manipulated Parameter Settings and New Imported Test Set

|  |  |  |
| --- | --- | --- |
| **Experiment** | **Purposes** | **Outputs** |
| Experiment 9 | 1. To curate and modify datasets from the existing long COVID datasets based on the common features. 2. To implement machine learning classifiers to predict the occurrence of long COVID. 3. To evaluate the accuracy, precision, and sensitivity of machine learning classifier for predicting the presence of long COVID. | Each loop produced a single classifier |
| Experiment 10 | 1. To curate and modify datasets from the existing long COVID datasets based on the common features 2. To implement ensemble model to predict the occurrence of long COVID. 3. To evaluate the accuracy, precision, and sensitivity of ensemble model for predicting the presence of long COVID. | Each loop produced an ensemble model |

* 1. **Experiment with** **Oversampling Technique**

The combined datasets from Kenya and Malawi have a significant imbalance, with a class non-long COVID to long COVID ratio of 1686:5. The Synthetic Minority Over-sampling Technique for Nominal (SMOTEN) was employed to address this issue (Chawla et al., 2002). SMOTEN anticipates that the resampled data will only contain categorical features, and the datasets used meet this expectation. Long COVID, a minority group, will have an identical number of instances resampled as non-long COVID. The conditions for two subsequent experiments are identical to those of experiments 7 and 8 respectively. These experiments will be used to achieve objectives 1, 2, 3, and 4 by assessing the generalisation of developed models and investigating whether the use of resampling technique will enhance the outcomes. The summary of two experiments is shown in Table 16.

Table 16: Summary of Experiments with Oversampling Technique

|  |  |  |
| --- | --- | --- |
| **Experiment** | **Purposes** | **Outputs** |
| Experiment 11 | 1. To curate and modify datasets from the existing long COVID datasets based on the common features. 2. To implement machine learning classifiers to predict the occurrence of long COVID. 3. To evaluate the accuracy, precision, and sensitivity of machine learning classifier for predicting the presence of long COVID. | Single classifier |
| Experiment 12 | 1. To curate and modify datasets from the existing long COVID datasets based on the common features. 2. To implement ensemble model to predict the occurrence of long COVID. 3. To evaluate the accuracy, precision, and sensitivity of ensemble model for predicting the presence of long COVID. | Ensemble model |

* 1. **Experiment with Synthetic Data**

The combined datasets of Kenya and Malawi have a significant imbalance, with a class non-long COVID to long COVID ratio of 1686:5. Synthetic data is another method used to solve this issue, by using DataSynthesizer, a synthetic data is produced based on the characteristics and patterns of the Kenya and Malawi dataset, the code fragment is displayed in Figure 19. Therefore, the synthetic data is combined with NCIPR dataset to be applied in training and validation phases while Kenya and Malawi datasets remain as the testing set. The conditions for two subsequent experiments are identical to those of experiments 7 and 8 respectively. These experiments will be used to achieve objectives 1, 2, 3, and 4 by assessing the generalisation of developed models and investigating whether the use of synthetic data will enhance the outcomes. The summary of two experiments is shown in Table 17.

generator = DataGenerator()

generator.generate\_dataset\_in\_independent\_mode(num\_tuples\_to\_generate, description\_file)

generator.save\_synthetic\_data(synthetic\_data)

Figure 19: Code Fragment of Synthetic Data Generation with DataSynthesizer

Table 17: Summary of Experiments with Synthetic Data

|  |  |  |
| --- | --- | --- |
| **Experiment** | **Purposes** | **Outputs** |
| Experiment 13 | 1. To curate and modify datasets from the existing long COVID datasets based on the common features. 2. To implement machine learning classifiers to predict the occurrence of long COVID. 3. To evaluate the accuracy, precision, and sensitivity of machine learning classifier for predicting the presence of long COVID. | Single classifier |
| Experiment 14 | 1. To curate and modify datasets from the existing long COVID datasets based on the common features. 2. To implement ensemble model to predict the occurrence of long COVID. 3. To evaluate the accuracy, precision, and sensitivity of ensemble model for predicting the presence of long COVID. | Ensemble model |

* 1. **Conclusion**

The overall design of the experiments is explained in this chapter. In addition, this chapter explains why the experiments are being conducted and how each experiment might be used to meet the project's objectives.

**CHAPTER 5**

**IMPLEMENTATION**

* 1. **Introduction**

This chapter discusses the implementation and testing performed. The first section of this chapter serves as an introduction, while the second and third sections describe the baseline experiments. The studies that assess the effects of the adjustments on parameter settings are described in the next two parts. The implementation of AutoML libraries with feature deduction is covered in the sixth and seventh sections. The implementation on a new test set is detailed in the two parts that follow. Additionally, the testing procedures with new test set and adjusted parameters are covered in the next two parts. The following section applies the oversampling technique to a new test set and a section for the outputs by implementing synthetic data. The short conclusion is at the last section of this chapter.

* 1. **Experiment 1:** **NCIPR Dataset + AutoML Single Classifier**

The NCIPR dataset's training set is sent to the AutoML library, TPOT and because the parameter 'generations' is set to 5, the pipeline optimisation process will run for 5 iterations, with the optimal pipeline being offered as the output. [Figure 20](#fig17) shows the progress metre as well as the current best scores.

Graphical user interface, text

Description automatically generated

Figure 20: The Progress Bar and Current Best Score

The Random Forest Classifier (RF) is the best pipeline in this case, according to the AutoML and the output is presented in [Figure 21](#fig18). Figure 21 highlights the suggested single classifier from TPOT and also feature selection and pre-processing methods such as MinMaxScaler() to improve performance.

Best pipeline: RandomForestClassifier(MinMaxScaler(Normalizer(input\_matrix, norm=l1)), bootstrap=True, criterion=entropy, max\_features=0.4, min\_samples\_leaf=13, min\_samples\_split=11, n\_estimators=100)

Figure 21: The Output Fragment of AutoML Single Classifier (Experiment 1)

The NCIPR test set is then passed to the pipeline as shown in [Figure 14](#fig13), where the precision, recall, and accuracy scores are then calculated. The precision, recall, and accuracy scores are all 85.56 per cent of the precision score, 85.78 per cent of the recall score, and 85.78 per cent of the accuracy score, respectively as shown in [Figure 22](#fig19).

Precision: 85.56%

Recall: 85.78%

Accuracy: 85.78%

Figure 22: The Metric Scores of AutoML Single Classifier (Experiment 1)

* 1. **Experiment 2: NCIPR Dataset +** **Automated Ensemble ML**

The NCIPR dataset's training and testing sets are sent to the Auto-sklearn, the optimal ensemble model’s details can be printed out as shown in [Figure 15](#fig14). In this case, the automated ensemble ML library indicates that the best pipeline is an ensemble of two Random Forests, each with its unique parameter values, as circled them out in [Figure 23](#fig20).

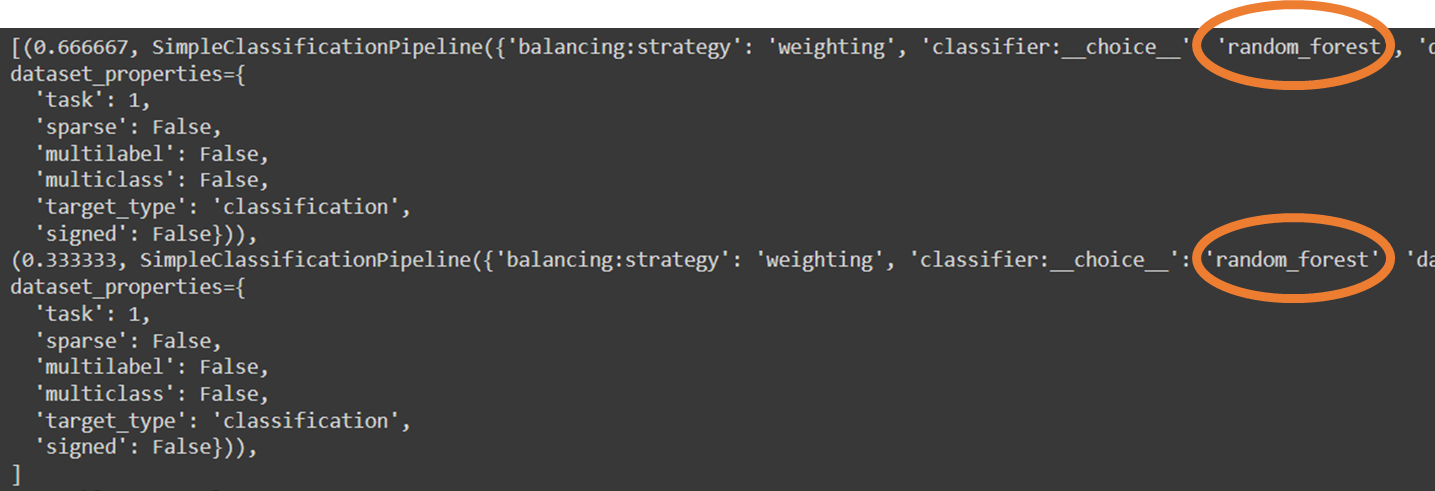


Figure 23: The Output Fragment of Automated Ensemble ML (Experiment 2)

The NCIPR test set is then passed to the pipeline, where the precision, recall, and accuracy scores are then calculated. The precision, recall, and accuracy scores are all 85.12 per cent of the precision score, 85.18 per cent of the recall score, and 85.18 per cent of the accuracy score, respectively as shown in [Figure 24](#fig21).

Precision: 85.12%

Recall: 85.18%

Accuracy: 85.18%

Figure 24: The Metric Scores of Automated Ensemble ML (Experiment 2)

* 1. **Experiment 3: NCIPR Dataset + AutoML with Different Generations**

In this experiment, the dataset setting remains the same. The AutoML library that generates a single ML model is applied. However, for each loop, the 'generations' will be set to 3, 7, and 10, as shown in [Figure 16](#fig15). The optimum pipeline and metric scores for each loop with a variable number of generations was shown in [Table 18](#table11). Furthermore, the single classifiers suggested are Gradient Boosting (GB), Random Forest, and GB once again as highlighted in [Table 18](#table11).

Table 18: The Experimental Results Generated from Each Loop in Different Generations (Experiment 3)

|  |  |  |
| --- | --- | --- |
| **Generations** | **Best Pipeline** | **Metric Scores** |
| 3 | GradientBoostingClassifier(input\_matrix, learning\_rate=0.1, max\_depth=1, max\_features=0.6000000000000001, min\_samples\_leaf=8, min\_samples\_split=16, n\_estimators=100, subsample=0.5) | Precision: 85.31%  Recall: 85.48%  Accuracy: 85.48% |
| 7 | RandomForestClassifier(Normalizer(input\_matrix, norm=l1), bootstrap=True, criterion=entropy, max\_features=0.4, min\_samples\_leaf=13, min\_samples\_split=13, n\_estimators=100) | Precision: 85.56%  Recall: 85.78%  Accuracy: 85.78% |
| 10 | GradientBoostingClassifier(input\_matrix, learning\_rate=0.1, max\_depth=1, max\_features=0.8, min\_samples\_leaf=17, min\_samples\_split=15, n\_estimators=100, subsample=0.9000000000000001) | Precision: 85.40%  Recall: 85.63%  Accuracy: 85.63% |

* 1. **Experiment 4: NCIPR Dataset + Automated Ensemble ML** **with Different Ensemble Size**

In this experiment, the dataset setting remains the same. The AutoML with ensemble features that generates the ensemble model is applied. However, for each loop, the 'ensemble\_size' will be set to 5, 7, and 10, as shown in [Figure 17](#fig16). In each loop, there are different combination of ensemble models produced, for example, the ensemble of three random forest and one passive aggressive model is generated when the ensemble size is set to 5. They are circled out in Figure 24. Besides, the example of output fragment is shown in Figure 25. The ensemble models produced and metric scores for each loop with a variety ensemble size are displayed in Table 19.

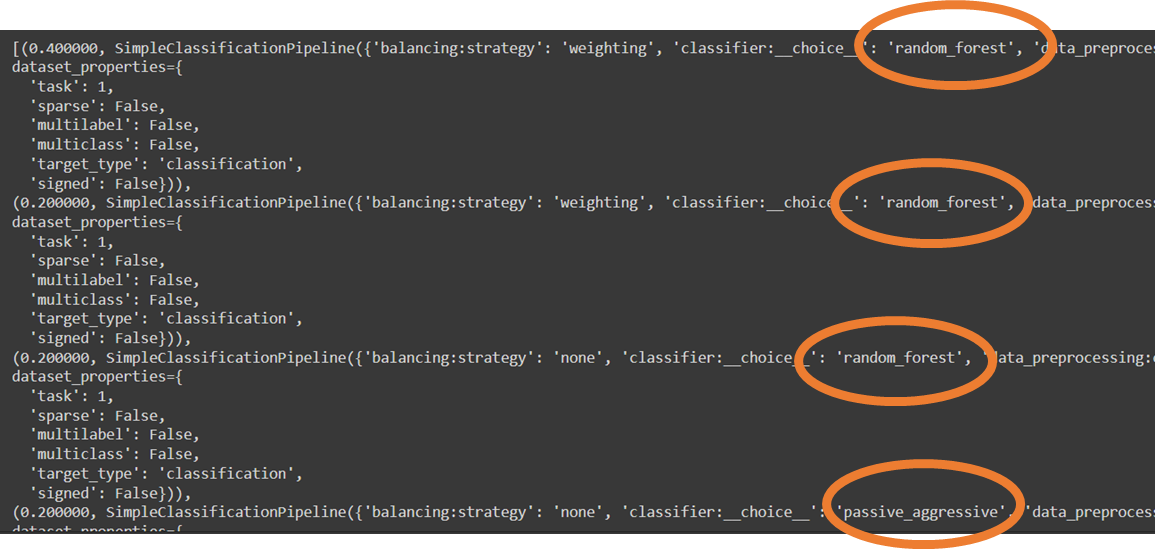


Figure 25: The Output Fragment of Ensemble ML in Different Ensemble Sizes (Experiment 4)

Table 19: The Experimental Results Generated from Each Loop in Different Ensemble Sizes (Experiment 4)

|  |  |  |
| --- | --- | --- |
| **Ensemble Size** | **Best Pipeline** | **Metric Scores** |
| 5 | Random Forest + Random Forest + Random Forest + Passive Aggressive | Precision: 84.23%  Recall: 84.58%  Accuracy: 84.58% |
| 7 | Random Forest + Random Forest + Random Forest + Linear discriminant analysis (LDA) + Passive Aggressive | Precision: 84.24%  Recall: 84.58%  Accuracy: 84.58% |
| 10 | LDA + Random Forest + Random Forest + Random Forest | Precision: 84.24%  Recall: 84.58%  Accuracy: 84.58% |

* 1. **Experiment 5: NCIPR Dataset + Feature Reduction + AutoML Single Classifier**

The AutoML library, TPOT, receives the NCIPR dataset's training set, which has fewer features, and because the parameter "generations" is set to 5, the pipeline optimization process will run for 5 iterations, with the best pipeline being provided as the output. According to AutoML, the XGB Classifier is the optimal pipeline in this situation, and the results are shown in Figure 25. The single classifier from TPOT that was recommended as well as the ideal parameter settings are highlighted in Figure 26.

Best pipeline: XGBClassifier(GradientBoostingClassifier(SGDClassifier(input\_matrix, alpha=0.001, eta0=1.0, fit\_intercept=True, l1\_ratio=0.75, learning\_rate=invscaling, loss=squared\_hinge, penalty=elasticnet, power\_t=0.1), learning\_rate=0.001, max\_depth=3, max\_features=0.25, min\_samples\_leaf=15, min\_samples\_split=19, n\_estimators=100, subsample=0.15000000000000002), learning\_rate=0.01, max\_depth=8, min\_child\_weight=12, n\_estimators=100, n\_jobs=1, subsample=0.8, verbosity=0)

Figure 26: The Output Fragment of AutoML Single Classifier (Experiment 5)

The NCIPR test set is then transferred to the pipeline, also shown in Figure 14, where the precision, recall, and accuracy scores are then calculated because this experiment's settings are the same as experiment 1 with the exception of the number of features used as inputs. The precision, recall, and accuracy scores are all 74.05% of the precision score, 75.15% of the recall score, and 75.15% of the accuracy score, respectively as shown in [Figure 27](#fig19).

Precision: 74.05%

Recall: 75.15%

Accuracy: 75.15%

Figure 27: The Metric Scores of AutoML Single Classifier (Experiment 5)

* 1. **Experiment 6: NCIPR Dataset + Feature Reduction + Automated Ensemble ML**

The NCIPR dataset's training and testing sets are sent to the Auto-sklearn with the similar settings as Experiment 2, the optimal ensemble model’s details can be printed out as shown in [Figure 15](#fig14). In this case, the automated ensemble ML library indicates that the best pipeline is an ensemble of two Extra Trees and LDA, each with its unique parameter values, as circled them out in [Figure 28](#fig20).

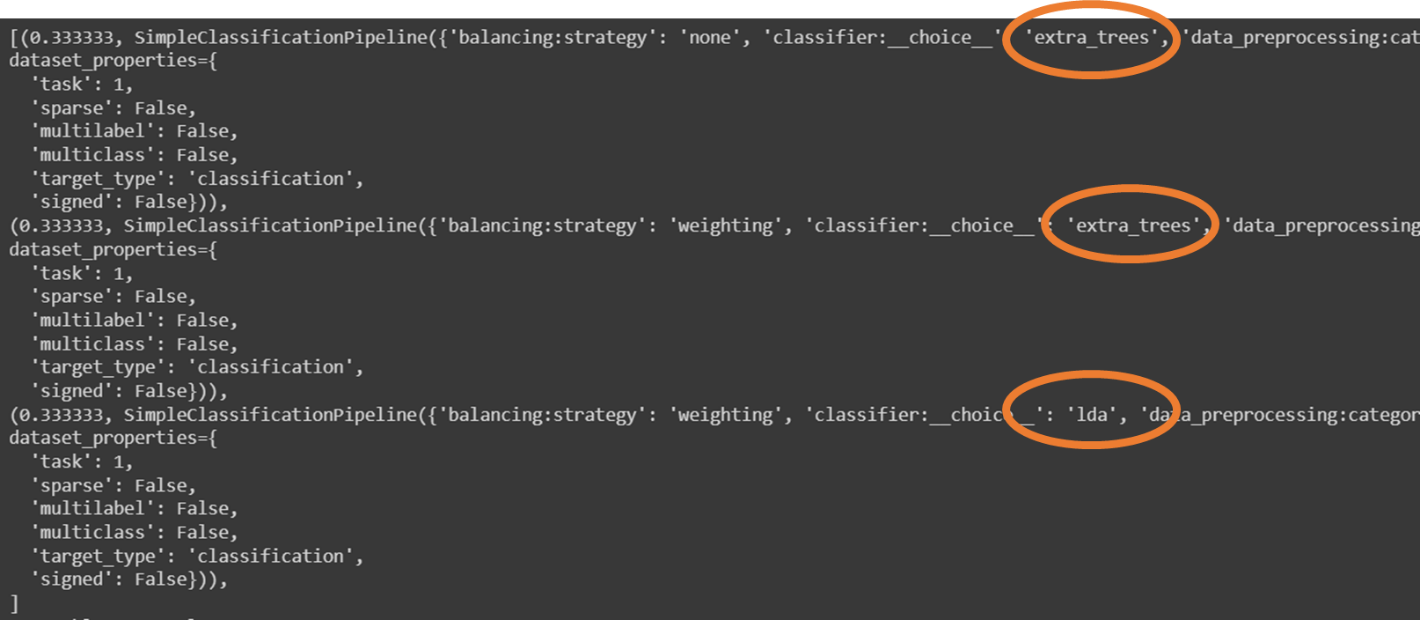


Figure 28: The Output Fragment of Automated Ensemble ML (Experiment 6)

The NCIPR test set is then passed to the pipeline, where the precision, recall, and accuracy scores are then calculated. According to Figure 29, the results are 72.70 percent for precision, 74.25 percent for recall, and 74.25 percent for accuracy.

Precision: 72.70%

Recall: 74.25%

Accuracy: 74.25%

Figure 29: The Metric Scores of Automated Ensemble ML (Experiment 6)

* 1. **Experiment 7: NCIPR Dataset + Kenya and Malawi Datasets + AutoML Single Classifier**

In this experiment, the NCIPR dataset are utilised with the new Kenya and Malawi datasets. Except for the datasets passed, the other parameter settings remain same as the experiment 1. According to AutoML, the XGB Classifier is the optimal pipeline in this situation and the results are shown in Figure 30. The single classifier from TPOT that was recommended as well as the ideal parameter settings are highlighted in Figure 30.

Best pipeline: XGBClassifier(MLPClassifier(SelectFwe(input\_matrix, alpha=0.019), alpha=0.1, learning\_rate\_init=0.5), learning\_rate=0.01, max\_depth=6, min\_child\_weight=2, n\_estimators=100, n\_jobs=1, subsample=0.9000000000000001, verbosity=0)

Precision: 99.26%

Figure 30: The Output Fragment of AutoML Single Classifier (Experiment 7)

The test set is then transferred to the pipeline, also shown in Figure 29, where the precision, recall, and accuracy scores are then calculated. The precision, recall, and accuracy scores are 99.26%, 66.35% and 66.35% respectively as shown in [Figure 31](#fig19).

Precision: 99.26%

Recall: 66.35%

Accuracy: 66.35%

Figure 31: The Metric Scores of AutoML Single Classifier (Experiment 7)

* 1. **Experiment 8: NCIPR Dataset + Kenya and Malawi Datasets + Automated Ensemble ML**

The combined datasets training and testing sets are sent to the Auto-sklearn with the similar settings as Experiment 2, the optimal ensemble model’s details can be printed out as shown in [Figure 32](#fig14). In this case, the automated ensemble ML library indicates that the best pipeline is an ensemble of two gradient boosting, extra trees classifier and LDA, each with its unique parameter values, as circled them out in [Figure 32](#fig20).

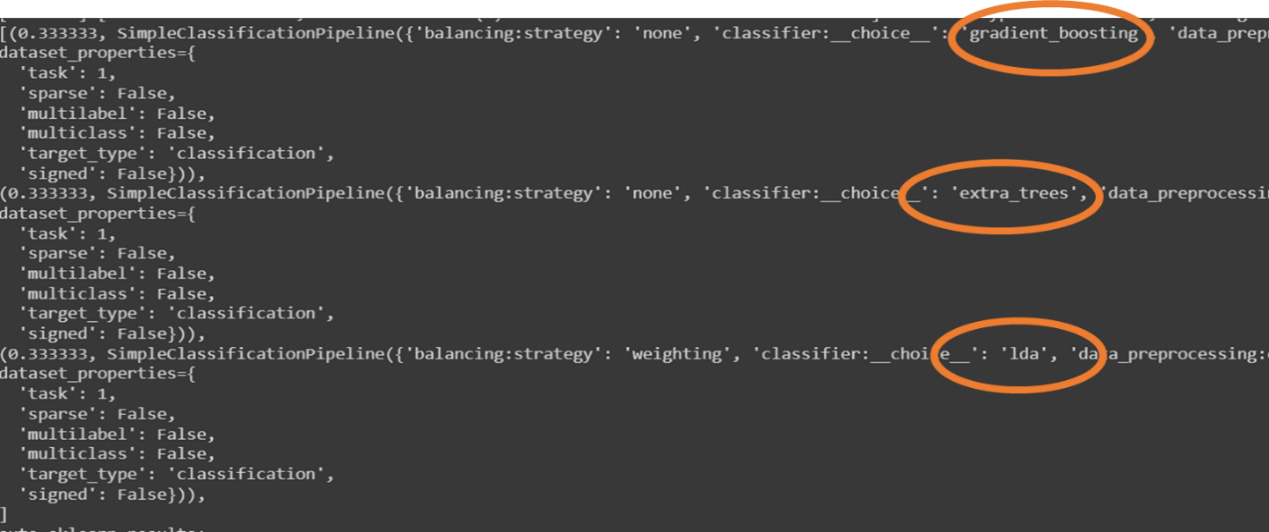


Figure 32: The Output Fragment of Automated Ensemble ML (Experiment 8)

The test set is then passed to the pipeline, where the precision, recall, and accuracy scores are then calculated. According to Figure 33, the results are 98.99% for precision, 41.16% for recall, and 41.16% % for accuracy.

Precision: 98.99%

Recall: 41.16%

Accuracy: 41.16%

Figure 33: The Metric Scores of Automated Ensemble ML (Experiment 8)

* 1. **Experiment 9: NCIPR Dataset + Kenya and Malawi Datasets + AutoML with Different Generations**

The Experiment 7 will be studied with manipulated parameter in this experiment. The AutoML library that generates a single ML model is applied. However, for each loop, the 'generations' will be set to 3, 7, and 10. The optimum pipeline and metric scores for each loop with a variable number of generations was shown in Table 20. Furthermore, the single classifiers suggested are Linear SVC, RF, Extra Trees with different suggested techniques should be applied as highlighted in [Table 20](#table11).

Table 20: The Experimental Results Generated from Each Loop in Different Generations (Experiment 9)

|  |  |  |
| --- | --- | --- |
| **Generations** | **Best Pipeline** | **Metric Scores** |
| 3 | LinearSVC(input\_matrix, C=0.01, dual=False, loss=squared\_hinge, penalty=l2, tol=0.001) | Precision: 98.91%  Recall: 36.84%  Accuracy: 36.84% |
| 7 | XGBClassifier(MLPClassifier(SelectFwe(input\_matrix, alpha=0.019), alpha=0.1, learning\_rate\_init=0.5), learning\_rate=0.01, max\_depth=6, min\_child\_weight=2, n\_estimators=100, n\_jobs=1, subsample=0.9000000000000001, verbosity=0) | Precision: 99.26%  Recall: 66.35%  Accuracy: 66.35% |
| 10 | XGBClassifier(MLPClassifier(SelectFwe(input\_matrix, alpha=0.019), alpha=0.1, learning\_rate\_init=0.5), learning\_rate=0.01, max\_depth=3, min\_child\_weight=2, n\_estimators=100, n\_jobs=1, subsample=0.9000000000000001, verbosity=0) | Precision: 99.24%  Recall: 63.75%  Accuracy: 63.75% |

* 1. **Experiment 10: NCIPR Dataset + Kenya and Malawi Datasets + Automated Ensemble ML with Different Ensemble Size**

In this experiment, the dataset setting remains the same. The AutoML with ensemble features that generates the ensemble model is applied. However, for each loop, the 'ensemble\_size' will be set to 5, 7, and 10. The ensemble models produced and metric scores for each loop with a variety ensemble size are displayed in [Table](#table12) 21.

Table 21: The Experimental Results Generated from Each Loop in Different Ensemble Sizes (Experiment 10)

|  |  |  |
| --- | --- | --- |
| **Ensemble Size** | **Best Pipeline** | **Metric Scores** |
| 5 | Extra Trees + Random Forest + LDA | Precision: 98.63%  Recall: 27.26%  Accuracy: 27.26% |
| 7 | Extra Trees + Random Forest + LDA | Precision: 98.63%  Recall: 27.26%  Accuracy: 27.26% |
| 10 | Extra Trees + Random Forest + LDA | Precision: 98.63%  Recall: 27.26%  Accuracy: 27.26% |

* 1. **Experiment 11: NCIPR Dataset + Kenya and Malawi Datasets + Oversampling Technique + AutoML Single Classifier**

Similar to Experiment 7, the training set in this experiment is the NCIPR dataset, while the test set is the combined dataset made up of the Kenya and Malawi datasets. According to AutoML, the XGB Classifier is the best pipeline in this scenario after using the SMOTEN to solve the imbalanced issue. The results are displayed in Figure 34. Figure 34 highlights the single classifier from TPOT that was suggested as well as the appropriate parameter settings.

Best pipeline: XGBClassifier(MLPClassifier(SelectFwe(input\_matrix, alpha=0.019), alpha=0.1, learning\_rate\_init=0.5), learning\_rate=0.01, max\_depth=6, min\_child\_weight=2, n\_estimators=100, n\_jobs=1, subsample=0.9000000000000001, verbosity=0)

Precision: 99.26%

Figure 34: The Output Fragment of AutoML Single Classifier (Experiment 11)

The test set is then transferred to the pipeline, also shown in Figure 34, where the precision, recall, and accuracy scores are then calculated. The precision, recall, and accuracy scores are 19.98%, 33.27% and 33.27% respectively as shown in [Figure 35](#fig19).

Precision: 19.98%

Recall: 33.27%

Accuracy: 33.27%

Figure 35: The Metric Scores of AutoML Single Classifier (Experiment 11)

* 1. **Experiment 12: NCIPR Dataset + Kenya and Malawi Datasets + Oversampling Technique + Automated Ensemble ML**

The combined datasets training and testing sets are sent to the Auto-sklearn with the similar settings as Experiment 8, the optimal ensemble model’s details can be printed out as shown in [Figure 36](#fig14). In this case, the automated ensemble ML library indicates that the best pipeline is an ensemble of two gradient boosting, extra trees classifier and LDA, each with its unique parameter values, as circled them out in [Figure 36](#fig20).

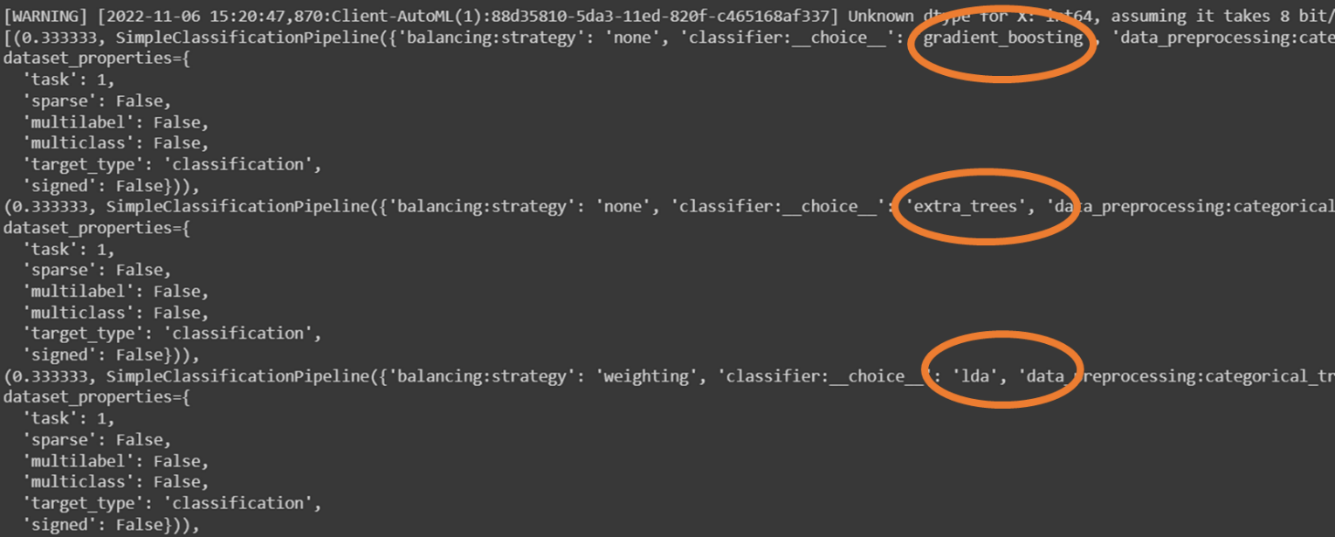


Figure 36: The Output Fragment of Automated Ensemble ML (Experiment 12)

The test set is then passed to the pipeline, where the precision, recall, and accuracy scores are then calculated. According to Figure 37, the results are 14.55% for precision, 20.52% for recall, and 20.52% for accuracy.

Precision: 14.55%

Recall: 20.52%

Accuracy: 20.52%

Figure 37: The Metric Scores of Automated Ensemble ML (Experiment 12)

* 1. **Experiment 13: Synthetic Data + AutoML Single Classifier**

After NCIPR joined with the synthetic data, the value count of each category in training and validation sets is 2744 and 1462 for non-long COVID (0) and long COVID (1) respectively as shown in Figure 38. According to AutoML, the MultinomialNB is the best pipeline in this scenario. The results are displayed in Figure 39. Figure 39 highlights the single classifier from TPOT that was suggested as well as the appropriate parameter settings.

Graphical user interface, application

Description automatically generated

Figure 38: The Value Count of Each Category with Synthetic Data

Best pipeline: MultinomialNB(SelectFwe(VarianceThreshold(input\_matrix, threshold=0.1), alpha=0.014), alpha=100.0, fit\_prior=False)

Figure 39: The Output Fragment of AutoML Single Classifier (Experiment 13)

The test set is then transferred to the pipeline, where the precision, recall, and accuracy scores are then calculated. The precision, recall, and accuracy scores are 99.40%, 95.39% and 95.39% respectively as shown in [Figure](#fig19) 40.

Precision: 99.40 %

Recall: 95.39%

Accuracy: 95.39%

Figure 40: The Metric Scores of AutoML Single Classifier (Experiment 13)

* 1. **Experiment 14: Synthetic Data + Automated Ensemble ML**

Similar to Experiment 13, the synthetic data is used. The optimal ensemble model’s details are printed out as shown in [Figure 41](#fig14). In this case, the automated ensemble ML library indicates that the best pipeline is an ensemble of two Random Forest and Adaboost with their unique parameter values, as circled them out in [Figure 41](#fig20).

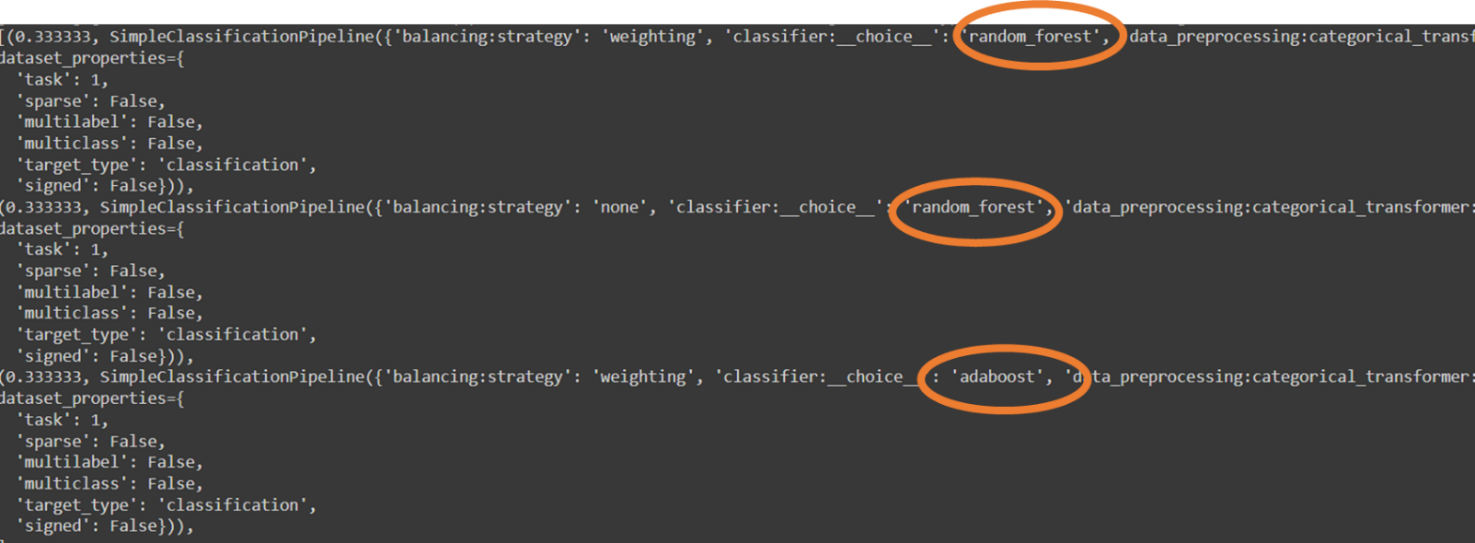


Figure 41: The Output Fragment of Automated Ensemble ML (Experiment 14)

The test set is then passed to the pipeline, where the precision, recall, and accuracy scores are then calculated. According to Figure 42, the results are 99.40% for precision, 98.17% for recall, and 98.17% for accuracy.

Precision: 99.40%

Recall: 98.17%

Accuracy: 98.17%

Figure 42: The Metric Scores of Automated Ensemble ML (Experiment 14)

* 1. **Conclusion**

In short, at the testing stage, the system can be run and function. AutoML libraries were successful in generating their best models; one offered the best single classifier, while the other proposed the best ensemble model.

**CHAPTER 6**

**RESULTS**

* 1. **Introduction**

This chapter summarises the findings obtained from the outcomes of the tests conducted in [Chapter 5](#imp). The first section contains a quick introduction. The outcomes of AutoML single classifier and ensemble ML are compared in the second section. Following that, a comparison of the AutoML single classifier in different generations is made. The performance of ensemble models in different ensemble sizes is compared in the following section. The outcomes of the AutoML single classifier and ensemble ML while encountering feature deduction are compared in Section 5. The contrast between a single classifier and an ensemble model that was run on a brand-new test set is then described. The following two sections compare results using modified parameters and using an oversampling technique. Before the conclusion which in the last portion, there is a section for the results discussion of applying synthetic data.

* 1. **Comparison between AutoML Single Classifier and Automated Ensemble ML**

When the outcomes of AutoML single classifier and ensemble ML are compared, the AutoML that develops a single classifier has slightly higher metrics scores than the ensemble models with similar parameter settings such as the resampling learning approach. These outcomes can be used to compare baseline models in subsequent studies. [Figure 43](#fig23) shows a comparison of AutoML single classifier and ensemble ML based on the scores of baseline models in bar charts.

Figure 43: Comparison of AutoML Single Classifier and Ensemble ML in terms of Baseline Models

* 1. **Comparison of AutoML Single Classifiers in Different Generations**

AutoML library using GP programming as an optimisation method has a robust structure that can generate similar results even when suggesting different single classifier. From the results, the three metric scores do not differ much from the baseline model, which set 'generations' to 5. The scores of increasing or decreasing the number of generations can be observed do not differ much from the baseline scores as shown in [Table 22](#table13). Based on [Table 22](#table13), the green shaded part is the results of baseline model and the orange shaded parts are the results of models with manipulated parameter. The models produced in this experiment will be a proof of increasing the number of 'generations' will not necessarily strengthen the performance, and this experiment will be used to satisfy objectives 3 and 4. All the scores except of the baseline single model’s scores are displayed in [Table 22](#table13) while the visualisation of the results is displayed in [Figure 44](#fig24).

Figure 44: Bar Chart of Metrics Scores VS Generations Running

Table 22: The Experimental Results of AutoML Single Classifiers

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Experiment** | **Generations** | **Precision (%)** | **Recall (%)** | **Accuracy (%)** |
| Experiment 1: Baseline | 5 | 85.56 | 85.78 | 85.78 |
| Experiment 3: Modify the parameter ‘generations’ | 3 | 85.31 | 85.48 | 85.48 |
| 7 | 85.56 | 85.78 | 85.78 |
| 10 | 85.40 | 85.63 | 85.63 |

* 1. **Comparison of Automated Ensemble ML in Different Ensemble Sizes**

From the results, the increasing of ensemble sizes does not necessarily lead to better performance. The three metric scores, according to the experiment results, performance decreases as the size of the size grows and the scores are lower than the baseline ensemble model in [Experiment 2](#exp2). Based on [Table 23,](#table14) the green shaded part is the results of baseline model and the orange shaded parts are the results of models with manipulated parameter. The visualisation of all the scores except of the baseline ensemble model’s scores are displayed in [Figure 45](#fig25) while the [Table 23](#table14) displays the experimental scores when comparing the baseline model and modified models. From Table 22, the recall and accuracy scores can be observed remain the same when the ensemble size increases.

Figure 45: Bar Chart of Metrics Scores VS Ensemble Sizes

Table 23: The Experimental Results of Ensemble ML

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Experiment** | **Ensemble Sizes** | **Precision (%)** | **Recall (%)** | **Accuracy (%)** |
| Experiment 2: Baseline | 3 | 85.12 | 85.18 | 85.18 |
| Experiment 4: Modify the parameter ‘ensemble\_size’ | 5 | 84.23 | 84.58 | 84.58 |
| 7 | 84.24 | 84.58 | 84.58 |
| 10 | 84.24 | 84.58 | 84.58 |

* 1. **Comparison of AutoML Single Classifiers and Automated Ensemble ML in Feature Reduction**

There are 35 features in Experiments 1, 2, 3, and 4, but since the next experiments use two quite distinct datasets from the NCIPR dataset, it is necessary to search for common features. Only 15 features are shared by all three datasets, thus two experiments are planned to examine the effects of fewer features before using the new test datasets. For both the AutoML single model and the AutoML ensemble models, the experiment conditions are the same as the baseline experiments. All of the scores are falling to between 70 percent and 76 percent, which is a substantial difference from the baseline levels as indicated in Table 24. The blue shaded part of Table 24 represents the outcomes of an AutoML single classifier, whereas the yellow shaded part represents the outcomes of an AutoML ensemble model. This experiment will be used to fulfil Objectives 1, 2, 3 and 4 and will employ models from AutoML's single classifier and ensemble models as proof that performance decreases as the number of features increases. The visual representation of the results is shown in Figure 44, and Table 24 lists all of the scores from the AutoML single classifier and ensemble ML. Figure 46 shows that the AutoML single classifier outperformed slightly the AutoML ensemble ML in terms of three metrics scores.

Figure 46: Bar Chart of Metrics Scores VS Type of Model in terms of Feature Reduction

Table 24: The Experimental Results of AutoML Single Classifiers and Automated Ensemble ML in terms of Feature Reduction

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Experiment** | **Single/ Ensemble** | **Number of Features** | **Precision (%)** | **Recall (%)** | **Accuracy (%)** |
| Experiment 1: Baseline | AutoML Single | 35 | 85.56 | 85.78 | 85.78 |
| Experiment 5 | 15 | 74.05 | 75.15 | 75.15 |
| Experiment 2: Baseline | AutoML Ensemble | 35 | 85.12 | 85.18 | 85.18 |
| Experiment 6 | 15 | 72.70 | 74.25 | 74.25 |

* 1. **Comparison of AutoML Single Classifiers and Automated Ensemble ML** **in Unseen Test Set**

The new test set is imported into the models created by the AutoML libraries in this experiment. The experiment circumstances are kept the same for the baseline experiments for both the AutoML single model and the AutoML ensemble models. The precision score of both the AutoML single classifier and the ensemble ML is enhanced to above 95%, according to the bar chart in Figure 47, but the recall and accuracy scores are much lower. This is because the test set is unbalanced; the ratio of patients with long COVID to those without it is 1686 to 5; as a result, the precision score is high, indicating that patients with long COVID are properly identified about 95% of the time. Additionally, accuracy demonstrates a measure of the pertinent data points because there is still a chance of false positives, but it is crucial to treat patients who do not genuinely have long COVID despite what the model predicted.

Recall rates for the AutoML ensemble model and single classifier are 66.35 and 41.36 percent, respectively. In particular for the ensemble model, both recall scores are dangerously low. Recall is a measurement of how accurately the model is able to recognise the relevant data, particularly for people who genuinely have long COVID. If the recall score is poor, the type-2 error is more likely to occur, the best timing to treat a long COVID patients will be missed. As there are only five instances are fall into long COVID group, it is more difficult that the models produced identifying actual long COVID patients. All of the results from the AutoML ensemble ML and single classifier are shown in Table 25. The blue shaded part represents the outcomes of an AutoML single classifier, whereas the yellow shaded part represents the outcomes of an AutoML ensemble model. The performance ratings clearly show that the models cannot be adjusted to completely new and untested test sets, indicating that there is still room for improvement in the generalisation of models.

Figure 47: Bar Chart of Metrics Scores VS Type of Model in Unseen Test Set

Table 25: The Experimental Results of AutoML Single Classifiers and Automated Ensemble ML in Unseen Test Set

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Experiment** | **Single/ Ensemble** | **Precision (%)** | **Recall (%)** | **Accuracy (%)** |
| Experiment 7 | AutoML Single | 99.26 | 66.35 | 66.35 |
| Experiment 8 | AutoML Ensemble | 98.99 | 41.16 | 41.16 |

* 1. **Comparison of AutoML Single Classifiers and Automated Ensemble ML in Unseen Test Set with Manipulated Parameter**

Some parameters are changed to improve the outcomes of prior studies while examining the effects of those changes. The modified parameters were selected based on the experiments 3 and 4. For the AutoML single classifier, the 'generations' parameter was set to 3, 7, and 10, while the 'ensemble sizes' parameter was set to 5, 7 and 10. The training dataset is NCIPR and the test set is the joined dataset which constituted by Kenya and Malawi datasets.

Despite changing the parameters values, all scores decreased or remained constant. The AutoML single classifier still has an edge over the ensemble ML. Additionally, the metrics scores show that changing the number of "generations" does not improve the performance of a single model; as a consequence, "generations" with a value of 5 yield the best results. The ensemble model's performance is not improved by using more ensemble models, hence the 'ensemble sizes' setting of 3 yielded the best results. The experimental results are shown in Table 26. The results of an AutoML single classifier are shown in the blue shaded area, whereas those of an AutoML ensemble model are shown in the yellow shaded area.

Table 26: The Experimental Results of AutoML Single Classifiers and Automated Ensemble ML with New Test Set and Manipulated Parameter

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Experiment** | **Single/ Ensemble** | **Manipulated Parameter** | | **Precision (%)** | **Recall (%)** | **Accuracy (%)** |
| Experiment 7: Baseline with new test set | AutoML Single | Generations | 5 | 99.26 | 66.35 | 66.35 |
| Experiment 9: Modify the parameter with new test set | 3 | 98.91 | 36.84 | 36.84 |
| 7 | 99.26 | 66.35 | 66.35 |
| 10 | 99.24 | 63.75 | 63.75 |
| Experiment 8: Baseline with new test set | AutoML Ensemble | Ensemble Sizes | 3 | 98.99 | 41.16 | 41.16 |
| Experiment 10: Modify the parameter with new test set | 5 | 98.63 | 27.26 | 27.26 |
| 7 | 98.63 | 27.26 | 27.26 |
| 10 | 98.63 | 27.26 | 27.26 |

* 1. **Comparison of AutoML Single Classifiers and Automated Ensemble ML** **in Unseen Test Set with Oversampling Technique**

Although the fact that both single classifier and ensemble models for AutoML have recommended the appropriate resampling approaches to overcome the imbalanced issue, the methods are suitable for and employed on the training dataset but not the test set. To balance the ratio of binary classes, an external oversampling technique is used on the test set. The results are shown in Table 27, and Figure 48 shows how the data were visualised. The blue shaded area shows the results of a single AutoML classifier, whereas the yellow shaded area shows the results of an AutoML ensemble model. It is evident from both the picture and the table that the oversampling strategy worsens rather than improves performance. All of the below-50 percent scores led to mediocre outcomes. Therefore, it is demonstrated that the oversampling technique used is inappropriate in this circumstance.

Figure 48: Bar Chart of Metrics Scores VS Type of Model in Unseen Test Set with Oversampling Technique

Table 27: The Experimental Results of AutoML Single Classifiers and Automated Ensemble ML with Oversampling Technique in Unseen Test Set with Oversampling Technique

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Experiment** | **Single/ Ensemble** | **Precision (%)** | **Recall (%)** | **Accuracy (%)** |
| Experiment 11 | AutoML Single | 19.98 | 33.27 | 33.27 |
| Experiment 12 | AutoML Ensemble | 14.55 | 20.52 | 20.52 |

* 1. **Comparison of AutoML Single Classifiers and Automated Ensemble ML with Synthetic Data**

Based on the results from previous experiments, it is clearly seen that the datasets have a serious imbalanced issue. Resampling techniques and ensemble models do not solve the problem satisfactory and resampling technique has one backslash which is making exact copies of existing examples, it makes overfitting likely to be occurred. The precision scores from last two experiments have proved that as the scores are deviated too far from accuracy and recall scores. To solve this issue, the three datasets used are joined and formed a single dataset, this method does not change the original number of data but can have a chance to solve this issue. The results are shown in Table 28, and Figure 49 shows how the data were visualised. The blue shaded area shows the results of a single AutoML classifier, whereas the yellow shaded area shows the results of an AutoML ensemble model. It is evident from both the picture and the table that the synthetic data has improved the scores. However, another issue is brought up, underfitting. The validation scores of AutoML single and ensemble model are around 70%, the testing scores are much higher which around 95%, so it is reasonable to suspect that there is underfitting problem.

Figure 49: Bar Chart of Metrics Scores VS Type of Model with Synthetic Data

Table 28: The Experimental Results of AutoML Single Classifiers and Ensemble ML with Synthetic Data

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Experiment** | **Single/ Ensemble** | **Precision (%)** | **Recall (%)** | **Accuracy (%)** |
| Experiment 11 | AutoML Single | 99.40 | 95.39 | 95.39 |
| Experiment 12 | AutoML Ensemble | 99.40 | 98.17 | 98.17 |

* 1. **Overall Experimental Results**

[Table 29](#table15) shows the overall experimental results, the blue shaded parts are the results of AutoML single classifiers while the light yellow shaded areas are the results of automated ensemble ML.

All the experiments can be divided into two main parts. The first part is utilising the NCIPR dataset only for training, validating and testing purposes. The experiments 1, 2, 3 and 4 are designed to fulfil the objectives 2, 3 and 4. The results shows that Experiment 1 and Experiment 3 yielded the highest and same scores but since Experiment 1 run for small number of ‘generations’, it saves time thus the model from Experiment 1 is the most optimal model for part one. Before utilising the new test set, the NCIPR dataset which has been deducted the number of features from 35 to 15 is test with the AutoML solely, the results shows the performance is downgraded as shown in Experiment 5 and Experiment 6.

For the second part of experiment, a new test set is imported. The subsequent experiments 7 to 14 are aimed to achieve the objectives 1, 2, 3 and 4. To solve the imbalanced issue in Kenya and Malawi datasets, oversampling and synthetic data generation are employed. Despite the results with synthetic data are improved and higher than the baseline model in Experiment 1, Given that the testing scores are about 95% and the validation scores are around 70%, it is evident that the underfitting issue needs to be addressed. The generalization ability of the model produced from AutoML libraries also be tested and the results proves that this ability still have room to be improved as the recall and accuracy scores are unsatisfactory and other issue is brought up.

Another crucial feature is that some experiments, including experiments 3, 4, and 9, entail modifying one of the parameters to see the effect after setting default parameter values. The results demonstrate that changing the "generations" and "ensemble size" does not improve the models; some continue to perform well, while others do worse. Additionally, using an oversampling strategy to resolve the imbalance in the test set has a negative impact on the outcomes. This suggests that using the oversampling strategy in these situations is unwise.

Table 29: Overall Experimental Results

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Experiment** | **Single/ Ensemble** | **Manipulated Parameter** | | **Precision (%)** | **Recall (%)** | | **Accuracy (%)** | |
| **PART I: NCIPR** | | | | | | | | |
| Experiment 1: Baseline | AutoML Single | - | 5 | 85.56 | 85.78 | | 85.78 | |
| Experiment 3: Modify the parameter | Generations | 3 | 85.31 | 85.48 | | 85.48 | |
| 7 | 85.56 | 85.78 | | 85.78 | |
| 10 | 85.40 | 85.63 | | 85.63 | |
| Experiment 5: Decrease number of features | - | 5 | 74.05 | 75.15 | | 75.15 | |
| Experiment 2: Baseline | AutoML Ensemble | - | 3 | 85.12 | 85.18 | | 85.18 | |
| Experiment 4: Modify the parameter | Ensemble Sizes | 5 | 84.23 | 84.58 | | 84.58 | |
| 7 | 84.24 | 84.58 | | 84.58 | |
| 10 | 84.24 | 84.58 | | 84.58 | |
| Experiment 6: Decrease number of features | - | 3 | 72.70 | 74.25 | | 74.25 | |
| **PART II: NCIPR + Kenya + Malawi** | | | | | | | | |
| Experiment 7: Baseline with new test set | AutoML Single | - | 5 | 99.26 | | 66.35 | | 66.35 | |
| Experiment 9: Modified parameter with new test set | Generations | 3 | 98.91 | | 36.84 | | 36.84 | |
| 7 | 99.26 | | 66.35 | | 66.35 | |
| 10 | 99.24 | | 63.75 | | 63.75 | |
| Experiment 11: Oversampling | - | 5 | 19.98 | | 33.27 | | 33.27 | |
| Experiment 13: Synthetic Data | - | 5 | 99.40 | | 95.39 | | 95.39 | |
| Experiment 8: Baseline with new test set | AutoML Ensemble | - | 3 | 98.99 | | 41.16 | | 41.16 | |
| Experiment 10: Modified parameter with new test set | Ensemble Sizes | 5 | 98.63 | | 27.26 | | 27.26 | |
| 7 | 98.63 | | 27.26 | | 27.26 | |
| 10 | 98.63 | | 27.26 | | 27.26 | |
| Experiment 12: Oversampling | - | 3 | 14.55 | | 20.52 | | 20.52 | |
| Experiment 14: Synthetic Data | - | 3 | 99.40 | | 98.17 | | 98.17 | |

* 1. **Conclusion**

In short, the AutoML single classifier in Experiment 1 that served as the baseline had the best performance with a precision score of 85.56%, a recall score of 85.78%, and an accuracy score of 85.78% when solely using the NCIPR dataset. Furthermore, in the studies that use Kenya and Malawi datasets to address the imbalanced problem, oversampling and synthetic data generation are used. The outcomes demonstrated that using synthetic data generation is more suitable.

**CHAPTER 7**

**CONCLUSION AND FUTURE WORK**

* 1. **Introduction**

This chapter will provide a conclusion and summary of the entire project by describing the work that has been completed. This chapter also offers a discussion of possible future projects for this project.

* 1. **Summary**

The occurrence of long COVID still under a lack condition currently. Since the discover of the related cases, health professionals carried out various statistical studies to find out the features that can be used to determine long COVID.

The main concern of this project is predicting the occurrence of long COVID using AutoML and ensemble ML. The first objective of this project is to curate and modify datasets from the existing long COVID datasets based on the common features in order to identify the most relevant features that develop long COVID. Therefore, the NCIPR dataset is used and 35 features are extracted to predict the long COVID. Also, 15 features are extracted for prediction from two additional datasets that are imported as test sets in order to assess the generalization of models.

The second objective is to design a comparison of automated machine learning models and ensemble models in terms of predicting the occurrence of long COVID and the third objective is to implement machine learning classifiers and ensemble models using Auto-sklearn and TPOT to predict the occurrence of long COVID. Thus, the all the experiments from 1 to 12 are designed and the AutoML libraries are applied to generate single ML classifier such as Gradient Boost and ensemble model such as the ensemble of different classifiers, the results of those models are compared.

The fourth objective is to evaluate the accuracy, precision, and sensitivity of machine learning models and ensemble models for predicting the presence of long COVID. Therefore, every pipeline generated in every experiment are evaluated by the three metric scores. Table 30 displays the current progress of project objectives in percentage.

Table 30: Table Of Objectives and Progress

|  |  |
| --- | --- |
| **Objective** | **Progress (%)** |
| To curate and modify datasets from the existing long COVID datasets based on the common features in order to identify the most relevant features that develop long COVID by applying data processing tools and automated feature engineering tools. | 100% Achieved |
| To design a comparison of automated machine learning models and ensemble models in terms of predicting the occurrence of long COVID. | 100% Achieved |
| To implement machine learning classifiers and ensemble models using Auto-sklearn and TPOT to predict the occurrence of long COVID. | 100% Achieved |
| To evaluate the accuracy, precision, and sensitivity of generated models for predicting the presence of long COVID. | 100% Achieved |

* 1. **Main Findings**

The findings in this project after conducted a series of experiments are as followings:

* The NCIPR dataset can be used to forecast the occurrence of long COVID, and the features collected are sufficient to look into long COVID.
* The TPOT and Auto-sklearn AutoML libraries have demonstrated their abilities in generating single models and ensemble models, respectively. Despite using distinct searching and optimising methods, two libraries provided comparable outcomes in the majority of the experiments.
* According to Experiments 1 through 12, the AutoML single models slightly outperformed the AutoML ensemble models in terms of performance scores. The findings of Experiments 13 and 14 show that the AutoML ensemble model did perform better when using synthetic data.
* The datasets from Kenya and Malawi present a challenge since they are severely unbalanced. Two strategies were used, and synthetic data generation proved to be the more efficient and acceptable way to address this problem.
  1. **Limitations**

There are two limitations in this project. First, the device used hardware and configurations. The device limited the number of iterations of AutoML libraries, if increases the parameter, the running time will be increased and might take few hours to few days to run. Second, the novelty and imbalanced issue in the datasets utilised in this project. The statistical research on long COVID did provide some useful information such as the demographic of patients are crucial to determine the occurrence of long COVID. However, there are different viewpoints on a person's medical history, medications consumed, or quarantine practises, it is still difficult to select the appropriate attributes before model training.

* 1. **Future Works**

In future, research should focus on Natural Language Processing (NLP) or Text Mining methods because some features contain descriptive information. Furthermore, it is important to look into the imbalanced problem that affected the datasets for Kenya and Malawi in greater detail and implement improved solutions as the underfitting issue is observed.

**REFERENCES**

Adiga, A., Wang, L., Hurt, B., Peddireddy, A., Porebski, P., Venkatramanan, S., ... & Marathe, M. (2021, August). All models are useful: Bayesian ensembling for robust high resolution covid-19 forecasting. In Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining (pp. 2505-2513).

Alnegheimish, S., Alrashed, N., Aleissa, F., Althobaiti, S., Liu, D., Alsaleh, M., & Veeramachaneni, K. (2020, October). Cardea: An open automated machine learning framework for electronic health records. In 2020 IEEE 7th International Conference on Data Science and Advanced Analytics (DSAA) (pp. 536-545). IEEE.

Anwar, T. (2021). COVID19 Diagnosis using AutoML from 3D CT scans. In Proceedings of the IEEE/CVF International Conference on Computer Vision (pp. 503-507).

Avetisian, M., Burenko, I., Egorov, K., Kokh, V., Nesterov, A., Nikolaev, A., ... & Umerenkov, D. (2021). CoRSAI: A system for robust interpretation of CT scans of COVID-19 patients using deep learning. ACM Transactions on Management Information Systems (TMIS), 12(4), 1-16.

Ayachit, S. S., Kumar, T., Deshpande, S., Sharma, N., Chaurasia, K., & Dixit, M. (2020, December). Predicting H1N1 and Seasonal Flu: Vaccine Cases using Ensemble Learning approach. In 2020 2nd International Conference on Advances in Computing, Communication Control and Networking (ICACCCN) (pp. 172-176). IEEE.

Bai, F., Tomasoni, D., Falcinella, C., Barbanotti, D., Castoldi, R., Mulè, G., ... & Monforte, A. D. A. (2022). Female gender is associated with long COVID syndrome: a prospective cohort study. Clinical Microbiology and Infection, 28(4), 611-e9.

Blomberg, B., Mohn, K. G. I., Brokstad, K. A., Zhou, F., Linchausen, D. W., Hansen, B. A., ... & Langeland, N. (2021). Long COVID in a prospective cohort of home-isolated patients. Nature Medicine, 27(9), 1607-1613.

Bodini, M., Rivolta, M. W., & Sassi, R. (2021, September). Classification of ECG Signals with Different Lead Systems Using AutoML. In 2021 Computing in Cardiology (CinC) (Vol. 48, pp. 1-4). IEEE.

Caruana, R., Niculescu-Mizil, A., Crew, G., & Ksikes, A. (2004, July). Ensemble selection from libraries of models. In Proceedings of the twenty-first international conference on Machine learning (p. 18).

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: synthetic minority over-sampling technique. Journal of artificial intelligence research, 16, 321-357.

Chopra, V., Flanders, S. A., O’Malley, M., Malani, A. N., & Prescott, H. C. (2021). Sixty-day outcomes among patients hospitalized with COVID-19. Annals of Internal Medicine, 174(4), 576-578.

Costa, A. B. D., Moreira, L., Andrade, D. C. D., Veloso, A., & Ziviani, N. (2021). Predicting the Evolution of Pain Relief: Ensemble Learning by Diversifying Model Explanations. ACM Transactions on Computing for Healthcare, 2(4), 1-28.

Danilatou, V., Antonakaki, D., Tzagkarakis, C., Kanterakis, A., Katos, V., & Kostoulas, T. (2020, October). Automated Mortality Prediction in Critically-ill Patients with Thrombosis using Machine Learning. In 2020 IEEE 20th International Conference on Bioinformatics and Bioengineering (BIBE) (pp. 247-254). IEEE.

Davis, H. E., Assaf, G. S., McCorkell, L., Wei, H., Low, R. J., Re'em, Y., Redfield, S., Austin, J. P., & Akrami, A. (2021). Characterizing long COVID in an international cohort: 7 months of symptoms and their impact. EClinicalMedicine, 38, 101019.

Dwivedi, P., Khan, A. A., Mugde, S., & Sharma, G. (2021, July). Diagnosing the major contributing factors in the classification of the fetal health status using cardiotocography measurements: An AutoML and XAI approach. In 2021 13th International Conference on Electronics, Computers and Artificial Intelligence (ECAI) (pp. 1-6). IEEE.

El Gannour, O., Cherradi, B., Hamida, S., Jebbari, M., & Raihani, A. (2022, March). Screening Medical Face Mask for Coronavirus Prevention using Deep Learning and AutoML. In 2022 2nd International Conference on Innovative Research in Applied Science, Engineering and Technology (IRASET) (pp. 1-7). IEEE.

Erickson, N., Mueller, J., Shirkov, A., Zhang, H., Larroy, P., Li, M., & Smola, A. (2020). Autogluon-tabular: Robust and accurate automl for structured data. arXiv preprint arXiv:2003.06505.

Feurer, M., Klein, A., Eggensperger, K., Springenberg, J., Blum, M., & Hutter, F. (2015). Efficient and robust automated machine learning. Advances in neural information processing systems, 28.

Goel, N., Goyal, N., Nagaraja, R., & Kumar, R. (2022). Systemic corticosteroids for management of ‘long-COVID’: an evaluation after 3 months of treatment. Monaldi Archives for Chest Disease.

Hensley, D., & Elgazzar, H. (2022, January). Classification of Stroke Victims through Supervised Machine Learning Algorithms and Ensemble Learning. In 2022 IEEE 12th Annual Computing and Communication Workshop and Conference (CCWC) (pp. 0058-0064). IEEE.

Hernandez, M., Epelde, G., Alberdi, A., Cilla, R., & Rankin, D. (2022). Synthetic Data Generation for Tabular Health Records: A Systematic Review. Neurocomputing.

Hooda, N., Gupta, R., & Gupta, N. R. (2020). Prediction of malignant breast cancer cases using ensemble Machine Learning: A case study of pesticides prone area. IEEE/ACM Transactions on Computational Biology and Bioinformatics, 1–1.

Hutter, F., Kotthoff, L., & Vanschoren, J. (2019). Automated machine learning: methods, systems, challenges (p. 219). Springer Nature.

Izdihar, K., Karim, M. A., Aresli, N. N., Radzi, S. F. M., Sabarudin, A., Yunus, M. M., Rahman, M.A.A., & Shamsul, S. (2021, July). Detection of Novel Coronavirus from Chest X-Ray Radiograph Images via Automated Machine Learning and CAD4COVID. In 2021 International Congress of Advanced Technology and Engineering (ICOTEN) (pp. 1-4). IEEE.

Jain, A., Wadhwa, M., & Garg, A. (2019, September). Hyper Optimized Molecular Subtype Prediction of Breast Cancer. In 2019 International Conference on Computing, Power and Communication Technologies (GUCON) (pp. 640-644). IEEE.

Jin, H., Song, Q., & Hu, X. (2019, July). Auto-keras: An efficient neural architecture search system. In Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining (pp. 1946-1956).

Kanter, J. M., & Veeramachaneni, K. (2015, October). Deep feature synthesis: Towards automating data science endeavors. In 2015 IEEE international conference on data science and advanced analytics (DSAA) (pp. 1-10). IEEE.

Karaarslan, F., Güneri, F. D., & Kardeş, S. (2022). Long COVID: Rheumatologic/musculoskeletal symptoms in hospitalized COVID-19 survivors at 3 and 6 months. Clinical rheumatology, 41(1), 289-296.

Katsimpras, G., Aisopos, F., Garrard, P., Vidal, M. E., & Paliouras, G. (2022). Improving Early Prognosis of Dementia Using Machine Learning Methods. ACM Transactions on Computing for Healthcare (HEALTH), 3(3), 1-16.

Kaur, H., Poon, P. K. C., Wang, S. Y., & Woodbridge, D. M. K. (2021, November). Depression Level Prediction in People with Parkinson’s Disease during the COVID-19 Pandemic. In 2021 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC) (pp. 2248-2251). IEEE.

Knight, D. R., Munipalli, B., Logvinov, I. I., Halkar, M. G., Mitri, G., & Hines, S. L. (2022). Perception, Prevalence, and Prediction of Severe Infection and Post-acute Sequelae of COVID-19. The American journal of the medical sciences.

Kozato, N., Mishra, M., & Firdosi, M. (2021). New-onset psychosis due to COVID-19. BMJ Case Reports CP, 14(4), e242538.

Kulkarni, G. N., Ambesange, S., Vijayalaxmi, A., & Sahoo, A. (2021, March). Comparision of diabetic prediction AutoML model with customized model. In 2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS) (pp. 842-847). IEEE.

Le, T. T., Fu, W., & Moore, J. H. (2020). Scaling tree-based automated machine learning to biomedical big data with a feature set selector. Bioinformatics, 36(1), 250-256.

LeDell, E., & Poirier, S. (2020, July). H2o automl: Scalable automatic machine learning. In Proceedings of the AutoML Workshop at ICML (Vol. 2020).

León, C. S., Cabon, S., Patural, H., Gascoin, G., Flamant, C., Roue, J. M., Favrais, G., Beuchée, A., Pladys, P., & Carrault, G. (2021). Evaluation of maturation in preterm infants through an ensemble machine learning algorithm using physiological signals. IEEE Journal of Biomedical and Health Informatics.

Lu, S., Wu, D., Zhang, Z., & Wang, S. H. (2021). An Explainable Framework for Diagnosis of COVID-19 Pneumonia via Transfer Learning and Discriminant Correlation Analysis. ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM), 17(3s), 1-16.

Manduchi, E., Le, T. T., Fu, W., & Moore, J. H. (2021). Genetic analysis of coronary artery disease using tree-based automated machine learning informed by biology-based feature selection. bioRxiv.

Mauldin, T., Ngu, A. H., Metsis, V., & Canby, M. E. (2020). Ensemble deep learning on wearables using small datasets. ACM Transactions on Computing for Healthcare, 2(1), 1-30.

Mikkelsen, M. E., Abramoff, B., Elmore, J. G., & Kunins, L. (2021). COVID-19: evaluation and management of adults following acute viral illness. UpToDate Accessed April 26.

Mustafa, A., & Rahimi Azghadi, M. (2021). Automated machine learning for healthcare and clinical notes analysis. Computers, 10(2), 24.

Nayve Villavicencio, C., Jeng, J. H., & Hsieh, J. G. (2021, September). Support Vector Machine Modelling for COVID-19 Prediction based on Symptoms using R Programming Language. In 2021 The 4th International Conference on Machine Learning and Machine Intelligence (pp. 65-70).

Ng, B. H., Ban, A. Y. L., Abeed, N. N. N., & Faisal, M. (2021). Organising pneumonia manifesting as a late-phase complication of COVID-19. BMJ Case Reports CP, 14(10), e246119.

Nolazco-Flores, J. A., Faundez-Zanuy, M., De La Cueva, V. M., & Mekyska, J. (2021). Exploiting Spectral and Cepstral Handwriting Features on Diagnosing Parkinson’s Disease. IEEE Access, 9, 141599-141610.

Nugroho, B., Yoganingrum, A., Febriandirza, A., & Prasetyadi, A. (2021, October). Predicting New Covid-19 Outbreak Clusters among the “Science Denial” Communities. In The 2021 International Conference on Computer, Control, Informatics and Its Applications (pp. 126-130).

Orrù, G., Bertelloni, D., Diolaiuti, F., Mucci, F., Di Giuseppe, M., Biella, M., Gemignani, A., Ciacchini, R. and Conversano, C., 2021. Long-COVID Syndrome? A Study on the Persistence of Neurological, Psychological and Physiological Symptoms. Healthcare, 9(5), p.575.

Rodriguez-Almeida, A. J., Fabelo, H., Ortega, S., Deniz, A., Balea-Fernandez, F. J., Quevedo, E., Soguero-Ruiz, C., M. Wägner, A. & Callico, G. M. (2022). Synthetic Patient Data Generation and Evaluation in Disease Prediction Using Small and Imbalanced Datasets. IEEE Journal of Biomedical and Health Informatics.

Sudre, C. H., Murray, B., Varsavsky, T., Graham, M. S., Penfold, R. S., Bowyer, R. C., ... & Steves, C. J. (2021). Attributes and predictors of long COVID. Nature medicine, 27(4), 626-631.

Sykes, D. L., Holdsworth, L., Jawad, N., Gunasekera, P., Morice, A. H., & Crooks, M. G. (2021). Post-COVID-19 symptom burden: what is long-COVID and how should we manage it?. Lung, 199(2), 113-119.

Tahmasebi, A., Wang, S., Daniels, K., Cottrill, E., Liu, J. B., Xu, J., Lyshchik, A., & Eisenbrey, J. R. (2020, September). Ultrasonographic risk stratification of indeterminate thyroid nodules; a comparison of an artificial intelligence algorithm with radiologist performance. In 2020 IEEE International Ultrasonics Symposium (IUS) (pp. 1-4). IEEE.

Taquet, M., Dercon, Q., Luciano, S., Geddes, J. R., Husain, M., & Harrison, P. J. (2021). Incidence, co-occurrence, and evolution of long-COVID features: A 6-month retrospective cohort study of 273,618 survivors of COVID-19. PLoS medicine, 18(9), e1003773.

Taribagil, P., Creer, D., & Tahir, H. (2021). ‘Long COVID’syndrome. BMJ Case Reports CP, 14(4), e241485.

Thomason, M. E., Werchan, D., & Hendrix, C. (2021, November 30). Novel Coronavirus (COVID) Illness – Patient Report (NCIPR). https://doi.org/10.17605/OSF.IO/82RKJ

Ucuzal, H., YAŞAR, Ş., & Çolak, C. (2019, October). Classification of brain tumor types by deep learning with convolutional neural network on magnetic resonance images using a developed web-based interface. In 2019 3rd International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT) (pp. 1-5). IEEE.

Waring, J., Lindvall, C., & Umeton, R. (2020). Automated machine learning: Review of the state-of-the-art and opportunities for healthcare. Artificial Intelligence in Medicine, 104, 101822.

Wellcome. (2022). Kenya, Malawi, Long Covid-19 effects survey dataset - Humanitarian Data Exchange. Data.humdata.org. Retrieved 22 February 2022, from <https://data.humdata.org/dataset/long-covidresearchagenda>.

Xue, H., & Salim, F. D. (2021, August). Exploring self-supervised representation ensembles for covid-19 cough classification. In Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining (pp. 1944-1952).

Yang, D., Myronenko, A., Wang, X., Xu, Z., Roth, H. R., & Xu, D. (2021). T-AutoML: Automated Machine Learning for Lesion Segmentation using Transformers in 3D Medical Imaging. In Proceedings of the IEEE/CVF International Conference on Computer Vision (pp. 3962-3974).

Yang, J., Shi, R., & Ni, B. (2021, April). Medmnist classification decathlon: A lightweight automl benchmark for medical image analysis. In 2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI) (pp. 191-195). IEEE.

Zhang, M., Li, H., Pan, S., Lyu, J., Ling, S., & Su, S. (2021). Convolutional neural networks-based lung nodule classification: A surrogate-assisted evolutionary algorithm for hyperparameter optimization. IEEE Transactions on Evolutionary Computation, 25(5), 869-882.

**APPENDIX A: Correction FYP 1 Presentation**

**REVIEWER 1/EXAMINER 1**

|  |  |
| --- | --- |
| **Comments** | **Reply to Comments** |
| - | - |

**REVIEWER 2/EXAMINER 2**

|  |  |
| --- | --- |
| **Comments** | **Reply to Comments** |
| 1. Include the TP, TN, FP and FN rate or confusion matrix. | * Will be included. |

**APPENDIX B: Meeting Logs**

**A picture containing table

Description automatically generated**

**APPENDIX C: Turnitin Plagiarism Proposal**