Automl.ai – Robust & accurate Automl for structured data

Thesis Submitted in partial fulfillment
of the Requirements of the Degree of

Master In Science with Specialization in

Artificial Intelligence

by

Mohapatra Shibu Trilochan

Roll Number – 02

G.R. Number – 3480531

Under the Supervision of

Prof. Asmita Marathe

April 2023

Nagindas Khandwala College (Autonomous)

Malad, Mumbai 400064

CERTIFICATE

This is to certify that the dissertation titled, **“Automl.ai – Robust & accurate Automl for structured data”**, is bonafied work of **“MOHAPATRA SHIBU TRILOCHAN”** (**Roll No: 02** and **G.R. No: 3480531**) submitted to the Nagindas Khandwala College (Autonomous), Mumbai in partial fulfillment of the requirements for the award of degree of **“Masters In Science with Specialization in Artificial Intelligence”.**

Prof. Asmita Marathe

Internal Examiner

External Examiner

Supervisor’s Certificate

This is to certify that the dissertation entitled **“Automl.ai – Robust & accurate Automl for structured data”** submitted by **“MOHAPATRA SHIBU TRILOCHAN”**, **Roll No: 02** and **G.R. No: 3480531**, is a record of original work carried out by him under my supervision and guidance in partial fulfillment of the requirements of the degree of **Masters In Science with Specialization in Artificial Intelligence** at Nagindas Khandwala College (Autonomous), Mumbai 400064. Neither this dissertation nor any part of it has been submitted earlier for any degree or diploma to any institute or university in India or abroad.

Prof. Asmita Marathe

Internal Examiner External Examiner

Declaration of Originality

I, ***Mohapatra Shibu Trilochan, Roll No: 02*** *and* ***G.R. No: 3480531***, hereby declare that this dissertation entitled ***“Automl.ai – Robust & accurate Automl for structured data”*** presents my original work carried out as a Master Student of Nagindas Khandwala College (Autonomous), Mumbai 400064. To the best of my knowledge, this dissertation contains no material previously published or written by another person, nor any material presented by me for the award of any degree or diploma of Nagindas Khandwala College (Autonomous), Mumbai or any other institution. Any contribution made to this research by others, with whom I have worked at Nagindas Khandwala College (Autonomous), Mumbai or elsewhere, is explicitly acknowledged in the dissertation. Works of other authors cited in this dissertation have been duly acknowledged under the sections “Reference” or “Bibliography”. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I am fully aware that in case of any non-compliance detected in future, the Academic Council of Nagindas Khandwala College (Autonomous), Mumbai may withdraw the degree awarded to me on the basis of the present dissertation.

Date: 11-April-2023

Place: Malad, Mumbai 400064

Mohapatra Shibu Trilochan

Acknowledgement

It's my pleasure to be indebted to varied people, who directly or indirectly contributed to the event of this work and who influenced our thinking, behavior and acts during the course of study.

I express my sincere gratitude to Coordinator **Dr. Pragati Hiwarkar** for her support, cooperation, and motivation provided to me during the training for constant inspiration, presence and blessings.

I'm thankful to the Project guide, **Prof. Asmita Marathe**, for her valuable suggestions which helps us lot in completion of this project. Lastly,

I might wish to thank the almighty and our parents for his or her moral support and friends with whom we shared our day-to-day experience and received plenty of suggestions that improved our quality of work.

Thanks for all your encouragement!

Mohapatra Shibu Trilochan

Abstract

The necessity of an effective pipeline for building machine learning models has never been more pressing, given the boom in the usage of machine learning across a variety of domains and fields. The Creation and training of models, however, still primarily follow traditional methods, relying on Machine Learning specialists, Data Scientists and requiring time-consuming data manipulation procedures.

This hinders the advancement of machine learning models in both academia and industry. This demand supports AutoML, a new paradigm focusing on fully automatic machine-learning model fitting. The goal of Automated Machine Learning (AutoML) is to provide an end-to-end procedure and fully automate the model development pipeline.

The exploratory field of automated machine learning (AutoML) has recently attracted practically all complexity. AutoML, which came into existence in 2013's took a crucial step to assist the data science measurement and acts as a bridge between various levels of expertise while creating machine learning frameworks. The model development workflow is capable to be automated from beginning to end with the help of AutoML, eliminating the need for outside support.

In the first part, we will discuss Automl, a brief discussion about the problem statement, regarding the existing system and its loopholes (if any) and a brief about the proposed system named Automl.ai. In the second part, we will focus on the literature survey and cite a few of the previously published papers. In the third part, we will discuss the methodology part where the execution/flow of the proposed system is stated with all the libraries, technologies and outcomes discussed.

On another part of the methodology, all the requirements and results of the proposed system are stated in images and tables to understand easier. In the fourth part, an overview of the AutoMl and automl.ai is discussed with future work to be initiated on the currently proposed system and lastly, a list of references is mentioned.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

List of Abbreviation

Automl	Automated Machine Learning
Sklean	Scikit-Learn library
cm	Classification Model
rm	Regression Model
df	Dataframe
pd	Pandas
NAS	Neural architecture search
CV	Cross Validation
TPR	True Positive Rate
AUC	Area Under ROC Curve

CHAPTER 1

INTRODUCTION

Machine learning, a branch of the larger area of artificial intelligence, is dominating a wide range of commercial sectors and this covers a range of industries, such as entertainment, finance, retail, automotive and healthcare. Learning how to use machine learning is becoming more and more crucial as a result of its growing acceptance across all types of operations and by a workforce with a diverse range of talents.

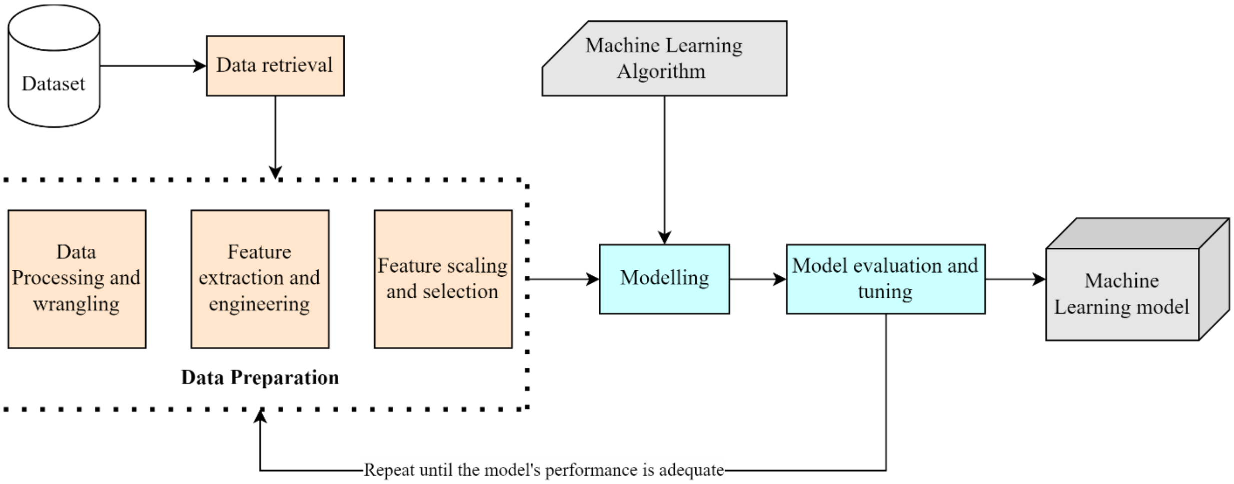

Dataset
Data retrieval
Machine Learning Algorithm
Data Processing and wrangling
Feature extraction and engineering
Feature scaling and selection
Data Preparation
Modelling
Model evaluation and tuning
Machine Learning model
Repeat until the model's performance is adequate

Fig 1: Traditional Machine Learning Pipeline

It is essential to create solutions that business professionals from a variety of backgrounds can use given the increase of Machine Learning usage by an organisation's constantly expanding staff base. Therefore, unlike the systems utilised by software engineers or data scientists, these systems cannot be solely coding or programming oriented. The AutoML systems come into play at this point.

As attention, importance, and the number of applications for Automated Machine Learning (AutoML) increase quickly, it is emerging as a distinct, autonomous sub-field of Machine Learning. The objectives of AutoML are to fully automate the use of complex data analytics techniques such as machine learning, statistical modelling, data mining, pattern recognition, and others. Therefore, AutoML can democratise machine learning (ML) to non-experts (Citizen Data Scientists), increase professional productivity, guard against statistical methodological errors, and even outperform manual expert analysis performance.

AutoML might make analyses more reproducible, make the results more accessible, and make collaborative analysis easier. To clarify the term AutoML, we consider the minimal requirements to be the ability to return a predictive model that can be applied to new data, and an estimate of the predictive performance of that model, given a data source

To elaborate concisely, AutoML enables developers to train high-quality models tailored to their company needs even with limited Machine Learning (ML) knowledge and coding experience.

1.1. Problem statement of thesis

The problem statement of AutoML is to simplify and automate the machine learning process for organizations, especially those that may not have the necessary expertise or resources to build effective machine learning models. Traditional machine learning requires significant domain knowledge, technical expertise, and time-consuming manual processes to select the right algorithm, tune hyperparameters, and deploy the model.

The problem statement for AutoML is to reduce the time, effort, and expertise required to build and deploy high-performing machine learning models. Traditionally, building a machine learning model involves a lot of trial-and-error experimentation, and requires significant domain expertise, mathematical and statistical knowledge, and programming skills.

Given a dataset and a machine learning task, build an automated system that can perform the entire process of building and deploying a machine learning model. The proposed AutoML system should be able to automatically perform tasks such as data pre-processing, feature selection, model selection, hyperparameter tuning and model generating while showing all the machine learning algorithms comparison with best algorithm selected. With addition to the execution, the proposed system should forecast and predict the score of each column. This shows a minute clearance that the model generated by the proposed AutoML system is working fine. This entire flow is executed without requiring any significant human intervention.

1.2. Existing System

AutoML (Automated Machine Learning) systems do exist and are becoming increasingly popular in the field of machine learning. These systems use various algorithms and techniques to automate the process of building and selecting the best machine learning models for a given dataset. There are several AutoML tools and frameworks available today, including Microsoft Azure AutoML and H2O.ai among others. These systems can help automate the machine learning process and save time and effort in developing accurate and reliable models. These are just a few examples of the existing AutoML systems available. Each system has its own set of features and limitations, and the choice of system will depend on the specific requirements of the task at hand.

To highlight some off the drawbacks in the existing application, there are some manual interventions present that require human touch or involvement. The existing system only highlights the outline of the Exploratory Data Analysis even though on most cases all the analysis is required with proper graphs and depictions. The existing also lacks to consider and select parameters on their own, there is a human intervention in selection of parameters which make overall model building slow. The existing software is also limited to use only one algorithm at a time even though that model had many algorithms. The exiting system does not show a comparison table and their performance metrics.

1.3. Objective of Project

The main objective of AutoML (Automated Machine Learning) is to automate the process of building machine learning models, thereby reducing the manual effort and time required to develop accurate and robust models. AutoML systems can perform tasks such as data preprocessing, feature engineering, model selection, hyperparameter tuning, and model evaluation, making it easier for non-experts to apply machine learning to their own problems.

By automating the machine learning process, AutoML systems can accelerate the development of machine learning models, making it possible for organizations to apply machine learning to a wider range of applications and problems. AutoML can also help to reduce the potential for human error and bias, and can lead to better model performance and generalization.

The objective of the proposed AutoML application system aims to automate these processes, making it easier for non-experts to build and deploy machine learning models. The ultimate goal of the proposed AutoML system is to democratize machine learning and make it accessible to a wider audience, including any-sized businesses, researchers, and individuals.

The ultimate goal of this AutoML application is to create a system that can quickly and accurately build high-quality machine learning models, even for users without significant machine learning expertise. This can significantly reduce the time and resources required to build and deploy machine learning models, making it more accessible to a wider range of users and use cases.

1.4. Proposed System

Since the existing AutoML systems are integrated into a form of API, a software feature or connected within a software or application. There is no standalone application or software as such. The proposed AutoML system introduces a dedicated standalone application for the industry and end-user. The proposed system involves Data pre-processing, Feature Engineering, Model Selection, Model training and evaluation and Model generation to deploy it.

The proposed AutoML system introduces an affordable solution for users and industries. Click button execution with no keyboard interference. Robust, efficient and easy-to-use Application approach. Overall, the proposed AutoML system would aim to automate the entire machine learning process, from data pre-processing to model generation and deployment, and provide accurate and reliable models for real-world use. Unlike the existing system the proposed AutoML system takes the dataset from the user. It does profiling phase by executing all the known EDA techniques. It also offers user to select a particular model from a range of multiple models as well as selection of target variable to build up the entire machine learning model. The proposed system takes all the machine learning parameters on its own for prediction. To add on to that the proposed system displays a range of machine learning algorithms on which the dataset was trained on and makes machine learning model (pkl or sav) based on the best performed algorithm. And lastly, it also has a forecasting feature to predict the scores of each row of dataset. The forecast feature is beneficial for monitoring purpose and give a clearance that the model build by the proposed system is working well.

CHAPTER 2

LITERATURE SURVEY

AutoML (Automated Machine Learning) is a rapidly growing field, and there is a significant amount of literature available on the topic. A literature survey in this field would typically involve reviewing research papers, books, and other publications related to AutoML. A literature survey of AutoML would involve reviewing existing research and literature related to automated machine learning. Some of the key topics and trends in AutoML literature include:

- AutoML techniques: There are various AutoML techniques, including neural architecture search, hyperparameter optimization, and meta-learning. Several papers have proposed novel AutoML techniques or improved existing ones.
- Domain-specific AutoML: Some papers have focused on developing AutoML systems for specific domains, such as computer vision, natural language processing, or time-series analysis.
- Explainable AutoML: Many papers have highlighted the need for AutoML systems to be transparent and explainable, especially in applications where decisions can have significant consequences.
- Benchmarking AutoML: Several papers have proposed benchmarks for evaluating AutoML systems' performance and compared different AutoML techniques on various datasets.
- Integration with cloud computing: Many papers have explored the potential for integrating AutoML with cloud computing platforms to enable more scalable and efficient machine learning.

Overall, AutoML literature survey can provide insights into the latest trends, techniques, and challenges in this field and help researchers and practitioners develop more effective AutoML systems. AutoML is an active and growing field of research with promising potential for automating the machine learning process and improving the performance of machine learning models.

2.1. Review of Literature

A literature review of AutoML would involve analysing and summarizing existing research and literature related to automated machine learning. here has been a significant increase in the number of research papers and commercial products related to AutoML in recent years.

The paper [1] "Automated Machine Learning (AutoML): an overview of opportunities for application and research" by Kailash Joshi (2022). In this paper, they introduce AutoML, identify some of the fundamental steps in model development, and currently available operationalizations of these systems, before concluding with an outline of potential research opportunities for IS researchers in the field.

The paper [2] "AutoML: A Survey of the State-of-the-Art" by Hutter et al. (2019) introduced AutoML methods according to the pipeline, covering data preparation, feature engineering, hyperparameter optimization, and neural architecture search. They focus more on neural architecture search, as it is currently very hot sub-topic of AutoML. We summarize the performance of the representative neural architecture search algorithms on the ImageNet datasets and further discuss several worthy studying directions of NAS methods: one/two-stage neural architecture search and joint hyperparameter and architecture optimization.

In paper [3] "An Open Source AutoML Benchmark" by Pieter Gijsbers (ICML 2019) discussed n open, ongoing, and extensible benchmark framework which follows best practices and avoids common mistakes. The framework is open-source, uses public datasets and has a website with up-to-date results. They use the framework to conduct a thorough comparison of 4 AutoML systems across 39 datasets and analyze the results. They had features comprising (a) curated suites of benchmarking datasets from OpenML. (b) Includes code to benchmark a number of popular AutoML systems on regression and classification tasks. (c) New AutoML systems can be added (d) Experiments can be run in Docker or Singularity containers.

In paper [8] "Automated Machine Learning: Methods, Systems, Challenges" by Feurer et al. (2019) intended to provide some background and starting points for researchers interested in developing their own AutoML approaches, highlight available systems for practitioners who want to apply AutoML to their problems, and provide an overview of the state of the art to researchers already working in AutoML.

CHAPTER 3

METHODOLOGY

The approach is to automate the Machine Learning modelling pipeline. The pipeline consists of Data Acquisition, Data Exploration, Data preparation, Feature Engineering, Model Selection, Model Training, Hyperparameter tuning and Prediction. With the help of technologies mentioned below the entire Machine Learning Pipeline will automate and a Machine Learning Model file that is a pkl or sav file will be available to download for the further deployment.

- Pandas profiling – It is used to help in creating Exploratory Data Analysis (EDA) experience in a consistent and fast solution. It has a method named as ProfileReport() which generate a basic report on the input DataFrame.
- Scikit-Learn – Also known as sklearn. It is implemented to perform machine learning modelling and statistical analysis. With the use of Scikit-Learn a user can make classification, regression or clustering models depending on the type of dataset.
- StreamLit – It helps us create web apps for data science and machine learning. It is compatible with major python libraries such as scikit-learn, keras, PyTorch, NumPy, Pandas, Matplotlib, etc.
- PyCaret – It is an open-source, low-code machine learning library in Python to make machine learning workflows. PyCaret is Fast, Explainable and Scalable which is beneficial for MLOps.

The entire process of the proposed Automl system is depicted in the form of flowchart. In the system the user just has to upload the dataset and then the Automl proceeds with further process following with the profiling part where Data Exploration, Data preparation, Feature Engineering is handled along with graphs and figures. In short, the profiling is Explanatory Data Analysis (EDA) is done. Next the system proceeds further with Model Selection, here user has to select the type of ML algorithm for the dataset (Classification or Regression model) and then selection of target variable to train the model.

When a user clicks the train the model button the training begins, after a while a table of multiple Machine Learning models along with their accuracies is displayed. Based on the best Machine Leaning algorithm and parameters taken the model is ready to downloaded.

Additionally, during the modelling phase the proposed system is concrete and efficient enough that it considers all the parameters based on the algorithm selected all by its own with help of PyCaret and Scikit-Learn and make a comparison table of all the model with their respective performance.

For instance, if a user selects classification model type by selecting the Classification button, the proposed system with choose the parameters and statistical analysis on its own and it will display a comparison of all the possible classification algorithms in a table format.

Here for building the model GridSerchCV and K-fold cross are used. For fitting and prediction Accuracy, AUC, Recall, Precision and F1 score are taken. Each of the terms can be stated with a short description.

- GridSearchCV – A set of hyperparameters and their respective values are specified, and the algorithm exhaustively searches through all possible combinations of hyperparameters to find the one that yields the best results. It does this by performing cross-validation on each combination of hyperparameters, and calculating the average performance score for each combination.
- CV – The "CV" in GridSearchCV stands for cross-validation, which is a technique used to evaluate the performance of a model by splitting the dataset into multiple subsets, training the model on one subset, and testing it on another. This process is repeated multiple times, with different subsets of the data used for training and testing, and the results are averaged to give a more accurate estimate of the model's performance.
- K-fold cross – A technique used to evaluate the performance of a machine learning model by dividing the dataset into k equal parts or "folds". The model is trained on k-1 folds, and the remaining fold is used for testing. This process is repeated k times, with a different fold used for testing each time, and the results are averaged to give an estimate of the model's performance.

- Accuracy – A metric used to evaluate the performance of a classification model. It measures the percentage of correct predictions made by the model, i.e., the ratio of the number of correct predictions to the total number of predictions made by the model.
- AUC – A measure of the overall performance of the model, and it ranges from 0 to 1, with 1 indicating a perfect classifier and 0.5 indicating a random classifier. The AUC represents the probability that a randomly chosen positive sample will be ranked higher than a randomly chosen negative sample, according to the classifier.
- Recall – Also known as sensitivity or true positive rate (TPR), is a metric used to evaluate the performance of a binary classification model. It measures the proportion of true positive cases that are correctly identified by the model, out of all positive cases in the dataset.
- Precision – It is defined as the ratio of true positives to the total number of predicted positives. In other words, precision measures the proportion of true positive predictions among all positive predictions made by the model.
- F1 score – The F1 score combines both precision and recall into a single score that ranges from 0 to 1, with higher values indicating better performance. It is calculated using the following formula: **F1 score = 2 * (precision * recall) / (precision + recall)**

To summarize the Machine Learning modelling for the proposed system, the parameters, building, fitting, predicting and comparing with the best algorithm are done and implemented without human touch. Once, the ML modelling is done a pkl file is made available to download containing the best-performed machine learning model.

The forecasting is based on pkl file generated by AutoML. In this, the user needs to upload the same dataset used on the “Upload the dataset menu” and forecast the scores. In this, an additional row is appended to show the row-wise scores. A test case file will be available to download.

Thus, the proposed AutoML system implements the Machine Learning modelling in an automated way where the user has to execute the entire process by uploading the dataset and clicking on the radio buttons to make a perfect Machine Learning model as well as a facility to forecast and create a test cases file. The proposed AutoML handles EDA, Model building, considering parameters for hyperparameter tuning, comparing all the possible machine learning models, describing the best model, downloading the best model and forecasting all on its own with minimal or no human interference.

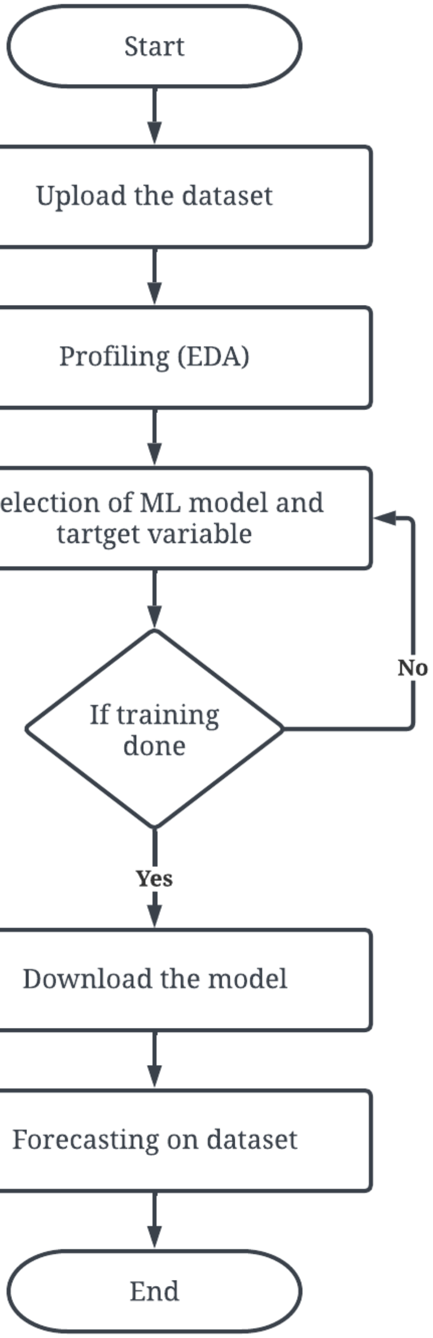
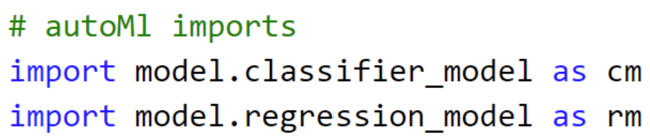


Fig. 2. The entire execution of Automl.ai

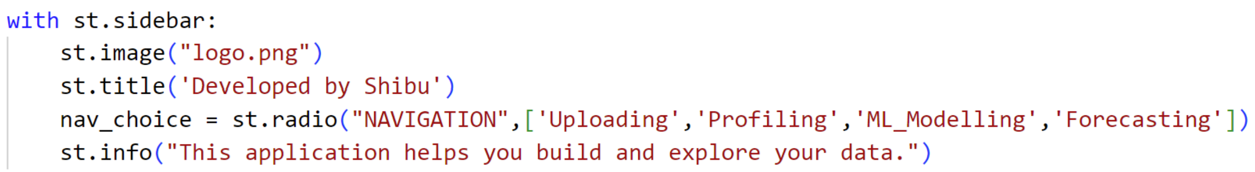
3.1. Core Code Segments



```
# autoMl imports
import model.classifier_model as cm
import model.regression_model as rm
```

Fig. 3. Separate models folder ML modelling

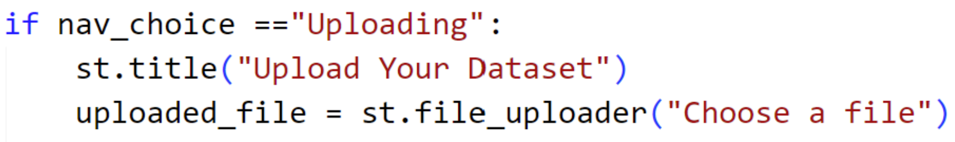
Here, the proposed system has a dedicated Machine Leaning modelling folder to make the algorithm easier to understand and cluttered free. Classification and Regression are two separate models are defined in the proposed AutoML system.



```
with st.sidebar:
    st.image("logo.png")
    st.title('Developed by Shibu')
    nav_choice = st.radio("NAVIGATION",['Uploading','Profiling','ML_Modelling','Forecasting'])
    st.info("This application helps you build and explore your data.")
```

Fig. 4. Sidebar

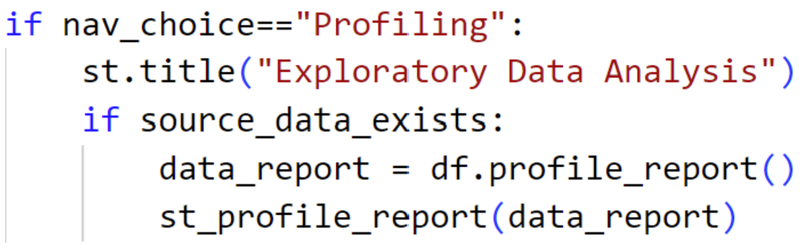
The sidebar contains all the radio buttons for navigating and developing the Machine Learning model. The navigation contains 4 button that is (a) Uploading – to upload the dataset. (b) Profiling – to perform EDA and feature engineering. (c) ML Modelling – To prepare and download a Machine Learning model. (d) Forecasting – To predict the scores on the dataset.



```
if nav_choice =="Uploading":
    st.title("Upload Your Dataset")
    uploaded_file = st.file_uploader("Choose a file")
```

Fig. 5. Uploading of dataset

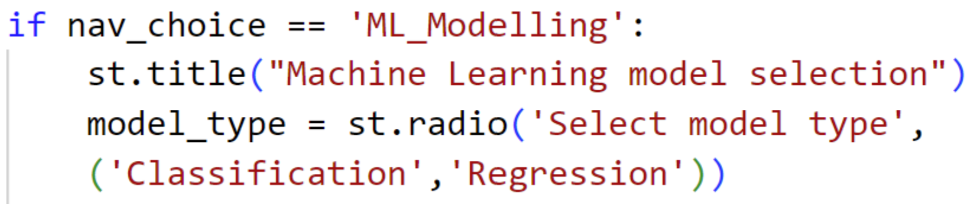
The user can upload the dataset in the form of csv or xlsx format.



```
if nav_choice=="Profiling":
    st.title("Exploratory Data Analysis")
    if source_data_exists:
        data_report = df.profile_report()
        st_profile_report(data_report)
```

Fig. 6. Profiling button

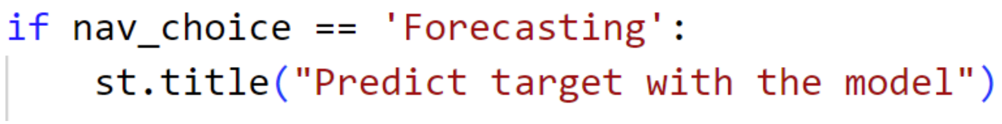
The profiling consists of Data Exploration, Data preparation, Feature Engineering along with graphs and figures. The following button creates a report of the dataset with all the available EDA techniques mentioned.



```
if nav_choice == 'ML_Modelling':
    st.title("Machine Learning model selection")
    model_type = st.radio('Select model type',
    ('Classification','Regression'))
```

Fig. 7. Machine Learning modelling and model selection

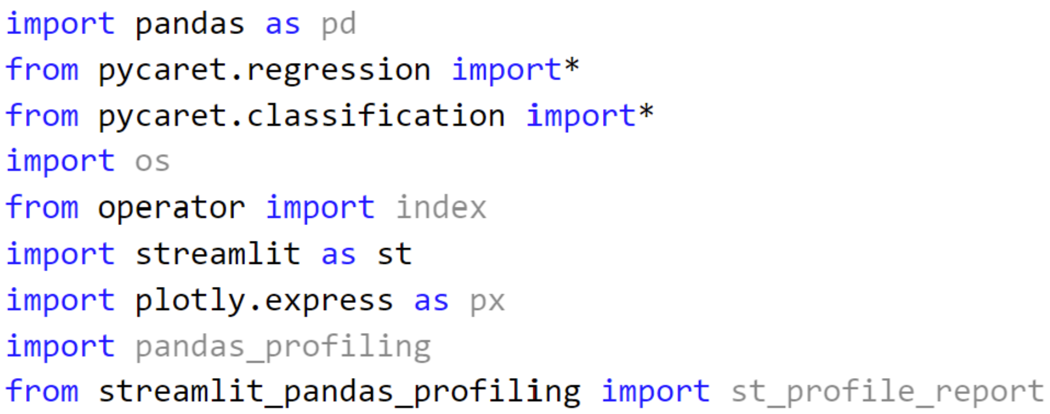
The user needs to click on the type of model to train the dataset on. Here, each of the model is connected to its dedicated model file to proceed with the model building and prediction.



```
if nav_choice == 'Forecasting':
    st.title("Predict target with the model")
```

Fig. 8. Forecasting phase

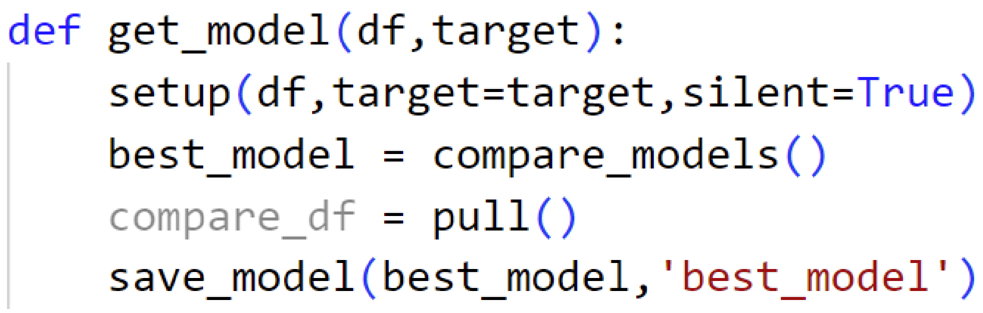
Here, the forecasting is based on the pkl generated during Model building. In forecasting the same dataset is upload and predict the target outcome of the model. It will show the accuracy of that target outcome row-wise.



```
import pandas as pd
from pycaret.regression import*
from pycaret.classification import*
import os
from operator import index
import streamlit as st
import plotly.express as px
import pandas_profiling
from streamlit_pandas_profiling import st_profile_report
```

Fig. 9. Libraries for Machine Learning models

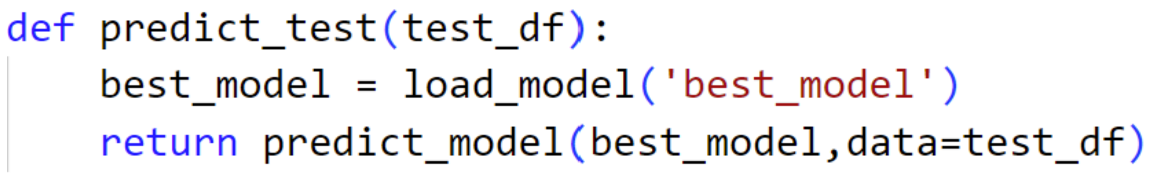
Libraries used for Machine Learning modelling and entire automl.ai process and execution. Here pandas and pandas profiling are used for data cleaning, data exploration, feature engineering, data transformation and data preparation. PyCaret is used for achine learning workflow and accelerate the development of machine learning models. Plotly is used for data visualization and provide an interactive and dynamic way to display data. Streamlit is used to simplify the process of building and deploying web applications.



```
def get_model(df,target):
    setup(df,target=target,silent=True)
    best_model = compare_models()
    compare_df = pull()
    save_model(best_model,'best_model')
```

Fig. 10. Machine Learning model

The above source code executes machine learning on the dataset and provides a comparison of all the ML models and make a pkl file of the model having best performance and accuracy. Here, the function calls the DataFrame and the target column specified by the user to train the model and compare the best performing model.



```
def predict_test(test_df):
    best_model = load_model('best_model')
    return predict_model(best_model,data=test_df)
```

Fig. 11. Testing the dataset

The above source code is used for forecasting where it needs to predict the outcome accuracy in the row wise format. This is exclusively for forecasting in which the prediction is based on the pkl file generated during ML modelling.

3.2. Requirements

- The necessities were gathered from approved sources.
- The necessities for improvement of a site are taken into account at each progression of the venture.
- The away type of data has led to extraordinary help with the overall prosperity of the task.

3.2.1. Hardware Requirements

The choice of hardware is incredibly important within the existence and proper working of any software. The hardware is a crucial factor where the user gives its to input to the software or program. When selecting hardware, size and requirements are important.

Processor	64-bit, 4 cores (minimum), 1.8 GHz minimum per core
RAM	16 GB for developer and evaluation use
Hard Disk Drive	512 GB
Hard Disk	3 GB for installation. For production use, you need additional free disk space for day-to-day deployment. Add twice as much free space as you have RAM for production environments.
Graphics Card	Nvidia, AMD

Table 1: Hardware Requirements

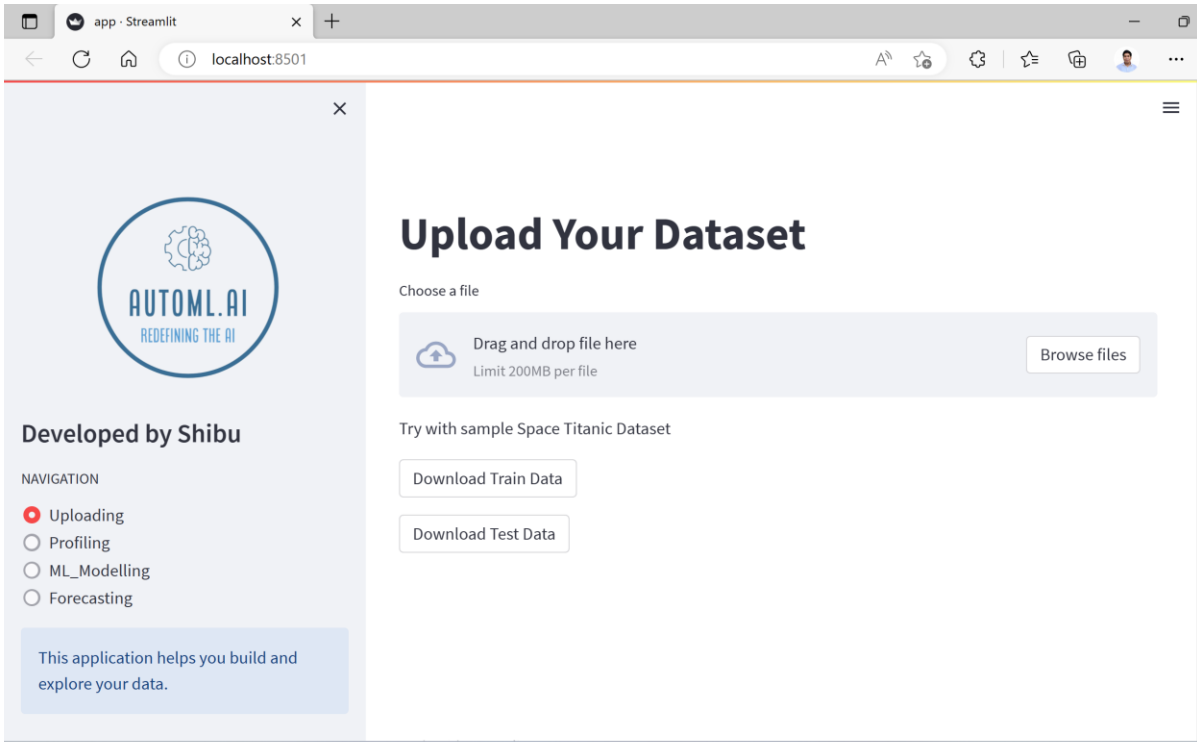
3.2.2. Software Requirements

Programming Requirements manage characterizing programming asset necessities and essentials that ought to be introduced on a PC to provide ideal working of an application. These necessities or pre-essentials are commonly excluded from the merchandise establishment bundle and will be introduced independently before the merchandise is introduced.

Tools	**Description**
Realtime Operating System	Windows 10, 11 home or pro, MacOS, Linux
Programming Languages	Python
Application	VsCode, Anaconda, Jupyter Notebook
DevOps	GitHub, Render, StreamLit
Web Browser	Microsoft Edge, Safari, Google Chrome, etc.

Table 2: Software Requirements

3.3. Results and Screenshots



Developed by Shibu

This application helps you build and explore your data.

Upload Your Dataset

Choose a file

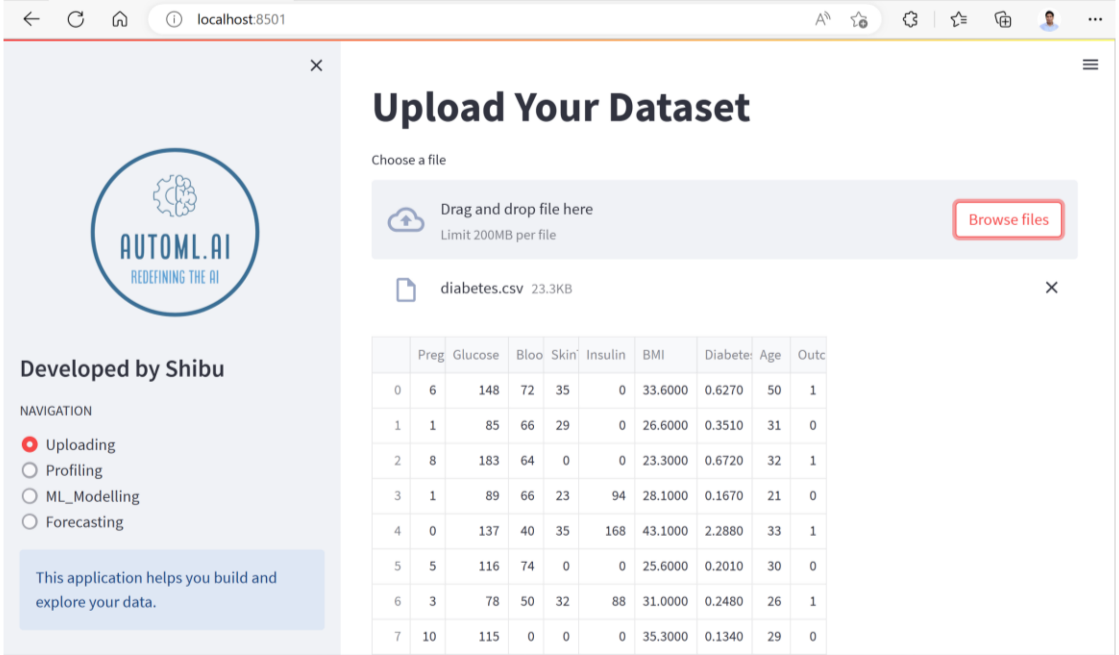
Drag and drop file here

Limit 200MB per file

Try with sample Space Titanic Dataset

Fig. 12. Home page

The home page of the application has all the things at one place, on the left pane the user input buttons are present and on the right side all the execution takes place.



Developed by Shibu

This application helps you build and explore your data.

Upload Your Dataset

Choose a file

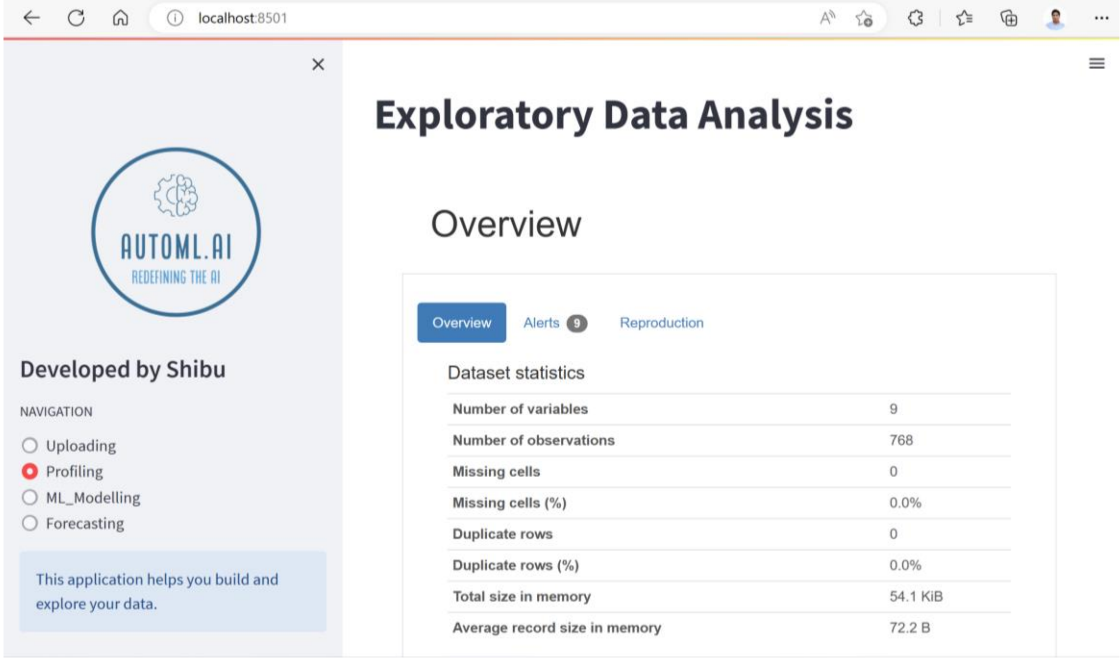
Drag and drop file here
Limit 200MB per file

diabetes.csv 23.3KB

	Preg	Glucose	Bloo	Skin	Insulin	BMI	Diabete	Age	Outc
0	6	148	72	35	0	33.6000	0.6270	50	1
1	1	85	66	29	0	26.6000	0.3510	31	0
2	8	183	64	0	0	23.3000	0.6720	32	1
3	1	89	66	23	94	28.1000	0.1670	21	0
4	0	137	40	35	168	43.1000	2.2880	33	1
5	5	116	74	0	0	25.6000	0.2010	30	0
6	3	78	50	32	88	31.0000	0.2480	26	1
7	10	115	0	0	0	35.3000	0.1340	29	0

Fig. 13. Uploading Dataset

The first phase starts with uploading of dataset and displaying few rows of the dataset. The user can also execute the entire process with a demo dataset by clicking on download Data button.



Developed by Shibu

This application helps you build and explore your data.

Exploratory Data Analysis

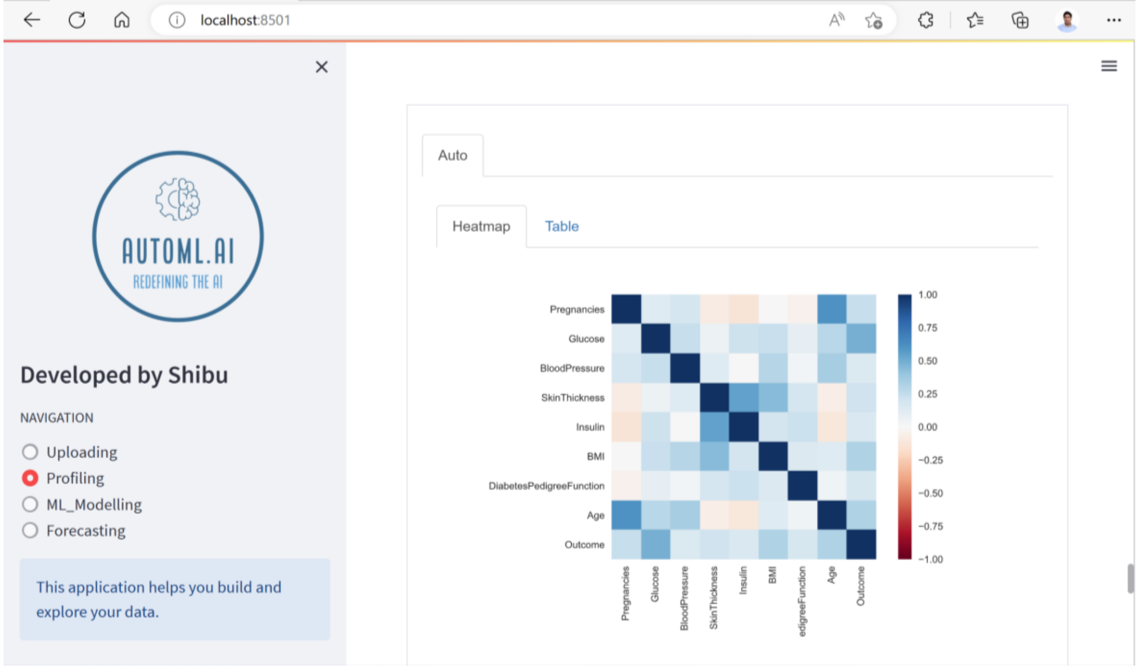
Overview

Dataset statistics

Number of variables	9
Number of observations	768
Missing cells	0
Missing cells (%)	0.0%
Duplicate rows	0
Duplicate rows (%)	0.0%
Total size in memory	54.1 KiB
Average record size in memory	72.2 B

Fig. 14. Profiling part

The profiling part handles data cleaning, feature engineering and all the Exploratory Data Analysis.



Developed by Shibu

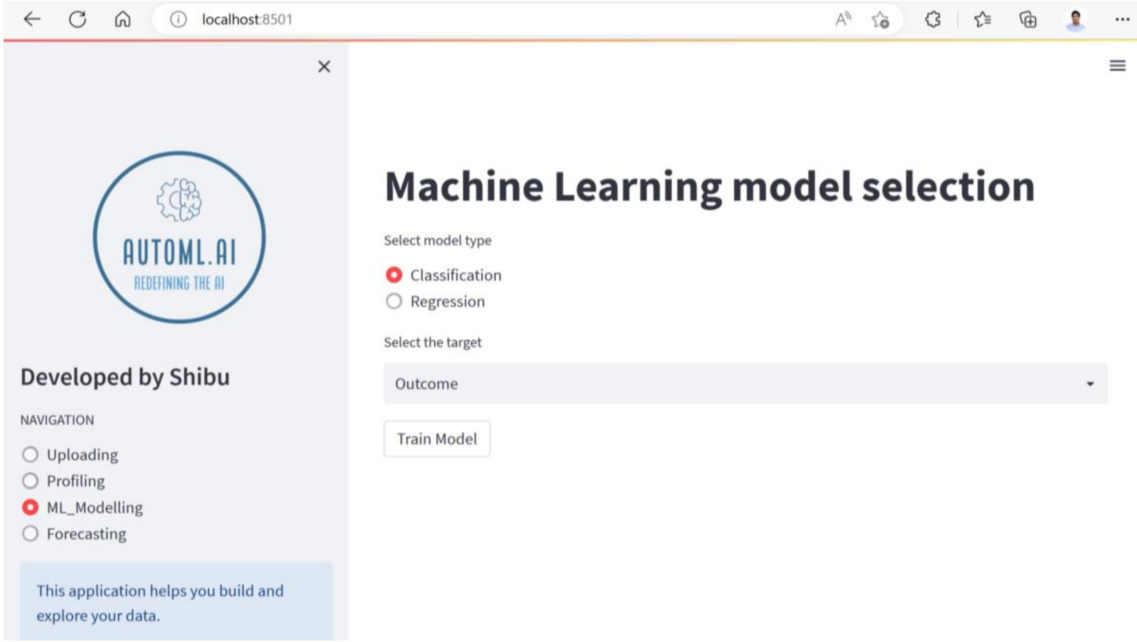
NAVIGATION

○ Uploading
● Profiling
○ ML_Modelling
○ Forecasting

This application helps you build and explore your data.

Fig. 15. A Heatmap generated during profiling

A heatmap generated by the Profiling (EDA) to show the corelation for the dataset.



Developed by Shibu

NAVIGATION

○ Uploading
○ Profiling
● ML_Modelling
○ Forecasting

This application helps you build and explore your data.

Machine Learning model selection

Select model type

● Classification
○ Regression

Select the target

Outcome

Fig. 16. Machine Learning Modelling

The next phase is the Machine Leaning Modelling, the user selects the model type (Regression or Classification) and select a target column from the drop-down menu. Once everything is selected the user can click on the train model button.

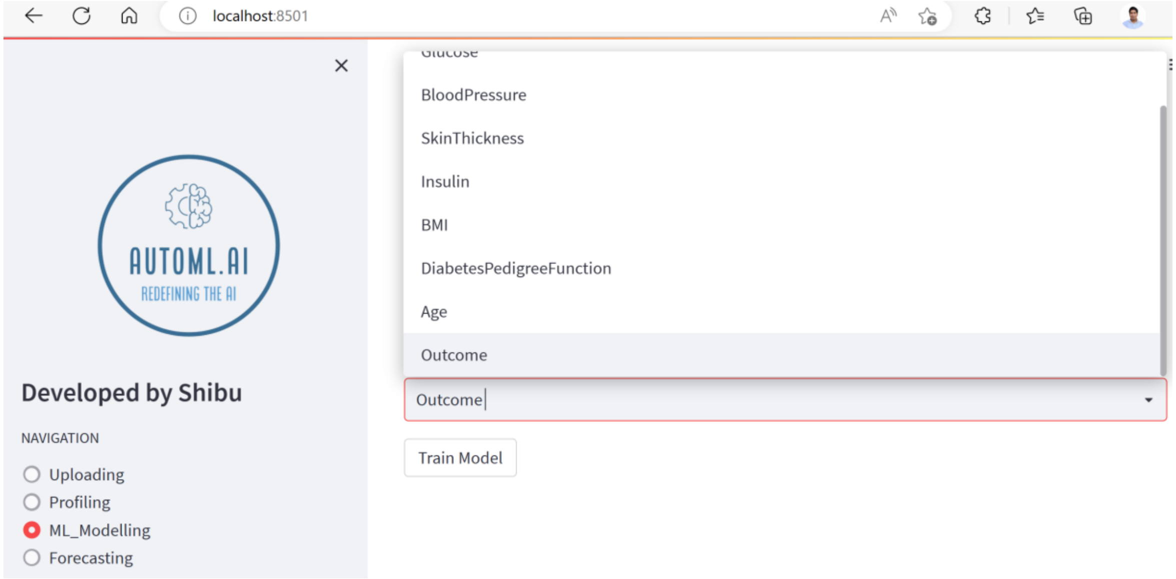
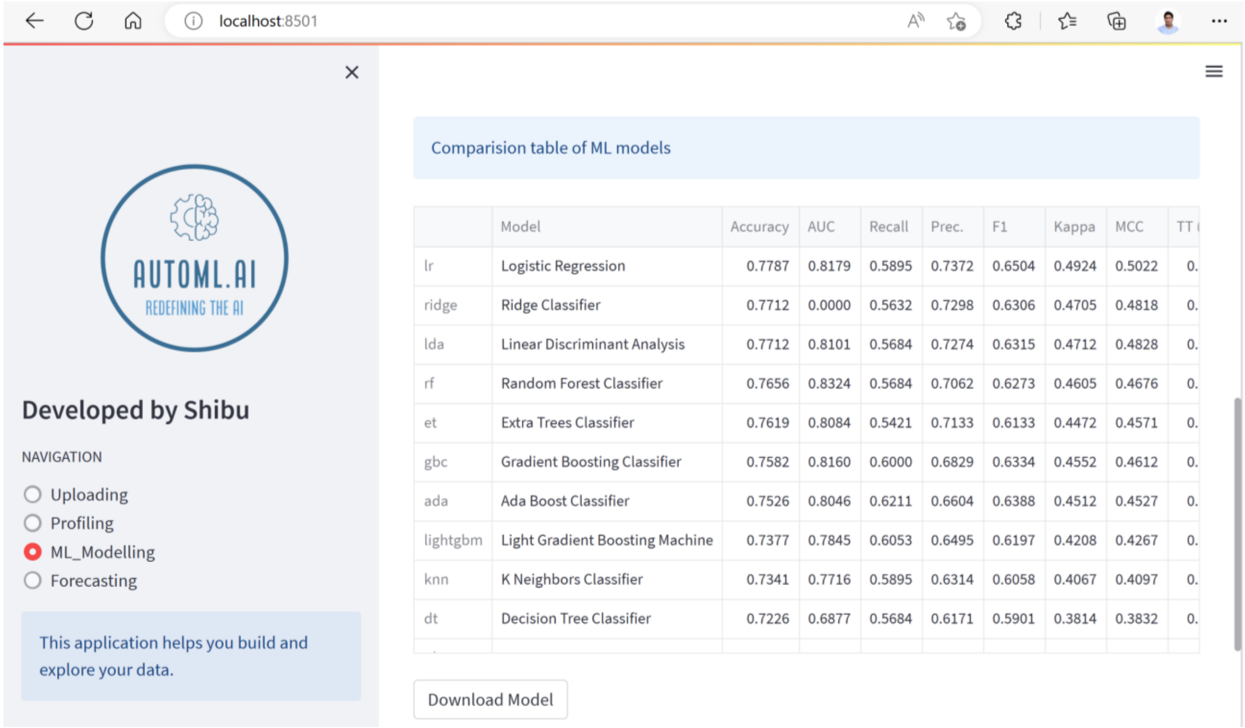


Fig. 17. Selection of target variable

Selection of target variable from the drop-down menu to begin with the model training phase.



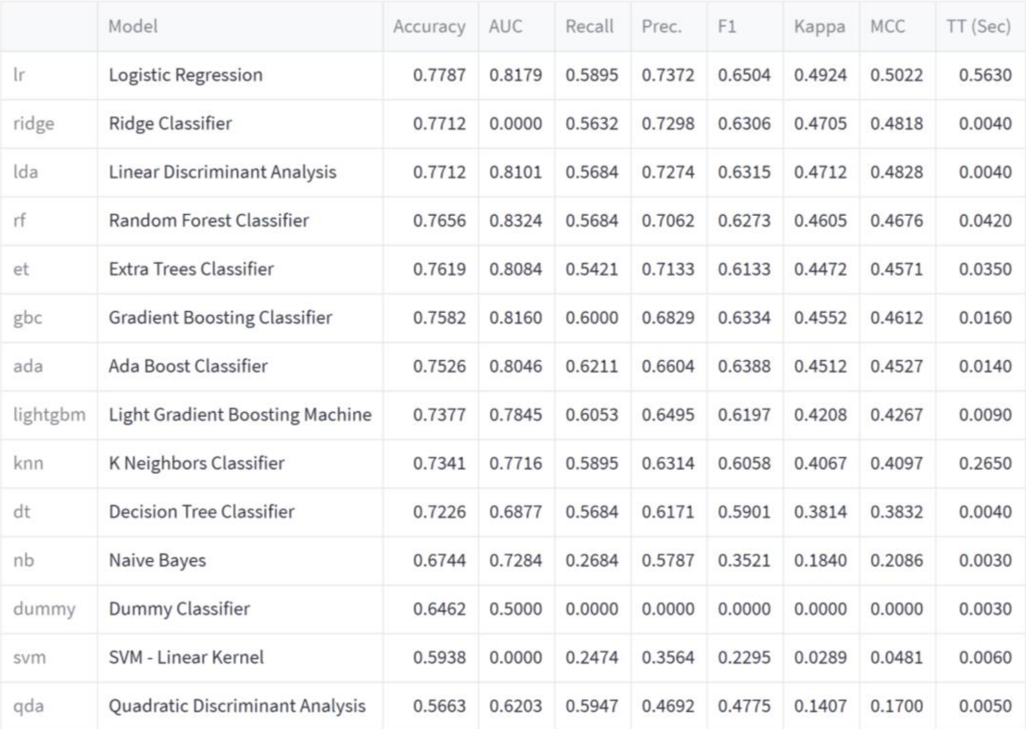
This application helps you build and explore your data.

Comparision table of ML models

	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC	TT (
lr	Logistic Regression	0.7787	0.8179	0.5895	0.7372	0.6504	0.4924	0.5022	0.
ridge	Ridge Classifier	0.7712	0.0000	0.5632	0.7298	0.6306	0.4705	0.4818	0.
lda	Linear Discriminant Analysis	0.7712	0.8101	0.5684	0.7274	0.6315	0.4712	0.4828	0.
rf	Random Forest Classifier	0.7656	0.8324	0.5684	0.7062	0.6273	0.4605	0.4676	0.
et	Extra Trees Classifier	0.7619	0.8084	0.5421	0.7133	0.6133	0.4472	0.4571	0.
gbc	Gradient Boosting Classifier	0.7582	0.8160	0.6000	0.6829	0.6334	0.4552	0.4612	0.
ada	Ada Boost Classifier	0.7526	0.8046	0.6211	0.6604	0.6388	0.4512	0.4527	0.
lightgbm	Light Gradient Boosting Machine	0.7377	0.7845	0.6053	0.6495	0.6197	0.4208	0.4267	0.
knn	K Neighbors Classifier	0.7341	0.7716	0.5895	0.6314	0.6058	0.4067	0.4097	0.
dt	Decision Tree Classifier	0.7226	0.6877	0.5684	0.6171	0.5901	0.3814	0.3832	0.

Fig. 18. Comparison table generated during ML modelling

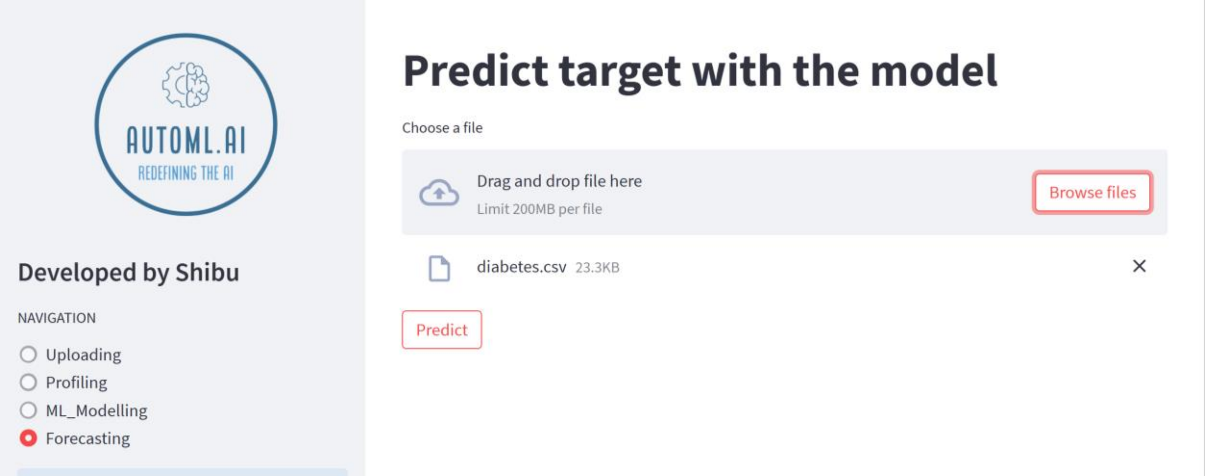
Once a model is trained succesfully, a table of comparison consisting all the algorithm and their performance. After that the user can download the model by clicking on Download button.



	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC	TT (Sec)
lr	Logistic Regression	0.7787	0.8179	0.5895	0.7372	0.6504	0.4924	0.5022	0.5630
ridge	Ridge Classifier	0.7712	0.0000	0.5632	0.7298	0.6306	0.4705	0.4818	0.0040
lda	Linear Discriminant Analysis	0.7712	0.8101	0.5684	0.7274	0.6315	0.4712	0.4828	0.0040
rf	Random Forest Classifier	0.7656	0.8324	0.5684	0.7062	0.6273	0.4605	0.4676	0.0420
et	Extra Trees Classifier	0.7619	0.8084	0.5421	0.7133	0.6133	0.4472	0.4571	0.0350
gbc	Gradient Boosting Classifier	0.7582	0.8160	0.6000	0.6829	0.6334	0.4552	0.4612	0.0160
ada	Ada Boost Classifier	0.7526	0.8046	0.6211	0.6604	0.6388	0.4512	0.4527	0.0140
lightgbm	Light Gradient Boosting Machine	0.7377	0.7845	0.6053	0.6495	0.6197	0.4208	0.4267	0.0090
knn	K Neighbors Classifier	0.7341	0.7716	0.5895	0.6314	0.6058	0.4067	0.4097	0.2650
dt	Decision Tree Classifier	0.7226	0.6877	0.5684	0.6171	0.5901	0.3814	0.3832	0.0040
nb	Naive Bayes	0.6744	0.7284	0.2684	0.5787	0.3521	0.1840	0.2086	0.0030
dummy	Dummy Classifier	0.6462	0.5000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0030
svm	SVM - Linear Kernel	0.5938	0.0000	0.2474	0.3564	0.2295	0.0289	0.0481	0.0060
qda	Quadratic Discriminant Analysis	0.5663	0.6203	0.5947	0.4692	0.4775	0.1407	0.1700	0.0050

Table 3: All classification models comparison

A comparison table of all the algorithms. The proposed system will make the Machine Learning model (.pkl file) based on best performed algorithm (the proposed system will make the model of Logistic Regression since it performed great).



Developed by Shibu

NAVIGATION

○ Uploading
○ Profiling
○ ML_Modelling
● Forecasting

Predict target with the model

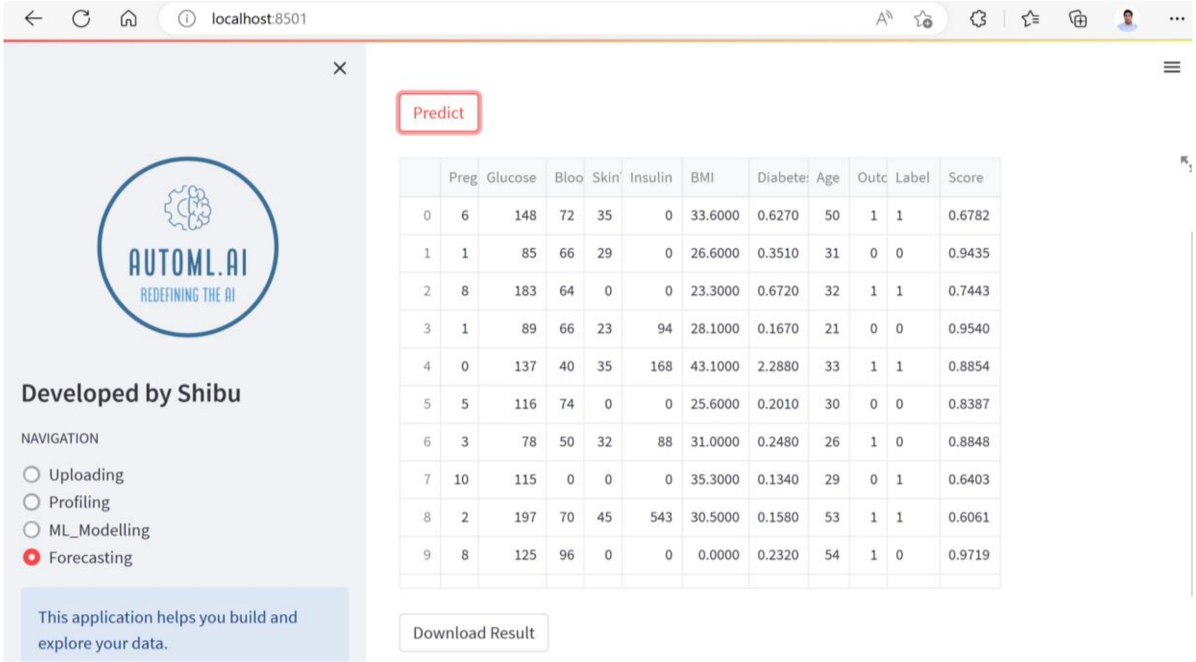
Choose a file

Drag and drop file here
Limit 200MB per file

diabetes.csv 23.3KB

Fig. 19. Forecasting prediction

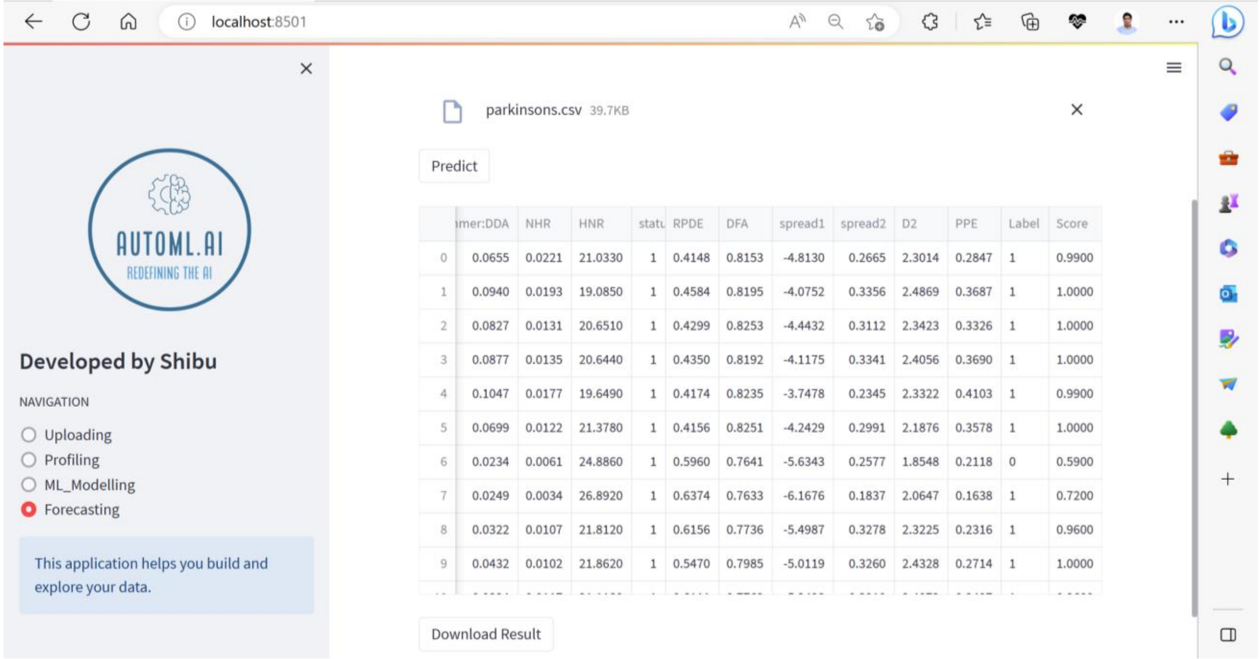
In forecasting phase, the user need to upload the same dataset it generated the .pkl for. The forecasting will generate a test cases csv file.



	Preg	Glucose	Bloo	Skin'	Insulin	BMI	Diabete:	Age	Outc	Label	Score
0	6	148	72	35	0	33.6000	0.6270	50	1	1	0.6782
1	1	85	66	29	0	26.6000	0.3510	31	0	0	0.9435
2	8	183	64	0	0	23.3000	0.6720	32	1	1	0.7443
3	1	89	66	23	94	28.1000	0.1670	21	0	0	0.9540
4	0	137	40	35	168	43.1000	2.2880	33	1	1	0.8854
5	5	116	74	0	0	25.6000	0.2010	30	0	0	0.8387
6	3	78	50	32	88	31.0000	0.2480	26	1	0	0.8848
7	10	115	0	0	0	35.3000	0.1340	29	0	1	0.6403
8	2	197	70	45	543	30.5000	0.1580	53	1	1	0.6061
9	8	125	96	0	0	0.0000	0.2320	54	1	0	0.9719

Fig. 20. Predicted scores (Diabetes model)

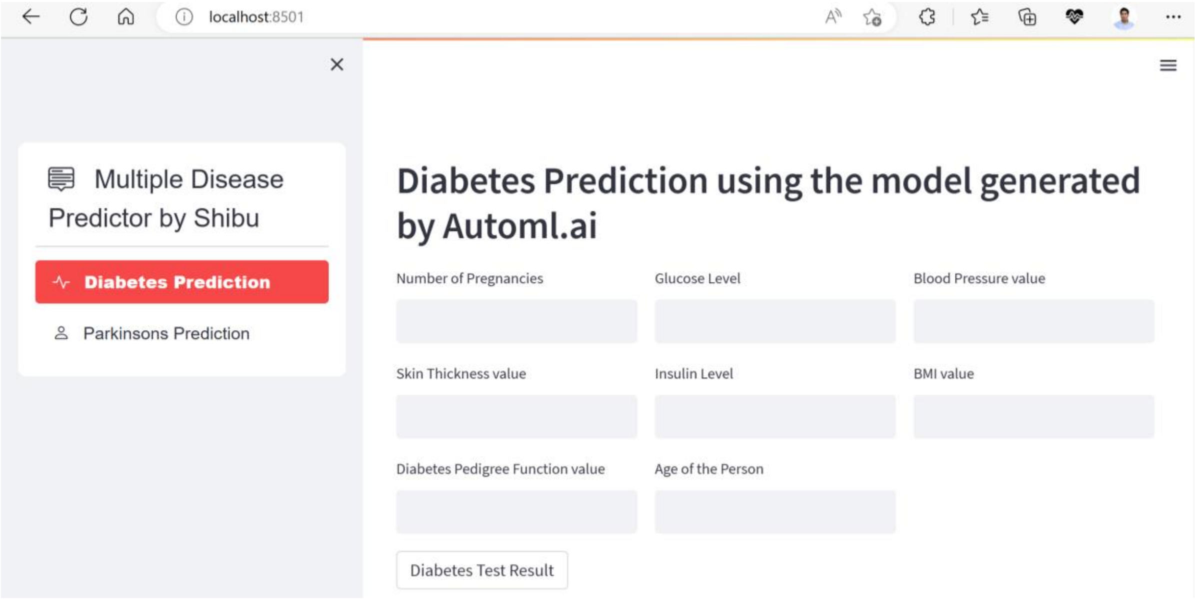
The forecasting will predict the score for each row in the dataset and create a test cases csv file.



	ımer:DDA	NHR	HNR	statu	RPDE	DFA	spread1	spread2	D2	PPE	Label	Score
0	0.0655	0.0221	21.0330	1	0.4148	0.8153	-4.8130	0.2665	2.3014	0.2847	1	0.9900
1	0.0940	0.0193	19.0850	1	0.4584	0.8195	-4.0752	0.3356	2.4869	0.3687	1	1.0000
2	0.0827	0.0131	20.6510	1	0.4299	0.8253	-4.4432	0.3112	2.3423	0.3326	1	1.0000
3	0.0877	0.0135	20.6440	1	0.4350	0.8192	-4.1175	0.3341	2.4056	0.3690	1	1.0000
4	0.1047	0.0177	19.6490	1	0.4174	0.8235	-3.7478	0.2345	2.3322	0.4103	1	0.9900
5	0.0699	0.0122	21.3780	1	0.4156	0.8251	-4.2429	0.2991	2.1876	0.3578	1	1.0000
6	0.0234	0.0061	24.8860	1	0.5960	0.7641	-5.6343	0.2577	1.8548	0.2118	0	0.5900
7	0.0249	0.0034	26.8920	1	0.6374	0.7633	-6.1676	0.1837	2.0647	0.1638	1	0.7200
8	0.0322	0.0107	21.8120	1	0.6156	0.7736	-5.4987	0.3278	2.3225	0.2316	1	0.9600
9	0.0432	0.0102	21.8620	1	0.5470	0.7985	-5.0119	0.3260	2.4328	0.2714	1	1.0000

Fig. 21. Predicted scores (Parkinson model)

The forecasting will predict the score for each row in the dataset and create a test cases csv file.



Diabetes Prediction using the model generated by Automl.ai

Number of Pregnancies:

Glucose Level:

Blood Pressure value:

Skin Thickness value:

Insulin Level:

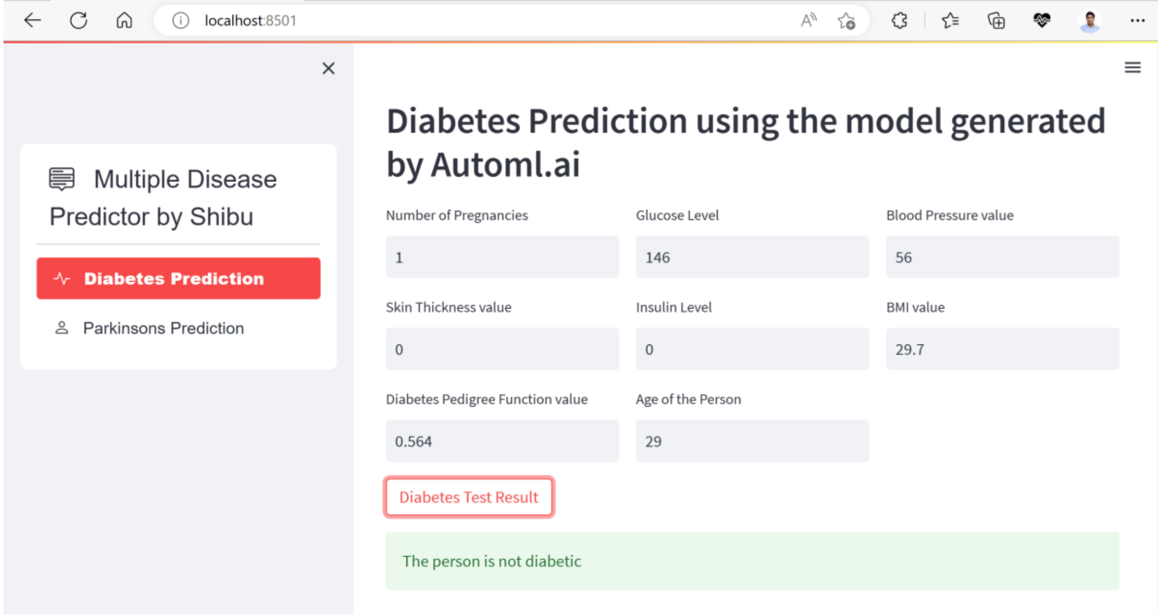
BMI value:

Diabetes Pedigree Function value:

Age of the Person:

Fig. 22. Multiple Disease prediction Application (Diabetes Prediction)

For validating the model generated by the proposed system automl.ai, I have tested it with the multiple disease predictor WebApp. Since the forecasting is works on train and test dataset, to see the working it on the unknown dataset, I will place the values for the diabetes.



Diabetes Prediction using the model generated by Automl.ai

Number of Pregnancies: 1

Glucose Level: 146

Blood Pressure value: 56

Skin Thickness value: 0

Insulin Level: 0

BMI value: 29.7

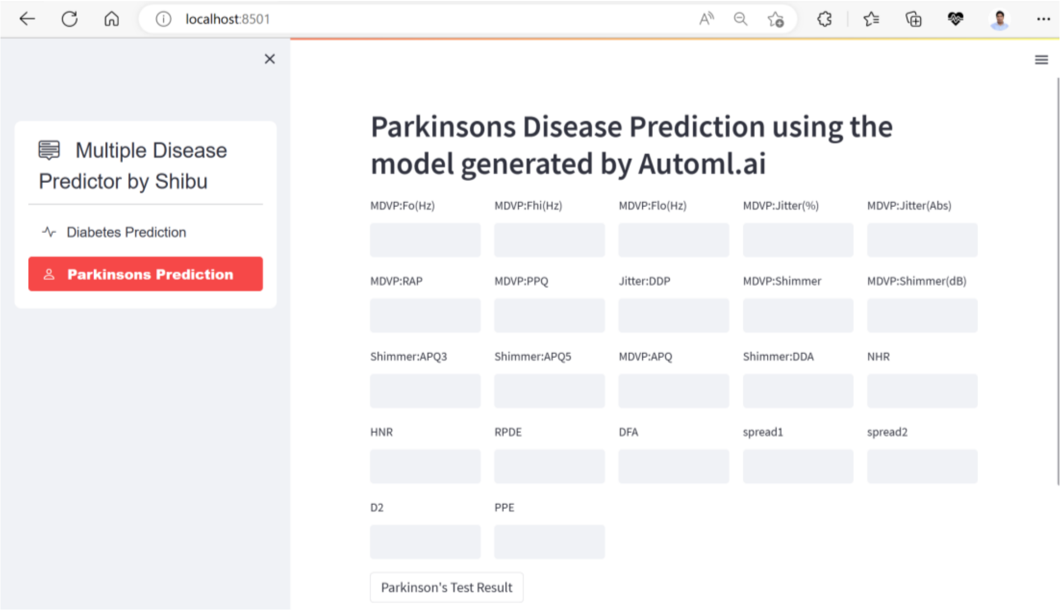
Diabetes Pedigree Function value: 0.564

Age of the Person: 29

The person is not diabetic

Fig. 23. A final output of prediction of a diabetic and non-diabetic person

The user will give its inputs to predict the diabetes. The model generated by the proposed system is able to detect the whether the person is diabetic or non-diabetic.

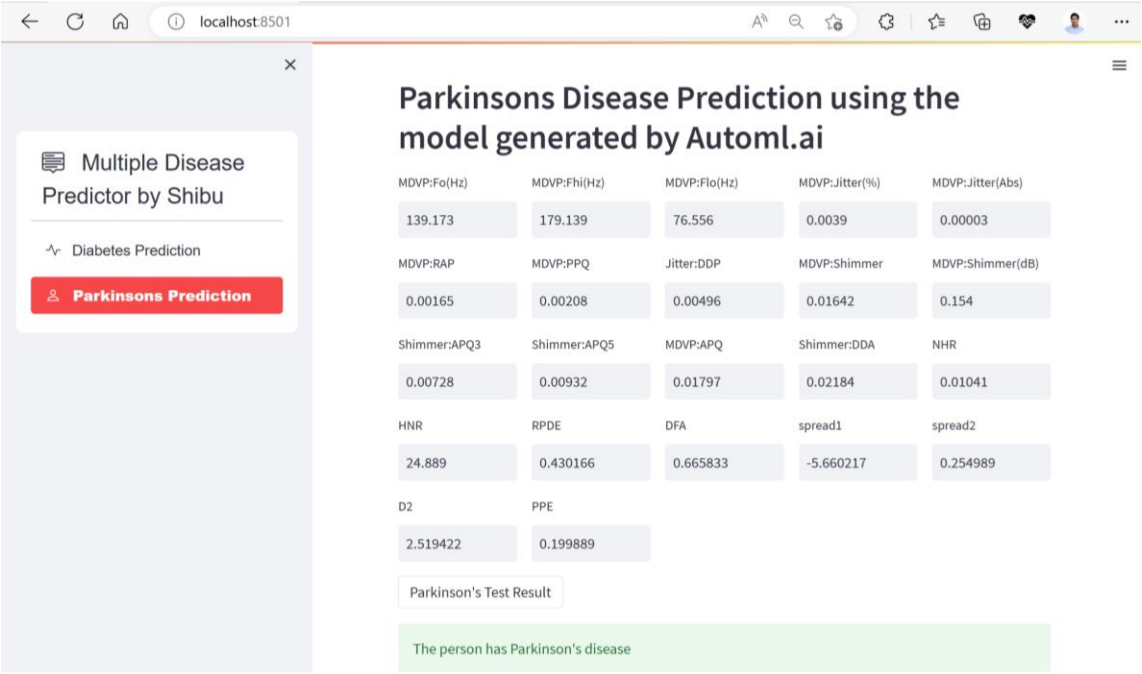


Parkinsons Disease Prediction using the model generated by Automl.ai

MDVP:Fo(Hz)	MDVP:Fhi(Hz)	MDVP:Flo(Hz)	MDVP:Jitter(%)	MDVP:Jitter(Abs)
MDVP:RAP	MDVP:PPQ	Jitter:DDP	MDVP:Shimmer	MDVP:Shimmer(dB)
Shimmer:APQ3	Shimmer:APQ5	MDVP:APQ	Shimmer:DDA	NHR
HNR	RPDE	DFA	spread1	spread2
D2	PPE			

Fig. 24. Multiple Disease prediction Application (Parkinson's Prediction)

The second test will be for Parkinson's prediction. Here I will be predicting by giving unknown data inputs. I will put the values for Parkinson disease.



Parkinsons Disease Prediction using the model generated by Automl.ai

MDVP:Fo(Hz)	MDVP:Fhi(Hz)	MDVP:Flo(Hz)	MDVP:Jitter(%)	MDVP:Jitter(Abs)
139.173	179.139	76.556	0.0039	0.00003
MDVP:RAP	MDVP:PPQ	Jitter:DDP	MDVP:Shimmer	MDVP:Shimmer(dB)
0.00165	0.00208	0.00496	0.01642	0.154
Shimmer:APQ3	Shimmer:APQ5	MDVP:APQ	Shimmer:DDA	NHR
0.00728	0.00932	0.01797	0.02184	0.01041
HNR	RPDE	DFA	spread1	spread2
24.889	0.430166	0.665833	-5.660217	0.254989
D2	PPE			
2.519422	0.199889			

The person has Parkinson's disease

Fig. 25. A final output of prediction of a person having Parkinson disease or not

The user will give its inputs to predict the disease. The model generated by the proposed system is able to detect the whether the person has Parkinson disease or not.

CHAPTER 4

CONCLUSION AND FUTURE SCOPE

Working on this project was a delightful experience, it has also got me familiar with new and advanced techniques as well as knowledge. I had understood the importance of Planning & Designing as a part of Machine Learning. But it's also very challenging for a person to complete the project single-handedly.

AutoML (Automated Machine Learning) is a growing field that focuses on automating the process of building and training machine learning models. The goal of AutoML is to simplify and speed up the development process of machine learning models, making it more accessible to users without a strong background in machine learning. AutoML has many benefits, including reducing the time and cost associated with developing machine learning models. It also enables non-experts to build and deploy models, which can help organizations make more informed decisions and gain insights from their data.

Overall, AutoML is a valuable tool for companies and organizations looking to quickly develop and deploy machine learning models. However, it should not be seen as a replacement for expert data scientists and machine learning engineers, who bring a deeper understanding of the nuances of machine learning models and can develop highly customized solutions.

The future work towards my system is that to make a progressive and dynamic interface for Deep Learning models and create an entire ETL pipeline for further Business Intelligence and Business decisions. Also, a setup for a GPU and cloud integration or API for faster processing for Deep Learning models and handle unstructured data. Lastly, to fine tune the system so well to handle semi-structured datasets (JSON and XML) and unstructured datasets (image, video and audio files).

REFERENCES

[1] “Automated Machine Learning (AutoML): an overview of opportunities for application and research” by Kailash Joshi (2022)

[2] "AutoML: A Survey of the State-of-the-Art" by Hutter et al. (2019)

[3] "Neural Architecture Search with Reinforcement Learning" by Zoph and Le (2017)

[4] “An Open Source AutoML Benchmark” by Pieter Gijsbers (ICML 2019)

[5] "Progressive Neural Architecture Search" by Liu et al. (2018)

[6] "AutoKeras: An Efficient Neural Architecture Search System" by Jin et al. (2019)

[7] "Learning Transferable Architectures for Scalable Image Recognition" by Real et al. (2018)

[8] "Bayesian Optimization for Automated Model Selection" by Snoek et al. (2012)

[9] "Automated Machine Learning: Methods, Systems, Challenges" by Feurer et al. (2019)

[10] "Auto-Sklearn: Efficient and Robust Automated Machine Learning" by Feurer. (2015)

[11] "MnasNet: Platform-Aware Neural Architecture Search for Mobile" by Tan et al. (2019)

ANNEXURE

(Note: All entries of the proforma of approval should be filled up with appropriate and complete information. Incomplete proforma of approval in any respect will be summarily rejected.)

GR No.: 3480531 Roll no: 02

1. Name of the Student: **MOHAPATRA SHIBU TRILOCHAN**
2. Title of the Project: **Automl.ai**
3. Name of the Guide: **Prof. Asmita Marathe**
4. Teaching/Industry experience of the Guide: **6.4 years**
5. Is this your first submission? **Yes**

Shibu

Signature of the Student: Signature of the Guide:

Date: ……………………. Date: …………………….

Signature of the Coordinator:

Date: …………………