Low Level Design (LLD)

Air Quality Index

Written By	Diana Laveena DSouza
Document Version	0.1
Last Revised Date	23-Jan-2023

Document Version Control

Change Record:

Version	Date	Author	Author
0.1	23/01/2023	Diana Laveena DSouza	Introduction & Architecture Defined

Reviews:

Version	Date	Reviewer	Comments
0.1	23/01/2023		Document Content, Version Control and Unit Test Cases to be added

Approval Status:

Version	Review Date	Reviewed By	Approved By	Comments

Contents

Abstract

Air quality for a human being and all the living species is a vital part of life. Variation of air quality creates a high impact on human life; continuous monitoring and evaluation and investigation of AIR quality are critical. Based on that, we are supposed to take and action and control all the activities in every sense. To achieve this purpose, we have to build a model that will predict the quality of air for various reasons. I will be able to generate an alarm for the same to create a regulation and guidelines for industries and the general public. We can try to inform them about various dangerous gas emissions and their impact on human life.

1 Introduction

1.1 Why is Low-Level Design Document?

The goal of LLD or a low-level design document (LLDD) is to give the internal logical design of the actual program code for Facebook Status Prediction. LLD describes the class diagrams with the methods and relations between classes and program specs. It describes the modules so the programmer can directly code the program from the document.

1.2 Scope

Low-level design (LLD) is a component-level design process that follows a step-by-step refinement process. This process can be used for designing data structures, required software architecture, source code and ultimately, performance algorithms. Overall, the data organization may be defined during requirement analysis and then refined during data design work.

2 Technical specifications

2.1 Dataset

DataSet	Finalized	Source
Air_Quality_22_23 .csv	yes	https://airquality.cpcb.gov.in/ccr/#/caaqm-dashboard-all/caaqm-landing/data

2.1.1 Dataset Overview

The Dataset consists of State, City, Station, From Date, To Date, PM2.5, PM10, NO2, NH3, CO, SO2, Ozone, AQI, AQI_Bucket.

There are a total of 6276 records in the training set and 602 in the validation set after applying oversampling technique.

Air_Quality_22_23

State	City	Station	From Date	To Date	PM2.5	PM10	NO2	SO2	CO	Ozone	NH3	AQI	AQI_Bucket
Andhra Pr	Amaravati	Secretaria	01-01-2023	02-01-2023 00:00	41.58	79	9.6	11.52	0.58	29.93	16.43	79	Satisfactory
Andhra Pr	Amaravati	Secretaria	02-01-2023	03-01-2023 00:00	50.13	79	10.76	10.69	0.63	31.77	15.68	84	Satisfactory
Andhra Pr	Amaravati	Secretaria	03-01-2023	04-01-2023 00:00	49.76	79	13.58	4.67	0.61	33.05	14.49	83	Satisfactory
Andhra Pr	Amaravati	Secretaria	04-01-2023	05-01-2023 00:00	52.78	79	10.21	8.59	0.57	45.77	16.05	88	Satisfactory
Andhra Pr	Amaravati	Secretaria	05-01-2023	06-01-2023 00:00	80.26	106.04	14.63	9.16	0.77	50.11	14.42	168	Moderate
Andhra Pr	Amaravati	Secretaria	06-01-2023	07-01-2023 00:00	102.92	156.46	14.75	12.84	1	63.11	14.4	243	Poor
Andhra Pr	Amaravati	Secretaria	07-01-2023	08-01-2023 00:00	99.66	159.26	18.35	14.52	1	44.36	13.42	232	Poor
Andhra Pr	Amaravati	Secretaria	08-01-2023	09-01-2023 00:00	71.79	126.02	16.59	12.28	0.76	41.2	13.79	139	Moderate
Andhra Pr	Amaravati	Secretaria	09-01-2023	10-01-2023 00:00	74.62	126.9	21.58	19.87	0.82	48.27	11.88	149	Moderate
Andhra Pr	Amaravati	Secretaria	10-01-2023	11-01-2023 00:00	73.68	131.66	19.72	17.17	0.82	49.8	11.92	146	Moderate
Andhra Pr	Amaravati	Secretaria	11-01-2023	12-01-2023 00:00	73.31	130.15	23.1	22.59	0.76	50.97	12.18	144	Moderate
Andhra Pr	Amaravati	Secretaria	12-01-2023	13-01-2023 00:00	59.96	97.13	16.1	10.81	0.83	45.41	14.1	100	Satisfactory
Andhra Pr	Amaravati	Secretaria	13-01-2023	14-01-2023 00:00	48.18	78.31	13.63	13.54	0.69	36.63	14.68	80	Satisfactory
Andhra Pr	Amaravati	Secretaria	14-01-2023	15-01-2023 00:00	44.09	72.33	13.78	8.76	0.9	38.97	14.85	73	Satisfactory
Andhra Pr	Amaravati	Secretaria	15-01-2023	16-01-2023 00:00	36.06	57.7	10.46	9.44	0.72	36	16.23	60	Satisfactory
Andhra Pr	Amaravati	Secretaria	16-01-2023	17-01-2023 00:00	36	55.22	10.29	9.31	0.59	36.39	16.32	60	Satisfactory
Andhra Pr	Amaravati	Secretaria	17-01-2023	18-01-2023 00:00	41.2	62.76	10.42	6.93	0.62	34.95	16.06	69	Satisfactory
Andhra Pr	Amaravati	Secretaria	18-01-2023	18-01-2023 00:00	41	73	13.4	8.7	0.61	18	15.1	73	Satisfactory
Andhra Pr	Anantapu	Gulzarpet	01-01-2023	02-01-2023 00:00	16.76	36.4	27.2	7.02	0.77	65.28	9.7	65	Satisfactory
Andhra Pr	Anantapu	Gulzarpet	02-01-2023	03-01-2023 00:00	14.86	32.79	19.58	7.14	0.71	59.7	10.4	60	Satisfactory
Andhra Pr	Anantapu	Gulzarpet	03-01-2023	04-01-2023 00:00	14.4	32.04	23.17	6.81	0.66	48.88	9.85	49	Good
Andhra Pr	Anantapu	Gulzarpet	04-01-2023	05-01-2023 00:00	16.83	36.36	22.89	6.14	0.67	54.18	9.84	54	Satisfactory

2.1.2 Input schema

Feature Name	Datatype	Size	Null/Required
City	varchar	20	Not Required
From Date	date		Not Required
PM2.5	double		Not Required
PM10	double		Not Required
NO2	double		Not Required
NH3	double		Not Required
CO	double		Not Required
SO2	double		Not Required
O3	double		Not Required
State	varchar	20	Not Required
Station	varchar	20	Not Required
To Date	date		Not Required
AQI	double		Not Required
AQI_Bucket	varchar	20	Not Required

2.2 Predicting Air Quality Index

- The system presents the set of inputs required from the user.
- The user gives the required information.
- The System should be able to predict the exact Air Quality Index based on the user information.

2.3 Database

The database used in this project to store the data is the Cassandra database.

2.5 Deployment

3 Technology Stack

Front End	HTML/CSS
Backend	Python Flask
Database	Cassandra
Deployment	GCP

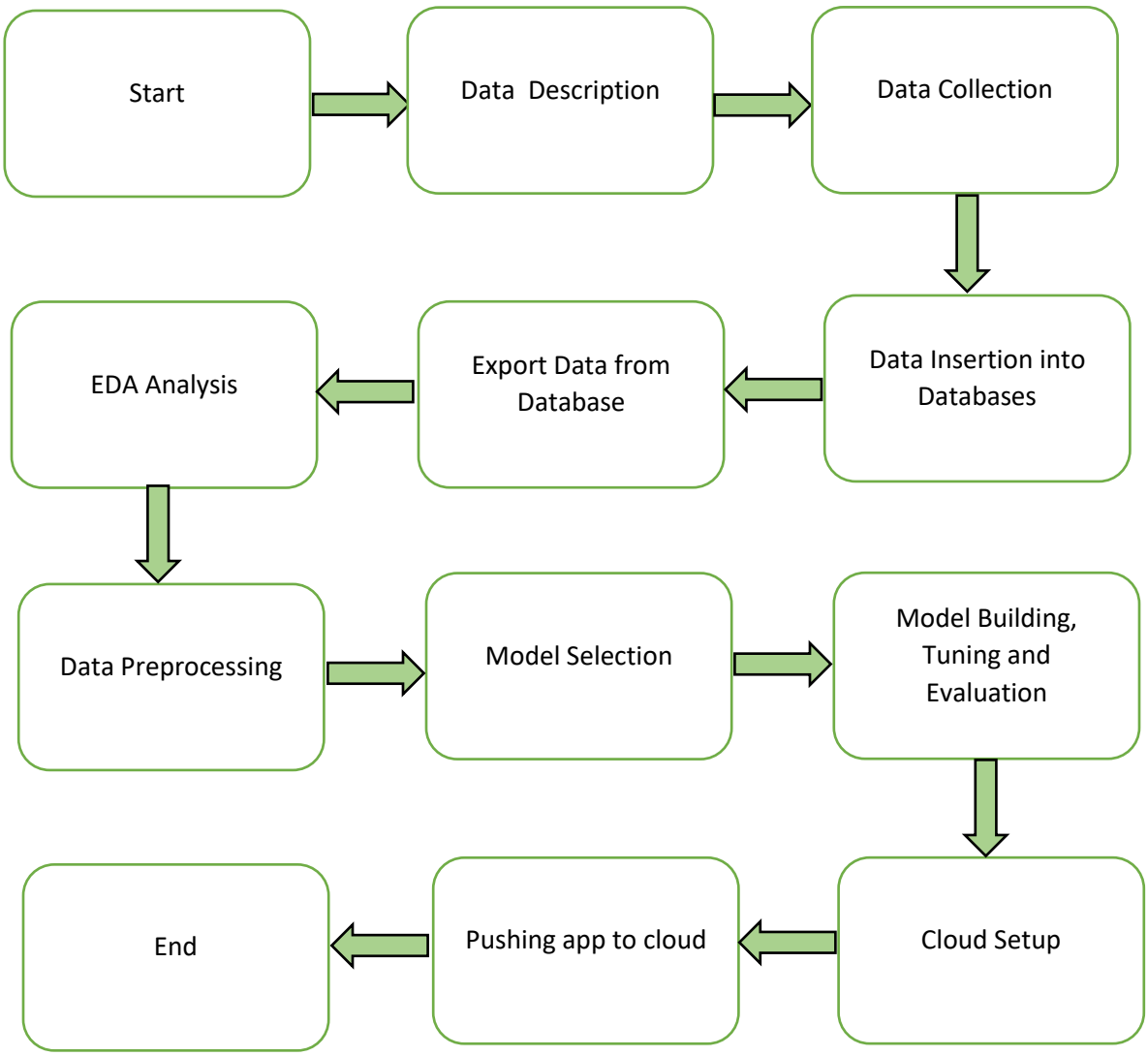
4 Proposed Solution

Refer: https://link.springer.com/article/10.1007/s13762-022-04241-5#:~:text=The%20ML%2Dbased%20AQI%20prediction,algorithms%20can%20deal%20with%20efficiently

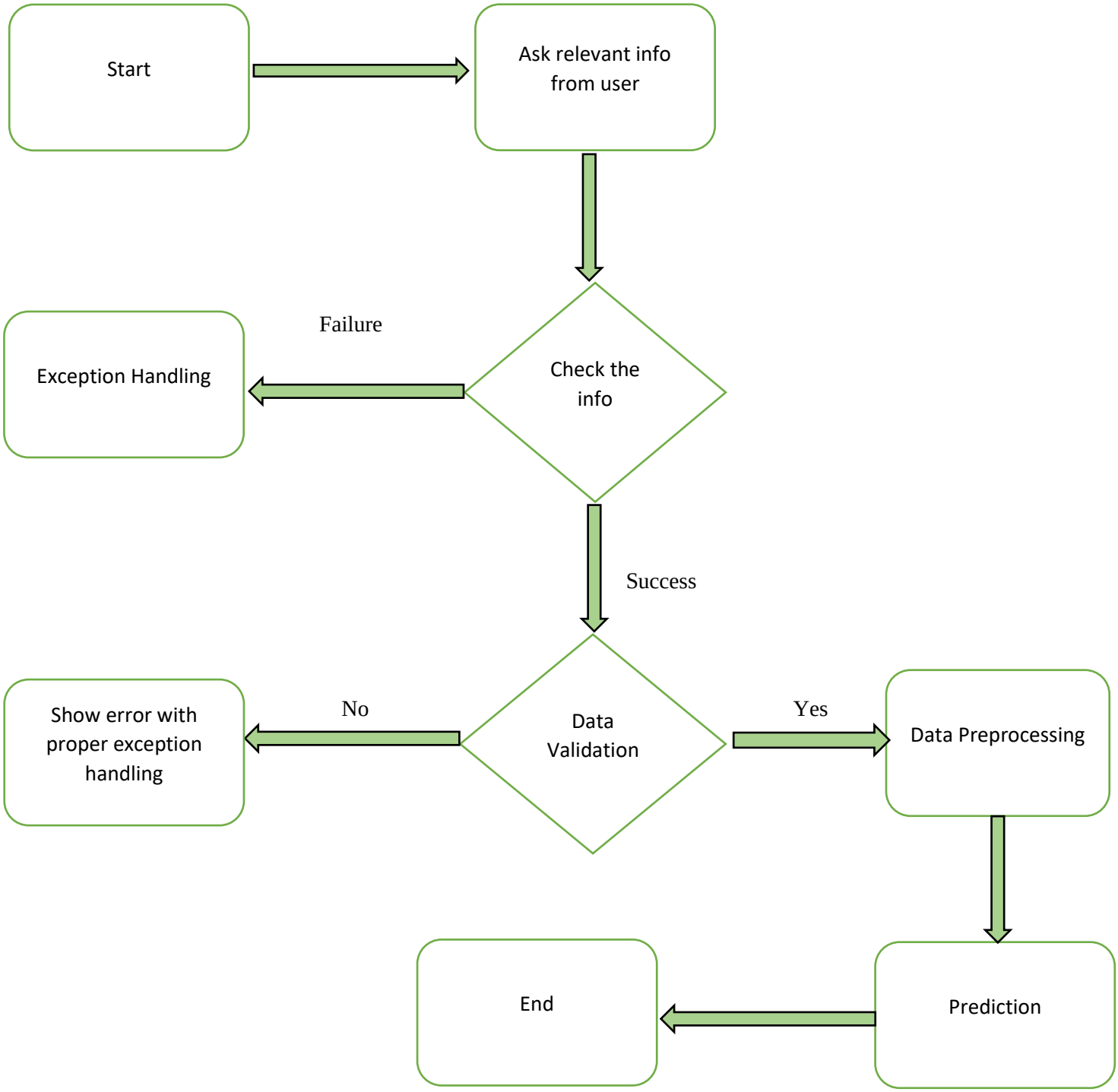
The actual research paper proposed the method of Machine Learning with oversampling techniques. Finally, we used the same methods. Under Machine learning-based methods, we choose Supervised. We decided on Supervised methods such as tree-based methods, K-nearest neighbour and SVR. But the tree-based method Random Forest gives better results than others.

Baseline Model: Random Forest, since this is a regression problem.

5 Model Training/Validation Workflow



6 User I/O workflow



7 Test Cases

Test Case Description	Pre-Requisite	Expected Result
Verify whether the Application URL is accessible to the user	1. Application URL should be defined	The application URL should be accessible to the user.
Verify whether the Application loads entirely for the user when the URL is accessed	1. Application URL is accessible 2. Application is deployed	The Application should load entirely for the user when the URL is accessed.
Verify whether the user can input the text in all input fields	1. Application is accessible	The user should be able to input the text in all input fields.
Verify whether the user gets Submit button to submit the inputs.		The user should get Submit button to submit the inputs.
Verify whether the user is presented with results on clicking submit.		The user should be presented with results on clicking submit