High Level Design (HLD)

Credit Card Default Prediction

Contents

Abstract:

The Banking Sector is dependent on the loan they provide to people. The interest paid by the people make it possible for banks to heir staff and provide various facility to the customers. The loan are provided in various forms by banks. One of the most popular way are credit cards. In credit cards user can spend a certain amount decided by the banks without having any actual money spend from their bank accounts and can then pay the bill in next month. User with credit card starts defaulting their payment it creates a problem. Hence predicting if the person is going to default or not is a good way to make a rational decision.

1   Introduction

1.1 Why this High-Level Design Document?

The purpose of this High-Level Design (HLD) Document is to add the necessary detail to the current project description to represent a suitable model for coding. This document is also intended to help detect contradictions prior to coding, and can be used as a reference manual for how the modules interact at a high level.

The HLD will:

- Present all of the design aspects and define them in detail
- Describe the user interface being implemented
- Describe the hardware and software interfaces
- Describe the performance requirements
- Include design features and the architecture of the project
- List and describe the non-functional attributes like:

    - Security
    - Reliability
    - Maintainability
    - Portability
    - Reusability
    - Application compatibility
    - Resource utilization
    - Serviceability

1.2 Scope

The HLD documentation presents the structure of the system, such as the database architecture, application architecture (layers), application flow (Navigation), and technology architecture. The HLD uses non-technical to mildly-technical terms which should be understandable to the administrators of the system.

2 General Description

2.1 Product Perspective

The credit card default prediction is a Machine Learning based project. It help in prediction of the people who will be going the default their payments

2.2 Problem statement

To find the person who is going to default their payments by seeing their payment behaviours and finding insights

2.3 PROPOSED SOLUTION

We need to see the behaviour pattern of the person regarding their payment of credit card bills and after seeing them we can make a prediction model and find whether they will default or not.

2.4 FURTHER IMPROVEMENTS

The credit cards prediction cannot be trusted completely also the credit score is also a important factor which is not used or considered in the model hence if we take credit score as a input parameter it can be used better.

2.5 Data Requirements

Data requirement completely depend on our problem statement.

2.6 Tools used

Python programming language and frameworks such as NumPy, Pandas, Scikit-learn are used to build the whole model. PyCharm is used as IDE. Heroku is used for the deployment of the project. Front end development is done using HTML and CSS. Python Flask is used for backend development. GitHub is used as version control system.
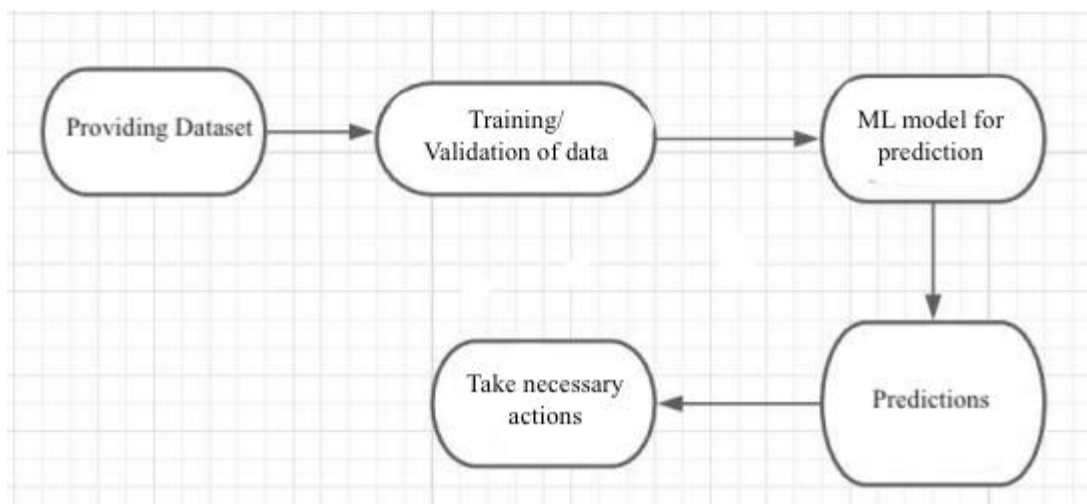

2.7 Constraints

The credit card default prediction system must be user friendly, as automated as possible and users should not be required to know any of the workings.
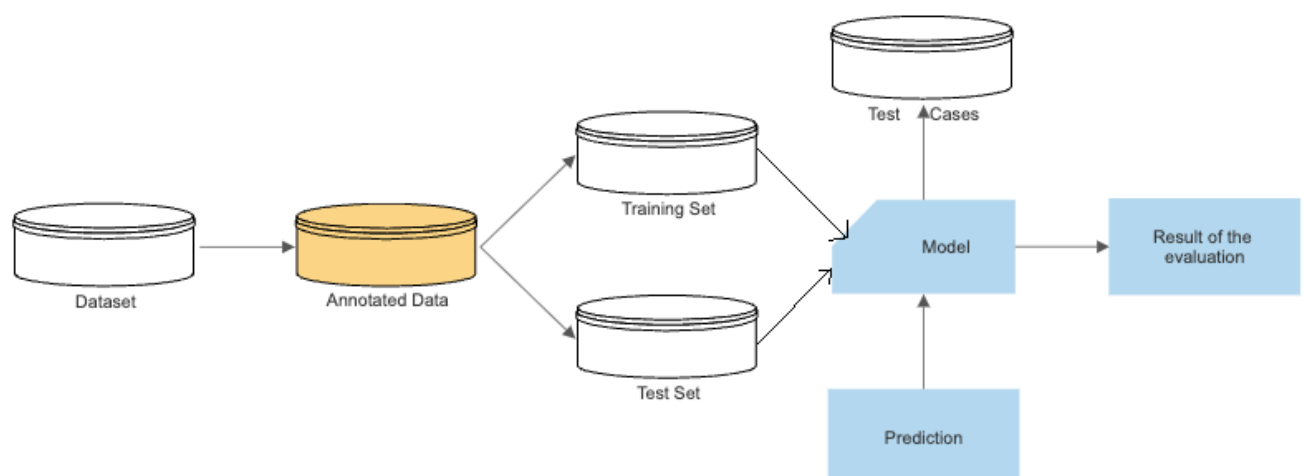
3 Design Details

3.1 Process Flow

For identifying the different types of anomalies, we will use a deep learning base model. Below is the process flow diagram is as shown below.
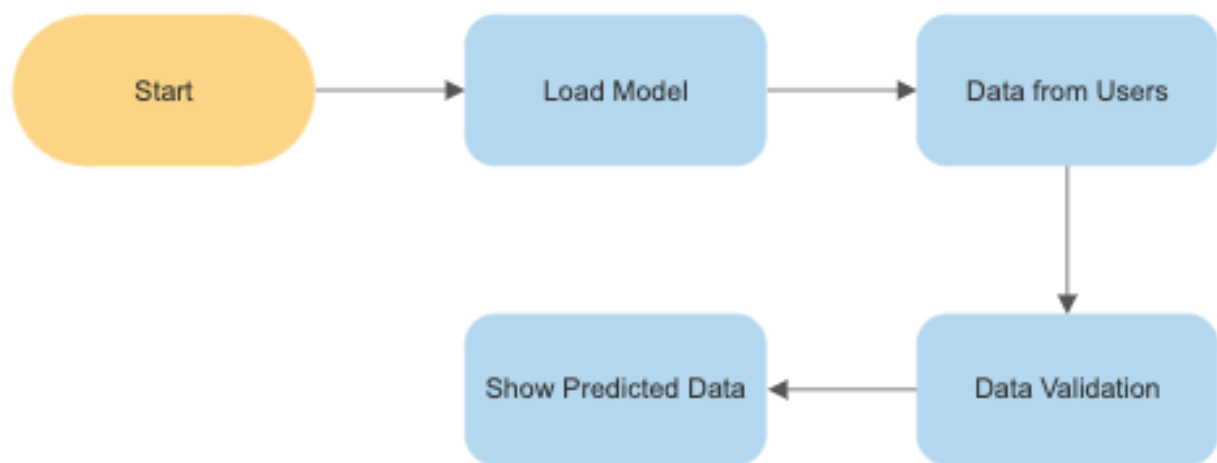
Proposed methodology

3.1.1 Model Training and Evaluation

3.1.2 Deployment Process

3.2 Event log

The system should log every event so that the user will know what process is running internally.

Initial Step-By-Step Description:

1. The System identifies at what step logging required
2. The System should be able to log each and every system flow.
3. System should not hang even after using so many loggings. Logging just because we can easily debug issues so logging is mandatory to do.

3.3 Error Handling

Should errors be encountered, an explanation will be displayed as to what went wrong? An error will be defined as anything that falls outside the normal and intended usage

3.4 Performance

The prediction on the test cases shows that the performance of the model is quite well.

3.5 Reusability

The code written and the components used should have the ability to be reused with no problems.

3.6 Application Compatibility

The different components for this project will be using Python as an interface between them. Each component will have its own task to perform, and it is the job of the Python to ensure proper transfer of information.

3.7 Resource Utilization

When any task is performed, it will likely use all the processing power available until that function is finished.

3.8 Deployment

The deployment of the project is done on the Heroku.

Conclusion

The credit card default prediction will detect whether the person will default their payment of credit card bills or not.