

# High-Level Design Document

## Big Mart Store Sales Prediction

Written By	Vijaykumar Rewate
Document Version	0.1
Last Revised Date	25.5.2023

## Abstract

The Big Mart Store Sales Prediction project aims to develop a solution that can accurately forecast the sales of different stores in the Big Mart chain based on historical sales data. By leveraging machine learning algorithms and data analysis techniques, this solution will enable the stores to optimize their inventory management and make informed business decisions.

This high-level design document provides an overview of the Big Mart Store Sales Prediction project. It outlines the scope, problem statement, proposed solution, and technical requirements. The document also covers the design details, including process flow, error handling, performance considerations, and deployment. Additionally, it discusses the dashboards and key performance indicators (KPIs) for monitoring the system. The document concludes with a summary of the project.

## Contents

### 1 Introduction

#### 1.1 Why this High-Level Design Document?

#### 1.2 Scope

#### 1.3 Definitions

### 2 General Description

#### 2.1 Product Perspective

#### 2.2 Problem Statement

#### 2.3 Proposed Solution

#### 2.4 Architecture Overview

#### 2.5 Further Improvements

#### 2.6 Technical Requirements

#### 2.7 Data Requirements

#### 2.8 Tools Used

##### 2.8.1 Hardware Requirements

##### 2.8.2 Tech Stack Used

#### 2.9 Constraints

#### 2.10 Assumptions

### 3 Design Details

#### 3.1 Process Flow

##### 3.1.1 Model Training and Evaluation

##### 3.1.2 Deployment Process

#### 3.2 Event Log

#### 3.3 Error Handling

#### 3.4 Performance

#### 3.5 Reusability

#### 3.6 Application Compatibility

#### 3.7 Resource Utilization

#### 3.8 Deployment

### 4 Conclusion

## Introduction

### 1 Introduction

#### 1.1 Why this High-Level Design Document?

This document is created to provide a comprehensive understanding of the design and architecture of the Big Mart Store Sales Prediction project. It serves as a reference for project stakeholders, developers, and system architects to ensure a clear understanding of the project goals and design decisions.

#### 1.2 Scope

The scope of the Big Mart Store Sales Prediction project is to develop a solution that can accurately predict the sales of different stores in the Big Mart chain based on historical sales data. The solution will involve data exploration, data cleaning, feature engineering, model building, and model testing using various machine learning algorithms.

#### 1.3 Definitions

- Big Mart: A retail chain consisting of multiple stores that sell a variety of products.
- Sales Prediction: The process of forecasting future sales based on historical sales data and other relevant factors.
- Database: Collection of all information monitored by this system

## 2. General Description

### 2.1 Product Perspective

The Big Mart Store Sales Prediction project aims to provide insights into sales patterns and help optimize inventory management in Big Mart stores. The solution will utilize machine learning algorithms to analyze historical sales data and make accurate predictions for future sales.

### 2.2 Problem Statement

The problem at hand is to predict the sales of Big Mart stores using the provided dataset. The solution will leverage machine learning techniques to identify relevant patterns and factors that contribute to sales, and building a predictive model to forecast future sales accurately.

### 2.3 Proposed Solution

The proposed solution involves data exploration to understand the dataset, data cleaning to handle missing values and outliers, feature engineering to extract meaningful features, model building using various machine learning algorithms, and model testing to evaluate the performance of the trained models. The best-performing model will be selected for deployment.

The following approach will be followed to solve the problem:

- **Data Exploration:** Perform an initial analysis of the dataset to gain insights into the available data. Understand the data distribution, identify missing values, and detect any outliers or anomalies.
- **Data Cleaning:** Preprocess the dataset to handle missing values, outliers, and inconsistencies. This may involve techniques such as imputation, outlier removal, and data normalization.
- **Feature Engineering:** Extract meaningful features from the dataset that can contribute to accurate sales prediction. This may include creating new features, transforming existing ones, or encoding categorical variables.
- **Model Building:** Utilize various machine learning algorithms to develop a predictive model. Experiment with different algorithms like linear regression, decision trees, random forests, or gradient boosting to determine the most suitable one for this specific case. Perform model training, hyperparameter tuning, and cross-validation to optimize model performance.
- **Model Evaluation:** Evaluate the trained model using appropriate evaluation metrics such as mean squared error (MSE), root mean squared error (RMSE), or mean absolute error (MAE). Validate the model's performance on a separate test dataset to assess its ability to generalize and make accurate predictions.

### 2.4 Architecture Overview

The high-level architecture of the Big Mart Store Sales Prediction project is as follows:

- **Data Ingestion:** The provided dataset will be ingested into the system for further processing. This dataset may reside in a file or a database.
- **Data Exploration and Cleaning:** Perform exploratory data analysis (EDA) to understand the dataset's characteristics and identify any data quality issues. Cleanse the data by handling missing values, outliers, and inconsistencies.
- **Feature Engineering:** Extract relevant features from the dataset and transform them into a suitable format for model training. This may include one-hot encoding, feature scaling, or dimensionality reduction techniques.
- **Model Training:** Utilize the processed data to train machine learning models using various algorithms. Perform hyperparameter tuning to optimize model performance.
- **Model Evaluation:** Assess the trained models' performance using appropriate evaluation metrics and cross-validation techniques. Select the best-performing model for deployment.
- **Model Deployment:** Build a FastAPI-based web service that exposes the trained model as an API endpoint. This API will accept input data and return predicted sales values.
- **Application Packaging:** Containerize the application components, including the API, ML model, and necessary dependencies, using Docker. This will ensure that the application can be deployed consistently across different environments.
- **Data Storage:** Store the processed and transformed data in a MongoDB database for future reference, analysis, or retraining of the model if necessary.
- **User Interface:** Develop a user interface (UI) that allows users to interact with the system. The UI can provide options to input new data for sales prediction, visualize historical sales trends, and display predictions generated by the model.
- **Monitoring and Logging:** Implement logging and monitoring mechanisms to track the performance of the system, capture errors or exceptions, and gather usage statistics. This will enable efficient troubleshooting and system maintenance.
- **Scalability and Performance:** Consider the scalability and performance requirements of the system, especially if dealing with a large volume of data or high user traffic. Use appropriate techniques such as distributed processing or caching to optimize system performance.

## 2.5 Further Improvements

In the future, the solution can be enhanced by incorporating advanced analytics techniques, such as clustering or time series analysis, to identify additional patterns in sales data. Integration with external data sources, such as weather data or social media trends, can also be explored to include additional factors that may impact sales predictions. While the initial solution focuses on accurate sales prediction, there are several potential enhancements that can be considered for future iterations:

- **Advanced Models:** Explore more sophisticated machine learning algorithms, such as ensemble methods or deep learning models, to improve prediction accuracy.
- **Real-time Data Processing:** Implement real-time data processing capabilities to handle streaming data and make instant predictions based on the most recent sales data.
- **Advanced Analytics:** Incorporate advanced analytics techniques, such as clustering or time series analysis, to identify additional patterns or insights from the sales data.
- **Integration with External Data Sources:** Integrate the solution with external data sources, such as weather data or social media trends, to incorporate additional factors that may impact sales.
- **Automated Deployment and CI/CD:** Implement automated deployment pipelines and continuous integration/continuous deployment (CI/CD) practices to streamline the deployment and updates of the application.
- **User Authentication and Authorization:** Enhance the system by adding user authentication and authorization mechanisms to secure access to sensitive data and functionalities.
- **Reporting and Visualization:** Develop comprehensive reporting and visualization capabilities to present sales predictions, trends, and insights in an intuitive and actionable manner.

## 2.6 Technical Requirements

The Big Mart Store Sales Prediction solution will be developed using the following tech stack:

- **Programming Language:** Python
- **Web Framework:** FastAPI
- **Libraries:** Pandas, Numpy, Scikit-learn, Category Encoders, Seaborn, Matplotlib
- **Containerization:** Docker
- **Database:** MongoDB

## 2.7 Data Requirements

The solution requires historical sales data from Big Mart stores. The dataset should include information about individual items, such as item weights, prices, and item categories, as well as store-specific details, such as store size and location. Additionally, any relevant external data, such as promotional offers or competitor data, can be incorporated for improved predictions.

## 2.8 Tools Used

### 2.8.1 Hardware Requirements

The solution can be deployed on standard hardware infrastructure, including servers or cloud-based platforms. Sufficient computing resources and storage capacity should be allocated to handle the data processing and model training tasks efficiently.

### 2.8.2 Tech Stack Used

The following technologies and frameworks will be utilized in the project:

- **Python:** The primary programming language for data manipulation, analysis, and model development.
- **FastAPI:** A Python web framework for building APIs. It will be used to create a RESTful API that exposes the trained model for making predictions.
- **Machine Learning Algorithm:** Various machine learning algorithms will be explored and implemented using libraries such as scikit-learn. The specific algorithms will be selected based on their suitability for sales prediction.
- **Docker:** Containerization technology to package the application, including the API and required dependencies, into a portable and reproducible environment.
- **MongoDB:** A NoSQL database that can be used to store the processed and transformed data for future reference or analysis.

### 2.9 Constraints

The development and implementation of the Big Mart Store Sales Prediction solution should adhere to the following constraints:

- **Budget:** The solution should be developed within the allocated budget.
- **Time:** The project should be completed within the specified timeframe.
- **Scalability:** The solution should be designed to handle a large volume of data and support a high number of concurrent users.
- **Security:** Measures should be implemented to ensure the privacy and security of sensitive data.
- **Compatibility:** The solution should integrate seamlessly with Big Mart's existing IT infrastructure, including data warehouse systems.
- **User-Friendliness:** The solution should provide an intuitive user interface for easy interaction and access to sales predictions.
- **Robustness:** The solution should handle exceptions and errors gracefully, with appropriate error handling mechanisms in place.

### 2.10 Assumptions

- It is assumed that the provided dataset is reliable and accurately represents the historical sales data of Big Mart stores.



- It is assumed that there are no significant external factors, such as economic crises or regulatory changes, that would drastically impact the sales patterns and render the predictions inaccurate.
- It is assumed that the stakeholders and users have a basic understanding of machine learning concepts and are willing to provide the necessary input and feedback during the development process.

### 3 Design Details

#### 3.1 Process Flow

The process flow of the Big Mart Store Sales Prediction project involves the following steps:

##### 3.1.1 Model Training and Evaluation

- Data exploration: Analyze the dataset to understand its structure and identify any data quality issues.
- Data cleaning: Handle missing values, outliers, and inconsistencies in the data.
- Feature engineering: Extract relevant features from the dataset and transform them into a suitable format for model training.
- Model selection and training: Apply various machine learning algorithms, such as linear regression, decision trees, or neural networks, and train them using the prepared dataset.
- Model evaluation: Assess the performance of each trained model using appropriate evaluation metrics, such as mean squared error or R-squared value, to select the best-performing model.

##### 3.1.2 Deployment Process

- Model deployment: Deploy the selected model in a production environment, integrating it with the application infrastructure.
- Data preprocessing: Preprocess new data inputs by applying the same data cleaning and feature engineering steps used during model training.
- Sales prediction: Apply the deployed model to the preprocessed data to generate sales predictions for the different Big Mart stores.
- Presentation and visualization: Present the sales predictions to users through a user interface, providing visualization of historical sales trends and displaying the predicted sales for each store.

#### 3.2 Event Log

An event log can be maintained to capture important events and actions within the system, such as model training, model deployment, and user interactions. The event log can be used for auditing purposes, troubleshooting, and monitoring system performance.

#### 3.3 Error Handling

The system should include robust error handling mechanisms to handle exceptions and errors that may occur during data processing, model training, or user interactions. Error messages should be informative and actionable, guiding users or system administrators on how to resolve the issues encountered.

#### 3.4 Performance

Performance considerations should be taken into account to ensure efficient processing and response times. Techniques such as caching, parallel processing, or distributed computing can be

employed to optimize system performance, especially when dealing with a large volume of data or high user traffic.

### 3.5 Reusability

The solution should be designed with reusability in mind, allowing for easy integration with other systems or future enhancements. Modular and well-documented code, standardized APIs, and clear separation of concerns can facilitate reusability and maintainability.

### 3.6 Application Compatibility

The application should be compatible with different devices and operating systems, providing a responsive and consistent user experience across platforms, including desktop and mobile devices.

### 3.7 Resource Utilization

Efficient utilization of computational resources, such as memory and processing power, should be considered to optimize system performance and ensure scalability. Resource allocation strategies, such as load balancing and resource pooling, can be employed to distribute the workload effectively and maximize resource utilization.

### 3.8 Deployment

The deployment process involves setting up the production environment for the Big Mart Store Sales Prediction solution. This includes:

- Configuring the server infrastructure or cloud platform to host the application.
- Setting up the necessary software dependencies, such as Python, FastAPI, and Docker.
- Deploying the trained machine learning model in a production-ready environment.
- Establishing secure connections and access controls to protect data and system resources.
- Conducting thorough testing and verification to ensure the deployed solution is functioning as expected.

#### 4 Conclusion

In conclusion, the Big Mart Store Sales Prediction project aims to leverage machine learning algorithms and data analysis techniques to accurately predict the sales of different stores in the Big Mart chain. This high-level design document has provided an overview of the project, including the problem statement, proposed solution, technical requirements, and design details, with scope for future enhancements and scalability. By following the outlined approach, incorporating the defined constraints and assumptions, utilizing the selected tech stack, and considering the specific requirements of the project, the goal of developing an effective sales prediction system for Big Mart can be achieved.