SALES FORECAST PREDICTION

FINAL YEAR PROJECT REPORT

Submitted by

SAKTHIVEL.S	**142219104107**
SASITHARAN.M	**142219104111**
SHAKIR AHAMED.M	**142219104114**

In a partial fulfillment for the award of the degree

Of

BACHELOR OF ENGINEERING

In

COMPUTER SCIENCE AND ENGINEERING

SRM VALLIAMMAI ENGINEERING

COLLEGE

(AN AUTONOMOUS INSTITUTION)

SRM NAGAR, KATTANKULATHUR, CHENGALPET

ANNA UNIVERSITY: CHENNAI 600 025

NOVEMBER 2022

ANNA UNIVERSITY

BONAFIDE CERTIFICATE

Certified this project report **"SALES FORECAST PREDICTION" is the bonafide work of "SAKTHIVEL.S (142219104107), SASITHARAN.M(142219104111) and SHAKIR AHAMED. M (142219104114)"** who carried out the work under my Supervision.

SIGNATURE	**SIGNATURE**
Dr. V. DHANAKOTI M.E., Ph.D.,	**Dr. B. VANATHI, M.E., Ph.D.,**
ASSOCIATE PROFESSOR	**PROFESSOR & HEAD**
Department of CSE,	Department of CSE,
SRM Valliammai Engineering College,	SRM Valliammai Engineering College,
Kattankulathur-603 203.	Kattankulathur-603 203.

Submitted for the university examination held on__________________at SRM Valliammai Engineering College, Kattankulathur.

INTERNAL EXAMINER **EXTERNAL EXAMINER**

ACKNOWLEDGEMENT

We sincerely express our gratitude in depth to our esteemed Founder Chairman & Chancellor **Dr. T. R. Paarivendhar, Thiru. Ravi Pachamoothoo** Chairman SRM Group & Pro-Chancellor (Admin), **Mrs. R. Harini,** Correspondent, SRM VEC and authorities of **SRM VALLIAMMAI ENGINEERING COLLEGE** for the patronage on our welfare rooted in the academic year.

We express our sincere gratitude to our respected **Director Dr. B. Chidhambararajan, M.E., Ph.D.,** for his constant encouragement, which has been our motivation to strive towards excellence.

We thank our sincere **Principal Dr. M. Murugan, M.E., Ph.D.,** for his constant encouragement, which has been our motivation to strive towards excellence.

We extend our hand of thanks to our Head of the Department,
Dr. B. Vanathi, M.E., Ph.D., Professor for her unstinted support.

We are grateful to our internal guide **Dr. V. Dhanakoti, M.E., Ph.D.,** Associate Professor without whose invaluable guidance and encouragement, this project would not have been a success.

We also like to thank all Teaching and non-teaching staff members of our department, for their support during the course of the project. We finally thank our friends and family for their support during the course of the project.

ABSTRACT

Business intelligence is one of the demanded skills in information technology and every aspect of life is changing as a result of machine learning and Business intelligence, which has also significantly impacted situations in the real world. Every industry, including education, healthcare, engineering, sales, entertainment, and transportation, has embraced the transformative uses of machine learning. as it is the current stipulation, we have developed model where it predicts the future sales pattern of a product by imparting the purchase history between a time period to the model. This project focuses on analyzing and visualizing the regional sales of products. The traditional method of achieving sales and marketing objectives doesn't benefit businesses anymore because it doesn't take into account how people actually buy things. Advances in machine learning have led to significant changes in the field of sales and marketing. The underlying algorithm is based on the linear regression, the random forest classifier and XGboost algorithm. Segmenting consumer-based buying behavior and applying 80/20 rule to identify top customers/products by applying XGboost, Linear regression and Random Forest classifier to calculate the optimal number of customer segments with similar buying habits (features) and predict the sales of a certain product using these algorithms.

சுருக்கம்

வணிக நுண்ணறிவு என்பது தகவல் தொழில்நுட்பத்தில் கோரப்படும் திறன்களில் ஒன்றாகும், இது தற்போதைய நிபந்தனையாக இருப்பதால், மாடலுக்கு ஒரு காலத்திற்கு இடையில் கொள்முதல் வரலாற்றை வழங்குவதன் மூலம் ஒரு தயாரிப்பின் எதிர்கால விற்பனை முறையை முன்னறிவிக்கும் மாதிரியை நாங்கள் உருவாக்கியுள்ளோம். இந்த திட்டம் தயாரிப்புகளின் பிராந்திய விற்பனையை பகுப்பாய்வு செய்வதிலும் காட்சிப்படுத்துவதிலும் கவனம் செலுத்துகிறது. அடிப்படை அல்காரிதம் நேரியல் பின்னடைவு மற்றும் சீரற்ற வன வகைப்படுத்தி ஆகியவற்றை அடிப்படையாகக் கொண்டது. நுகர்வோர் அடிப்படையிலான வாங்குதல் நடத்தையைப் பிரித்தல் மற்றும் 80/20 விதியைப் பயன்படுத்தி சிறந்த வாடிக்கையாளர்கள்/தயாரிப்புகளை அடையாளம் காண நேரியல் பின்னடைவு மற்றும் ரேண்டம் ஃபாரஸ்ட் வகைப்படுத்தி போன்ற வாங்கும் பழக்கம் (அம்சங்கள்) கொண்ட வாடிக்கையாளர் பிரிவுகளின் உகந்த எண்ணிக்கையைக் கணக்கிடுதல்.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER -1

INTRODUCTION

1.1 INTRODUCTION

Nowadays, shopping malls and Big Marts keep track of individual item sales data in order to forecast future client demand and adjust inventory management based on the prediction that is done on that data using various machine learning algorithms. In a data warehouse, these data stores hold a significant amount of consumer information and particular item details. By mining the data store from the data warehouse, more anomalies and common patterns can be discovered using these data. Most of our buying decisions are not based on well-defined logic. Emotions, trust, communication skills, culture, and intuition play a big role in our buying decisions. Although humans do not follow a well-defined logic, we do have some repeated patterns that we follow that vary for different people. We often buy the same things and behave in a similar way. When we look at DL algorithms, Neural networks are one of the most widely used DL algorithms these days. One of the main reasons of having widespread use of Neural Networks is because it can create an approximation of any function. The approximation is based on data, which it learns or is trained with. So neural nets can learn similar responses for similar inputs. Sales forecasts help businesses make better decisions based on future revenue, which will help them to: Forecast likely profit (or loss) in a designated period. Organize staffing levels and create HR plans. Plan the required level of production needed to meet demand.

1.2 GENERAL

Machine learning and artificial intelligence are something machine take their own decision based on the data we provide, and the architecture of the model without any human interference. As it is applied in real-time the performance of the model must be accurate to avoid the consequences. For example, there are 2 models based on traffic sign detection, one gives an accuracy of 87% and the other gives an accuracy of 90%. The 1st model predicts a traffic sign as 80km speed limit rather than predicting it as 60km speed limit which is needed to be predicted if suppose a vehicle is in auto-pilot mode the car considers the upcoming traffic sign as 80km and accelerate up to 80km so the vehicle is in Overspeed in that area, the density of traffic. This leads to the collision because of the low accuracy of the model. So, this is a real-time scenario that how important the performance of the model is and there are a lot of similar cases.

1.3 LITERATURE SURVEY

1. A Survey on Analysis of Online Consumer Behavior Using Association Rules

In vogue the competitive world is searching for customer-oriented approach. In order to accomplish that customer relation management (CRM) provides the services which meet the customer requirement. The requirements of a customer are collected in the form of data which is to be analyzed for obtaining customer behavior. Therefore, inquiry is made to determine the characteristic features of a customer by using the data which is collected from different sources of all the online shopping sites. Data mining is a process of extracting useful information from the historical data. In this analysis the data of a online shopping website which is take from mechanical sector is tested by using some data mining algorithms like Apriori, FP-growth and Eclat algorithm of association rule mining. As a result of the study, it is aimed to obtain frequent items purchased by the consumer by using the association results.

2. A Machine Learning Approach to Identify Potential Customer Based on Purchase Behavior

To lift the revenue boundary and stay ahead of the competitors it is important to understand customer’s purchase behavior. Different business industries proposed different policies to explore the potentiality of a customer based on statistical analysis. In this paper, we rather propose a machine learning approach to identify potential customers for a retail superstore. The paper proposed an engineered approach to classify potential customer, based on previously recorded purchase behavior. Using this classification as ground truth, we then apply machine learning algorithms to find a pattern to predict potential customers with an accuracy of 99.4%.

3. Predicting Consumer Behavior in Digital Market: A Machine Learning Approach

In recent times, customer behaviour models are typically based on data mining of customer data, and each model is designed to answer one question at one point in time. Predicting customer behaviour is an uncertain and difficult task. Thus, developing customer behaviour models requires the right technique and approach. Once a prediction model has been built, it is difficult to manipulate it for the purposes of the marketer, to determine exactly what marketing actions to take for each customer or group of customers. Despite the complexity of this formulation, most customer models are relatively simple. Because of this necessity, most customer behaviour models ignore so many pertinent factors that the predictions they generate are generally not very reliable. This paper aims to develop an association rule mining model to predict customer behaviour using a typical online retail store for data collection and extract important trends from the customer behaviour data.

4. Customer purchasing behavior prediction using machine learning classification techniques

This paper presents a comparative study of different machine learning techniques that have been applied to the problem of customer purchasing behavior prediction. Experiments are done using logistic regression, decision tree, k-nearest neighbors (KNN), Naïve Bayes, SVM, random forest, stochastic gradient descent (SGD), ANN, AdaBoost, XgBoost, and dummy classifier, as well as some hybrid algorithms that use stacking like SvmAda, RfAda, and KnnSgd. Furthermore, the confusion matrix and ROC curve are used to calculate the accuracy of each model. Finally, the best classifier is a hybrid classifier using the ensemble stacking technique (KnnSgd), with an accuracy of 92.42%. KnnSgd gives the highest accuracy with maximum features because the error of the KNN and SGD are minimized by the KNN at the end.

1.4 EXISTING SYSTEM

The primary objective of business sectors is market audience targeting. It is crucial that the business has been successful in achieving this goal by utilising a forecasting system. Forecasting requires examining data from a variety of sources, including market trends, customer behaviour, and other elements. Additionally, this study would assist businesses in efficiently managing their financial resources. Advances in machine learning have led to significant changes in the field of sales and marketing. Due to these developments, it is now much easier to evaluate important factors like target market demographics and sales projections for the upcoming years, which assists the sales staff in creating strategies for growing their firm. Although there are plenty of machine learning model developed for "SALES FORECASTING”, the scope and usage of the models is not up to the level of expectation and demand satisfying as many models referred by us makes use of common, well-known machine learning algorithms such as Linear Regression, Logistic Regression and K-Nearest Neighbor’s accuracy of the model is not considered much as it delivers a very low performance metrics, by considering this as an impairment case, it becomes a reason for developing a better model and user interactive that can be used to predict the user given data for their needs and to improve their marketing strategy.

1.5 PROPOSED SYSTEM

Here in this project, we put forward a inception of flask-"A framework developed in python", rather than making a machine learning model that predicts the outlet sales, we find it would be better if our work is published and a platform where users can make use of our project that is deploying the machine learning model in an web app that delivers a content about the title and the current demand, project insights such modules and its derivative, languages used, knowledge gathered and more on. Digging down deeper we introduced a never seen machine learning algorithm by us namely "XGBoost.” The reason behind selecting this model is, most of the machine learning algorithm works and tries to refine itself based on accuracy but XGBoost revamps itself based on error and predicts the next model by creating a Decision tree based on the previously entered data and its prediction comparing this value with the newly predicted value using a certain formula. So, we find it would be quite interesting to feed the data to the model that uses the above algorithm for the purpose of prediction. Based on a few key characteristics identified from the available raw data, the goal is to predict the sales pattern and the quantities of the products to be sold. To fully understand the data, analysis and study of the acquired data have also been done. At each crucial stage of the marketing strategy, analysis would assist business organisations in arriving at a probable decision using these machine learning algorithms.

CHAPTER- 2

REQUIREMENT AND SPECIFICATION

2.1 HARDWARE SPECIFICATION

OS: Microsoft Windows 11 Home Single Language

Processor: AMD Ryzen 5 5600H with Radeon Graphics

Memory: 16 GB

Disk space: 500 GB SSD

Anaconda Navigator, Jupyter:

The Jupyter Notebook is the platform used to implement the code. There is 50+ kernels running in the notebook, each containing visualization, performance metrics checking, and comparison.

Anaconda navigator is platform that provides various IDE for the developers to work upon, starting from web development to data analytics/scraping.

We recommend:

1.8 GHz or faster 64-bit processor; Quad-core or better recommended. ARM processors are not supported.

Minimum of 4 GB of RAM. Many factors impact resources used; we recommend 16 GB RAM for typical professional solutions.

Video card that supports a minimum display resolution of WXGA (1366 by 768); Visual Studio will work best at a resolution of 1920 by 1080 or higher.

2.2 SOFTWARE SPECIFICATION

2.2.1 Integrated Development Environment (IDE):

Jupyter Notebook is utilised as a platform for executing and implementing our ideas. Developers are free to use any IDE they like. Jupyter Notebook is employed because to its compilation mechanism, which uses independent individual kernels to code, allowing the code to be optimised separately. Jupyter notebook is free software that includes open optimised distinct standards and web services for interactive computing in any programming language. The original web application for producing and sharing computational documents is Jupyter Notebook. It provides a straightforward, simplified, document-centric interface. There are several functions it may give, a few of which are listed below.

Language of preference:

Jupyter supports approximately 40 programming languages, including Python, R, Julia, and Scala.

Distribute notebooks:

Notebooks can be shared with others by email, Dropbox, GitHub, or the Jupyter Notebook Viewer.

Interactive output:

HTML, pictures, movies, LaTeX, and custom MIME types may all be generated by your code.

Integration of big data:

Use big data technologies like Apache Spark from Python, R, and Scala. Investigate the same data using pandas, scikit-learn, ggplot2, and TensorFlow.

2.2.2 Programming Language:

Python:

Python is a general-purpose programming language that is high-level and interpreted. Its design philosophy places a strong emphasis on code readability, with a lot of indentation. Python is garbage-collected and typed dynamically. Instead of curly brackets or keywords, Python delimits blocks via whitespace indentation. After specific statements, indentation rises, and at the conclusion of the current block, indentation falls. As a result, the visual organisation of the programme correctly reflects the semantic structure of the programme. The off-side rule is a common name for this feature. Indentation is used in this fashion in certain other languages, although it has no semantic value in most of them. Indent four spaces is the optimum size. Python 3.9.13 is the most recent version, and pip may be used to install libraries and packages (Package installer for python). To verify that request's function, the pip programme searches PyPI for the package, resolves its dependencies, and installs everything in your current Python environment. The pip install command always searches for and installs the most recent version of the package.

Flask:

Flask is a web framework that provides libraries to build lightweight web applications in python. It is developed by **Armin Ronacher** who leads an international group of python enthusiasts (POCCO). It is based on WSGI toolkit and jinja2 template engine. Flask is considered as a micro framework.

Modules and packages:

This project makes extensive use of Python modules and techniques. Several library packages are available to process text, split data into training and testing groups, and feed training data to algorithms. Finally, visualisation charts and performance metrics such as accuracy and losses are displayed.

2.2.3 Modules:

Regular Expression:

A character sequence that generates a search pattern. The purpose is to check whether a string contains a specified search pattern.

Train_Test_Split:

Data is an important component of machine learning systems; how we calculate and assess it is critical. Sci-kit Learn offers a way for separating data into testing and training data. We input training data into the suggested model, which will learn a range of factors, including patterns, outliers, and insights. Users supply testing data to evaluate model performance such as accuracy and loss. It is entirely dependent on the model's performance.

accuracy_score:

The most significant factor in the model is accuracy. To achieve good accuracy, the model must have been properly manipulated. Datasets, model architecture, and finally, the methods and modules that are used in it. The more accurate the model provides, the more it is considered top-notch performance.

2.2.4 Algorithms:

XGboost:

Gradient Boosting is a popular boosting algorithm. In gradient boosting, each predictor corrects its predecessor's error. In contrast to Adaboost, the weights of the training instances are not tweaked, instead, each predictor is trained using the residual errors of predecessor as labels. There is a technique called the Gradient Boosted Trees whose base learner is CART (Classification and Regression Trees).

XGBoost is an implementation of Gradient Boosted decision trees. XGBoost models majorly dominate in many Kaggle Competitions. In this algorithm, decision trees are created in sequential form. Weights play an important role in XGBoost. Weights are assigned to all the independent variables which are then fed into the decision tree which predicts results. The weight of variables predicted wrong by the tree is increased and these variables are then fed to the second decision tree. These individual classifiers/predictors then ensemble to give a strong and more precise model. It can work on regression, classification, ranking, and user-defined prediction problems.

Linear Regression:

Linear Regression is a machine learning algorithm based on supervised learning. It performs a regression task. Regression models a target prediction value based on independent variables. It is mostly used for finding out the relationship between variables and forecasting. Different regression models differ based on – the kind of relationship between dependent and independent variables they are considering, and the number of independent variables getting used.

Linear regression performs the task to predict a dependent variable value (y) based on a given independent variable (x). So, this regression technique finds out a linear relationship between x (input) and y(output). Hence, the name is Linear Regression.

Random Forest Classifier:

Random Forest is a popular machine learning algorithm that belongs to the supervised learning technique. It can be used for both Classification and Regression problems in ML. It is based on the concept of ensemble learning, which is a process of combining multiple classifiers to solve a complex problem and to improve the performance of the model.

As the name suggests, "Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset." Instead of relying on one decision tree, the random forest takes the prediction from each tree and based on the majority votes of predictions, and it predicts the final output.

The greater number of trees in the forest leads to higher accuracy and prevents the problem of overfitting.

CHAPTER -3

SYSTEM DESIGN

3.1 GENERAL

The Object-Oriented programming language basically follows a simple flow of code with every work defined at different objects, each acting as a unique module. Object-oriented programming follows a bottom-up approach. Where each member is retrieved using objects which is particularly assigned to it. as mentioned above in abstract, the main motive is compared different machine learning algorithms and project a basic knowledge about the algorithms and where it is appropriate to apply. Every developer must know a prior knowledge of various machine learning and deep learning algorithms so that developers will have an idea in which cases/scenarios a specific type of algorithm must be used.

3.2 UML DIAGRAM

UML stands for Unified Modeling Language. Simply said, UML is a contemporary method of modelling and documenting software. It's really one of the most widely used business process modelling approaches. In the realm of software engineering, UML has mostly been employed as a general-purpose modelling language. It has, nevertheless, made its way into the documentation of several business processes and workflows. Activity diagrams, a sort of UML diagram, can, for example, be used to replace flowcharts. They offer a more uniform approach to modelling processes as well as a broader range of features to increase readability and efficacy.

ADVANTAGES:

1. Most popular and adaptable.
2. The Software Architecture Must Be Effectively Communicated.
3. You only need a fraction of the language to use it.
4. a variety of UML tools
5. No Need for Formal Notation

6. Increasing Degree of Complexity

The design scope should be examined while designing a use case to identify all parts that reside inside and beyond the limitations of the procedures. Anything important to the use case that is outside its bounds should be specified by a supporting actor or by another use case. The design scope might range from a single system to a whole organisation. Enterprise-level use cases often explain business processes.

Use-case diagrams describe a system's behaviour and aid in the capturing of the system's needs. Use-case diagrams explain the high-level functionality and scope of a system. These diagrams also depict the interactions between the system and its actors. The use cases and actors in use-case diagrams explain what the system does and how the players utilise it, but not how the system runs inside.

Use-case diagrams are useful in the following circumstances:

1. Before beginning a project, construct use-case diagrams to represent a firm so that all project participants have a shared knowledge of the workers, customers, and business operations.
2. Create use-case diagrams when gathering requirements to capture system needs and show people what the system should perform.
3. During the analysis and design phases, you may determine the classes that the system requires by using the use cases and actors from your use-case diagrams.
4. During the testing phase, use-case diagrams can be used to define system tests.

3.2.1 USE CASE DIAGRAM:

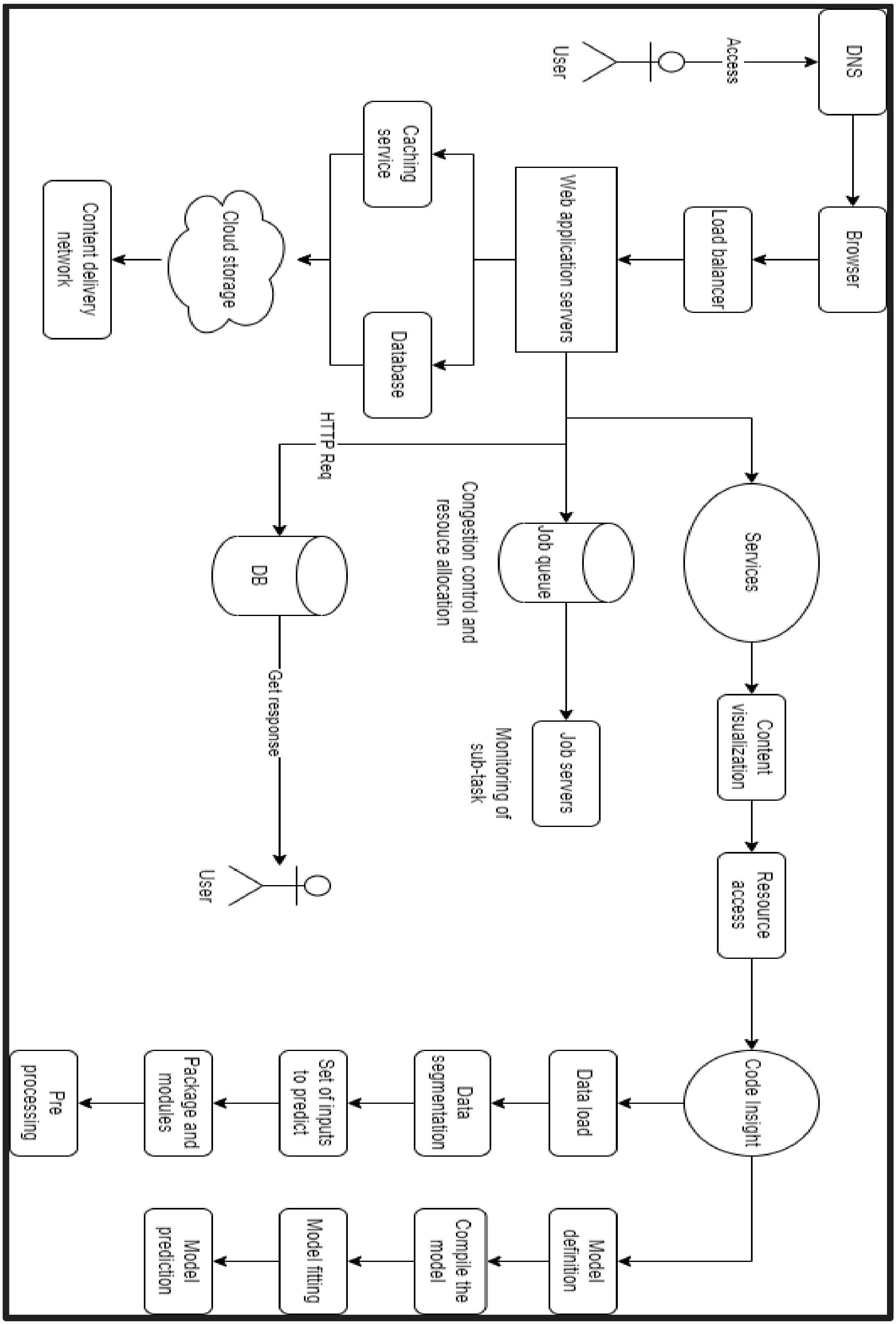


Fig. 3.2.1

Fig.3.2.1. Depicts the work flow of the source code, starting from the data extraction to final output.

Use-case Explanation:

This Use-case explains working of the three different modules which are Front end, middleware, and Back end and how they are integrated with each other to predict and show the result of our desired model. This Diagram helps us to understand the working of our model in a simpler way.

CHAPTER - 4

IMPLEMENTATION

4.1 IMPLEMENTATION

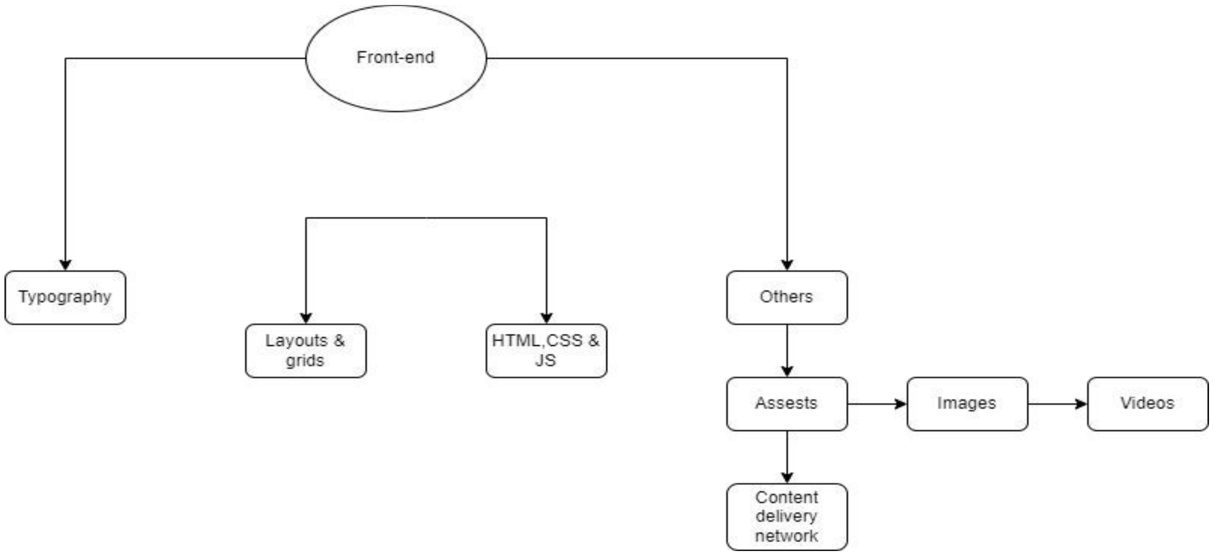
The Jupyter Notebook is the platform used to implement the code. There is 50+ kernels running in the notebook, each containing visualization, performance metrics checking, and comparison. The outcome of the project is a confusion matrix, which includes the accuracy and loss of each algorithm and plots those on a linear graph and scatters (Accuracy vs Loss). As the dataset used in the model is not balanced, let us consider an example. Suppose a person has a shoe factory and has 100,000 shoes, of which 99,000 are Adidas and 1,000 are Nike. Suppose a client needs a project where the model must be able to classify the shoe as two labels, whether it is Adidas or Nike, but the model always predicts Adidas no matter what brand and has an accuracy of 99%. It makes no sense in this case to provide high accuracy, but the model fails to predict correctly. So here the confusion matrix is projected. It has 2 columns and 2 rows, namely, true positive, true negative, and predicted positive and predicted negative. It shows a two-dimensional array of true and false values.

4.2MODULES:

Front End Development:

Front-end web development, also known as client-side development is the practice of producing HTML, CSS and JavaScript for a website or Web Application so that a user can see and interact with them directly. The challenge associated with front end development is that the tools and techniques used to create the front end of a website change constantly and so the developer needs to constantly be aware of how the field is developing.

The objective of designing a site is to ensure that when the users open up the site, they see the information in a format that is easy to read and relevant. This is further complicated by the fact that users now use a large variety of devices with varying screen sizes and resolutions thus forcing the designer to take into consideration these aspects when designing the site. They need to ensure that their site comes up correctly in different browsers (cross-browser), different operating systems (cross-platform) and different devices (cross-device), which requires careful planning on the side of the developer.

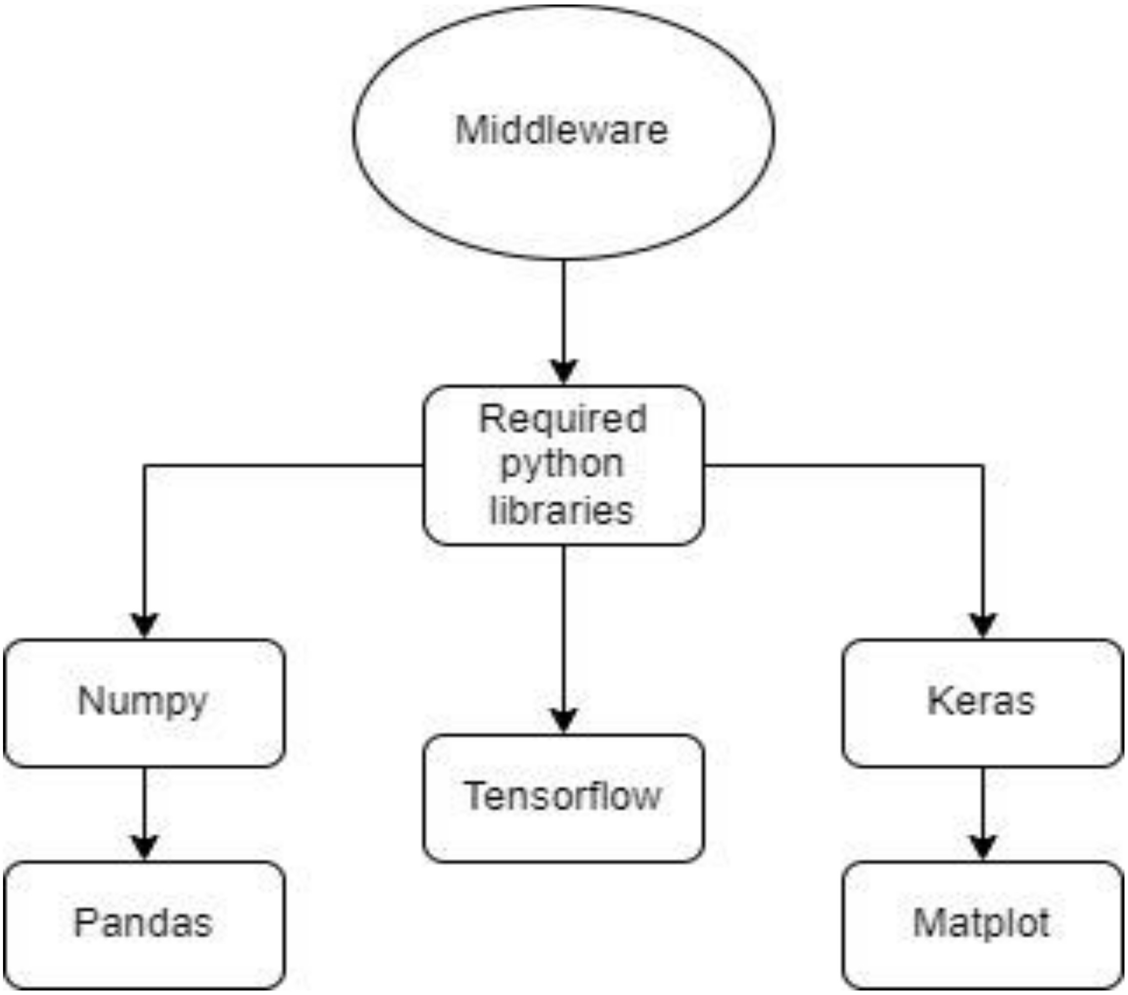


Middleware:

Middleware is software that provides common services and capabilities to applications outside of what is offered by the operating system. Data management, application services, messaging, authentication, and API management are all commonly handled by middleware.

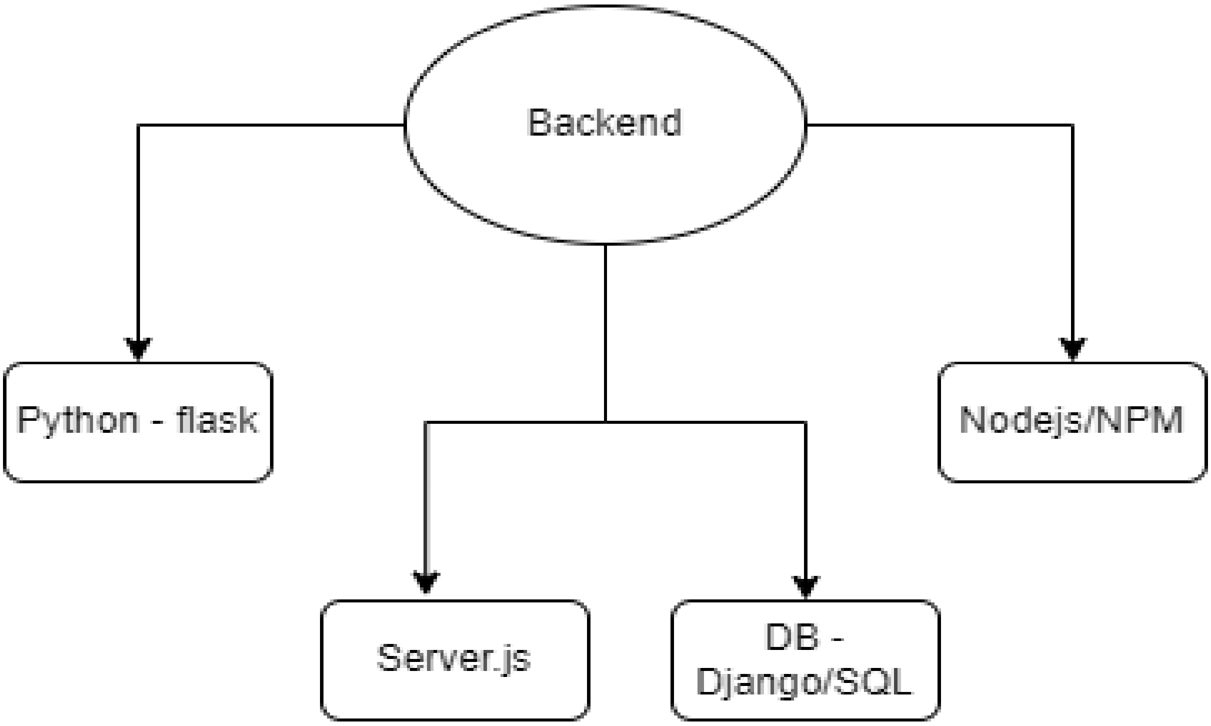
Middleware helps developers build applications more efficiently. It acts like the connective tissue between applications, data, and users.

For organizations with multi-cloud and containerized environments, middleware can make it cost-effective to develop and run applications at scale.



Back End Development:

Back-end development means working on server-side software, which focuses on everything you cannot see on a website. Back-end developers ensure the website performs correctly, focusing on databases, back-end logic, application programming interface (APIs), architecture, and servers. They use code that helps browsers communicate with databases, store, understand, and delete data. On a team, back-end developers collaborate with front-end developers, product managers, principal architects, and website testers to build the structure of a website or mobile app. Back-end developers must be familiar with many kinds of tools and frameworks, including languages such as Python, Java, and Ruby. They make sure the back-end performs quickly and responsively to front-end user requests.



4.3 DATA DEPICTION:

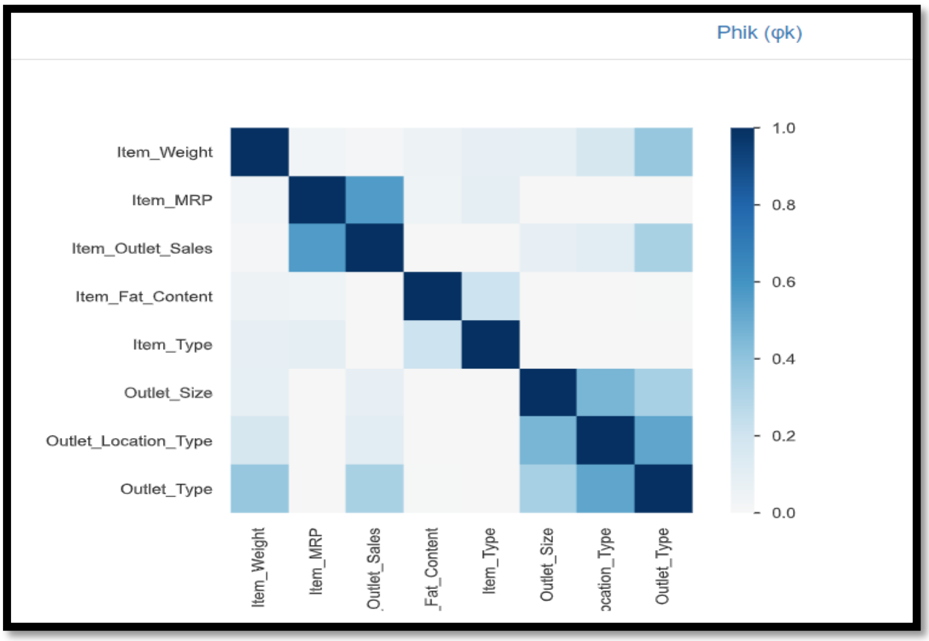


Fig 4.3.1

The Fig 4.3.1 depicts correlation between the different attributes

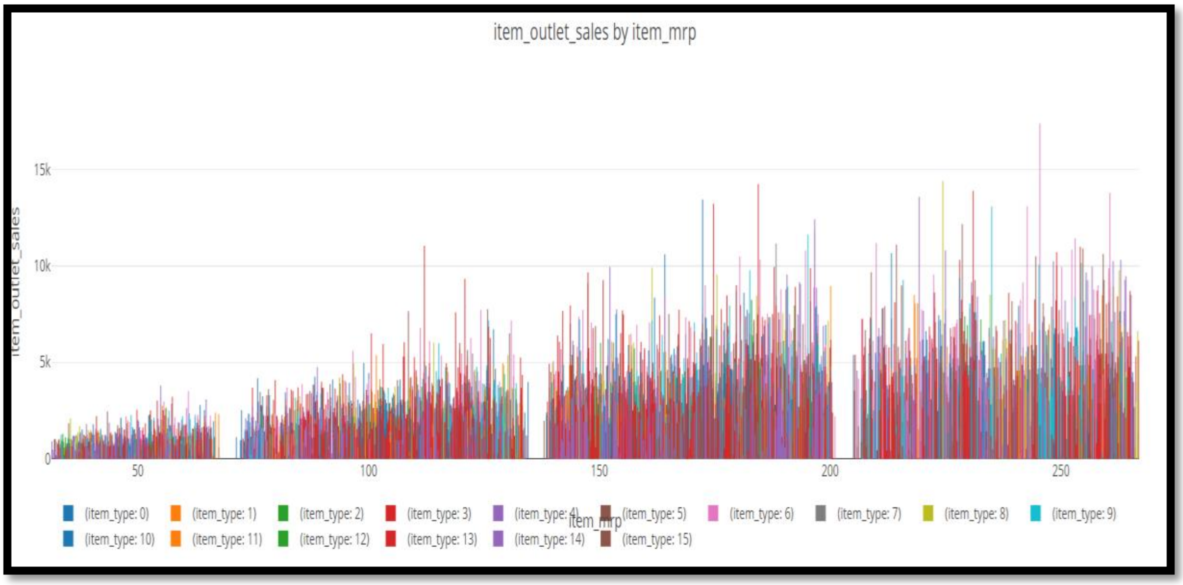


Fig 4.3.2

Fig 4.3.2 Depicts the Bar chart for the attributes such as Mrp and outlet sales

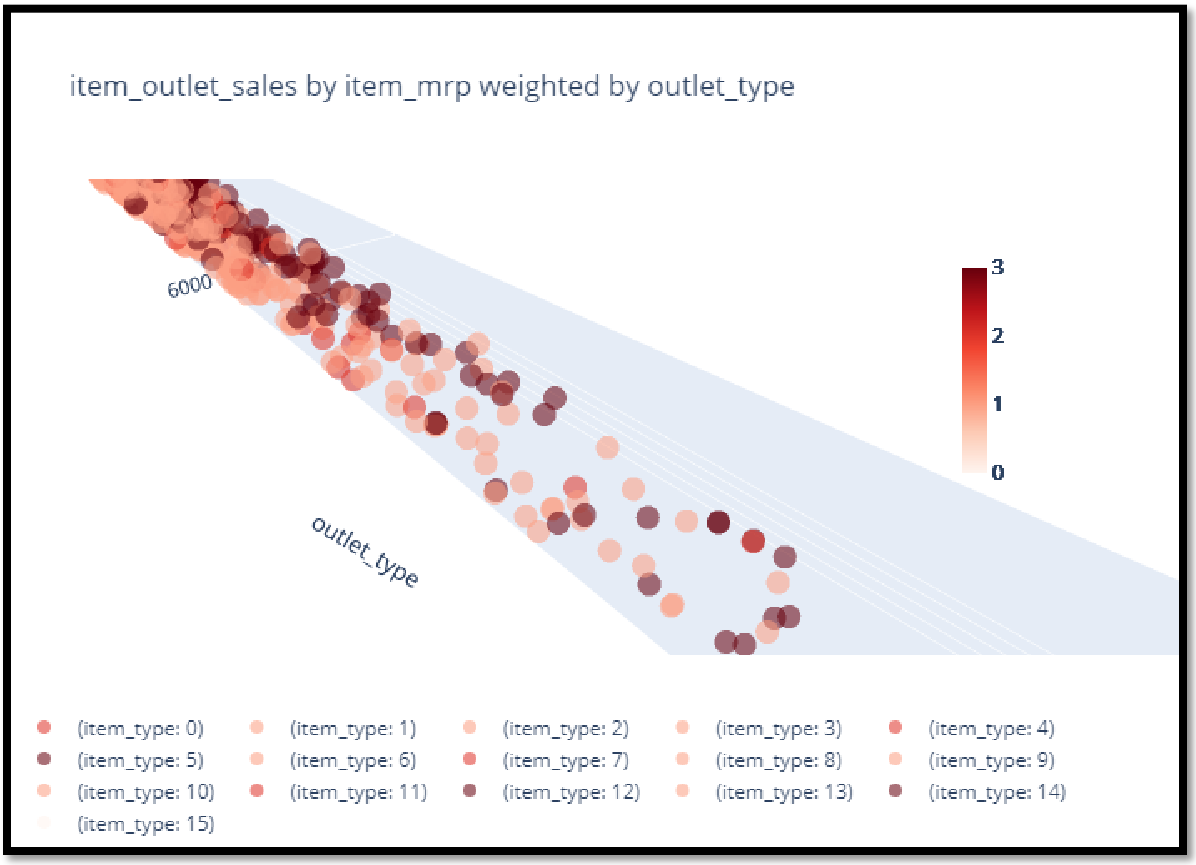


Fig 4.3.3

Fig 4.3.3 Depicts the item outlet sales by mrp weighted by outlet type

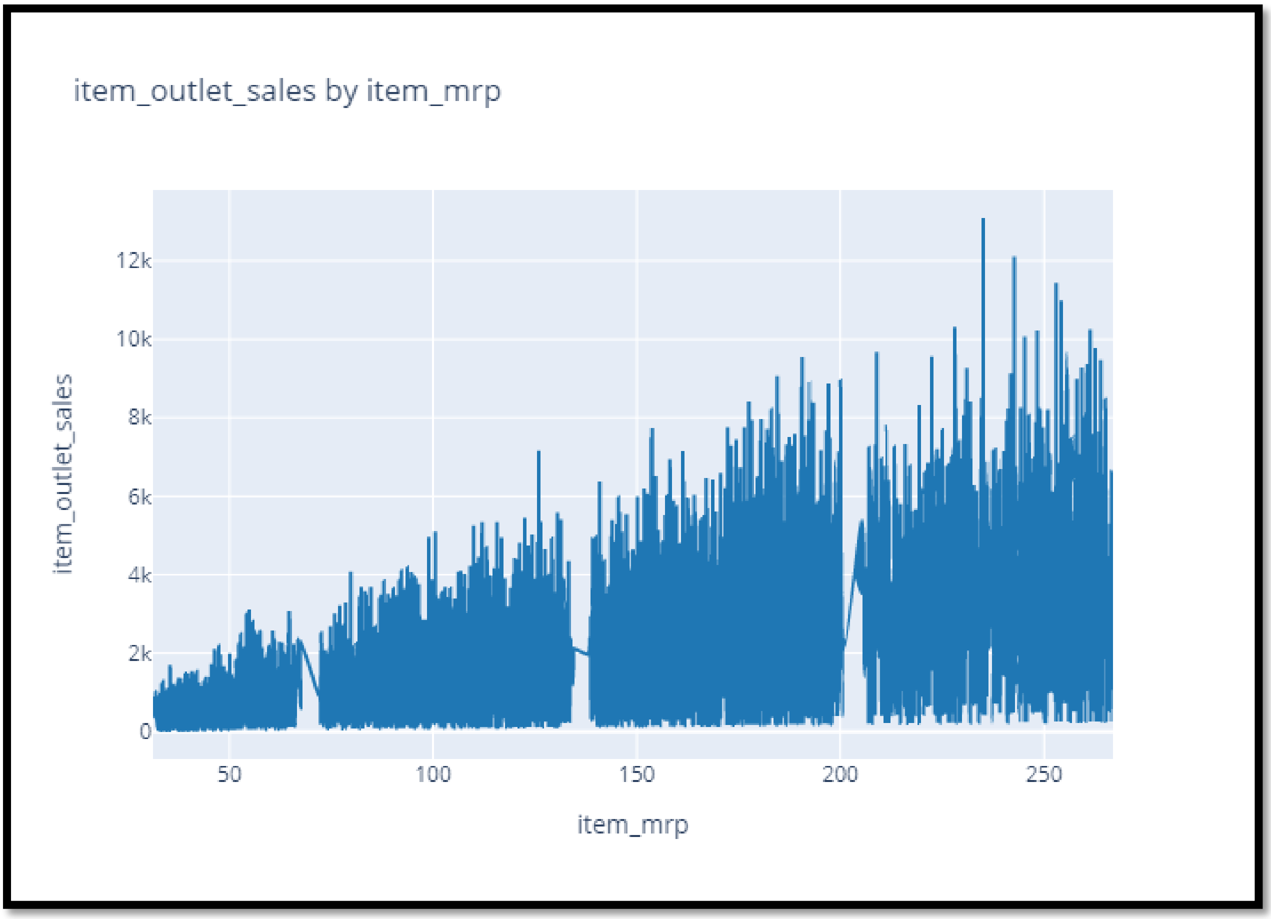


Fig 4.3.4

Fig 4.3.4 Depicts the line plot of the item outlet sales by item_mrp

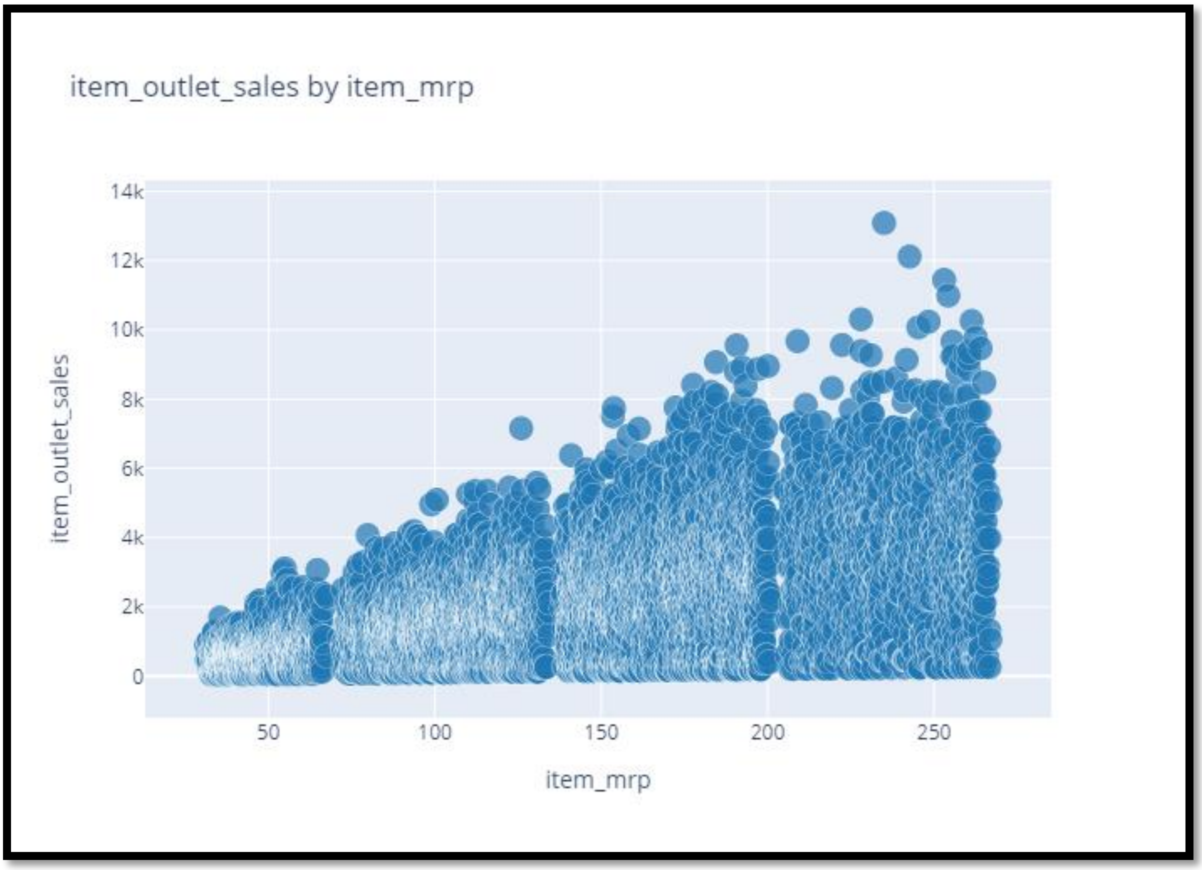


Fig 4.3.5

Fig 4.3.4 Depicts the scatter plot of the item outlet sales by item_mrp

CHAPTER - 5

RESULT AND TESTING

5.1 RESULT AND TESTING

5.1.1 RESULT

We have created a model that could predict sales outlet of a product using some of its attributes. Although there are various methods that have been suggested for detecting new clients, a machine learning method is rather uncommon. Our study is the first of its kind to employ machine learning to analyse consumer purchasing patterns for a major retail chain. To record the connection between categories, items, amount, measurement unit, and sales, we created features. Correctly recognizing a potential consumer can greatly help a firm.

5.1.2 TESTING

WHITE BOX TESTING

According to the definition, "white box testing" (also known as clear, glass box, or structural testing) is a testing approach that assesses a program's code and internal structure. White box testing entails inspecting the code's structure. When the internal structure of a product is known, tests may be done to check that the internal activities are executed as specified. And all internal components have been thoroughly tested. The examination of code coverage is a critical component of white box testing. It enables you to find holes and rectify them in order to increase the software's performance. Once the holes have been found, test cases must be developed based on the findings. The tester must go through all of the statements (which represent lines of code) and cover all of the code to verify there are no mistakes or problems. This method reveals any flaws and ensures complete code coverage. The goal is to run each line of code at least once.

GREY BOX TESTING

Gray box testing is a way of evaluating a software system – application or product both outside and internally by combining "white box testing" and "black box testing. "Gray box testing is performed with just a rudimentary understanding of the internal workings of the software system/application. Grey Box Testing (sometimes called Gray Box Testing) was designed as a fruitful fusion of white box and black box testing in order to overcome the limitations and ambiguities observed in such testing. This multimodal exam offers a thorough and highly targeted test, reducing testing time and expense. Furthermore, this technique allows our engineers to understand how your application works and assess whether a suspected vector of an attack is realistic or not at both ends, reducing false positive findings. This entails having access to internal data structures and algorithms in order to construct test cases. The test cases are developed based on this restricted information, and the tester tests the application from the outside on a Black Box level. The Gray Box tester considers the software to be a Black Box that must be examined from the outside

CHAPTER - 6

CONCLUSION AND FUTURE WORK

6.1CONCLUSION

Although there are various methods that have been suggested for detecting new clients, a machine learning method is rather uncommon. Our study is the first of its kind to employ machine learning to analyse consumer purchasing patterns for a major retail chain. To record the connection between categories, items, amount, measurement unit, and sales, we created features. Correctly recognizing a potential consumer can greatly help a firm. A tailored marketing strategy can be used to address the potential customer, increasing a business's sales. In the future, machine learning can be used to identify client behaviour, product interest, and purchasing frequency, allowing for more appropriate marketing plans and effective supply chain management.

6.2FUTURE WORK

Beginning from issues in selecting the appropriate algorithm, we have developed something quite easy to decide which algorithm performs well at which type of problem statement by comparing the accuracy and metrics of different algorithms on a base theory (sales forecast prediction). We thought of making our program more complex by collaborating with complex python libraries and frameworks, where we have planned to implement further project ideas, moreover although python has default parameter tunings, we planned to discover unique values for each parameter in an algorithm. Time is the only demon we are currently facing right now, as there are plans to develop as like mentioned in the above lines.

And there are lot of algorithms that are available that might give the results more accurate than the algorithm that we have used in this model.

CHAPTER - 7

REFERENCES

7. REFERENCES

1.Orogun Adebola and Bukola Onyekwelu, Predicting Consumer Behaviour in Digital Market: A Machine Learning Approach, 2019, Pp 17-23.

2.Gyanendra Chaubey, Prathamesh Rajendra Gavhane, Dhananjay Bisen, Siddhartha Kumar Arjaria et al., Customer purchasing behavior prediction using machine learning classification techniques, Journal of Ambient Intelligence and Humanized Computing, 2022, Pp.25-31.

3.B. Arivazhagan, S. Pandikumar, S. Bharani Sethupandian, R. Shankara Subramanian et al., Pattern Discovery and Analysis of Customer Buying Behavior Using Association Rules Mining Algorithm in E-Commerce, First International Conference on Electrical, Electronics, Information and Communication Technologies (ICEEICT), 2022, Pp.19-24.

4.Dr M.R.Narasingha Rao, K V.LSita Ratnam, M D.S.Prasanth, P.Lakshmi Bhavani et al., A Survey on Analysis of Online Consumer Behaviour Using Association Rules, International Journal of Engineering & Technology, 2018, Pp.36-40.

5.Adil Mahmud Choudhury, Kamruddin Nur, A Machine Learning Approach to Identify Potential Customer Based on Purchase Behavior, International Conference on Robotics, Electrical and Signal Processing Techniques (ICREST), 2019, Pp.26-31.

6.Quang Hung Do, Tran Van Trang, An approach based on machine learning techniques for forecasting Vietnamese consumers' purchase behaviour, 2020, Pp.14-19

7.Pornpimon Kachamas, Suphamongkol Akkaradamrongrat, Sukree Sinthupinyo, Achara Chandrachai et al., Application of Artificial Intelligent in the Prediction of Consumer Behavior from Facebook Posts Analysis, 2019, Pp.37-41.

8.Bo Zhao, Atsuhiro Takasu, Ramin Yahyapour, Xiaoming Fu, Loyal Consumers or One-Time Deal Hunters: Repeat Buyer Prediction for E-Commerce, 2019 International Conference on Data Mining Workshops (ICDMW), 2019, Pp.24-29.

9.Saifil Momin, Tanuj Bohra, Purva Raut et al., Prediction of Customer Churn Using Machine Learning, EAI International Conference on Big Data Innovation for Sustainable Cognitive Computing, 2020, Pp.19-26.

10.Rana Alaa El-Deen Ahmeda, Mohamed Elemam, Shereen Morsya, Nermeen Mekawiea et al., Performance Study of Classification Algorithms for Consumer Online Shopping Attitudes and Behavior Using Data Mining, Fifth International Conference on Communication Systems and Network Technologies (CSNT), 2019, Pp.27-36.

CHAPTER - 8

APPENDIX

8.1 ANNEXURE (SOURCE CODE):

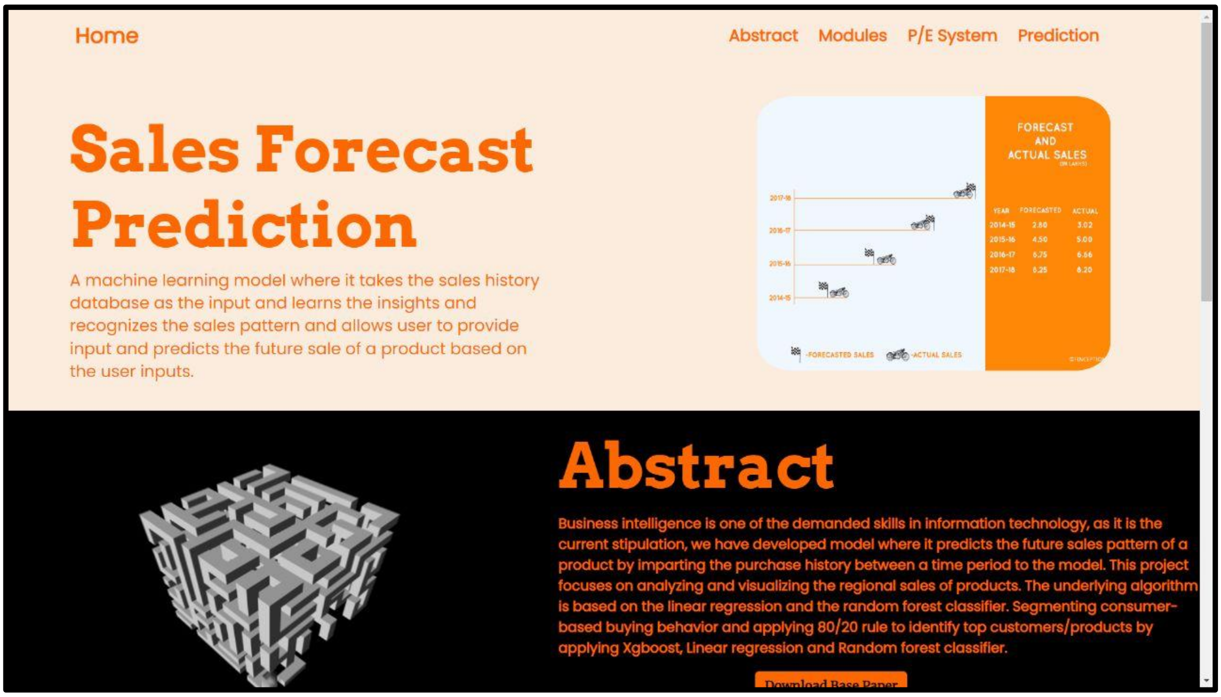
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
df_train=pd.read_csv(r'C:\Users\sasit\OneDrive\Desktop\pro\train.csv')
df_test=pd.read_csv(r'C:\Users\sasit\OneDrive\Desktop\pro\test.csv')
df_train
df_train['Item_Weight'].describe()
df_train['Item_Weight'].fillna(df_train['Item_Weight'].mean(),inplace=True)
df_test['Item_Weight'].fillna(df_test['Item_Weight'].mean(),inplace=True)
df_train.isnull().sum()
df_train['Outlet_Size']
df_train['Outlet_Size'].value_counts()
df_train['Outlet_Size'].fillna(df_train['Outlet_Size'].mode()[0],inplace=True)
df_test['Outlet_Size'].fillna(df_test['Outlet_Size'].mode()[0],inplace=True)
df_train
df_train.isnull().sum()
df_train.drop(['Item_Identifier','Outlet_Identifier'],axis=1,inplace=True)
df_test.drop(['Item_Identifier','Outlet_Identifier'],axis=1,inplace=True)
df_train
import pandas_profiling
from pandas_profiling import ProfileReport
profile = ProfileReport(df_train, title="Pandas Profiling Report")
profile
```

```
import klib
klib.cat_plot(df_train)
klib.data_cleaning(df_train)
klib.clean_column_names(df_train)
df_train=klib.convert_datatypes(df_train)
df_train.info()
klib.mv_col_handling(df_train)
from sklearn.preprocessing import LabelEncoder
le=LabelEncoder()
df_train['item_fat_content']=le.fit_transform(df_train['item_fat_content'])
df_train['item_type']=le.fit_transform(df_train['item_type'])
df_train['outlet_size']=le.fit_transform(df_train['outlet_size'])
df_train['outlet_location_type']=le.fit_transform(df_train['outlet_location_
type'])
df_train['outlet_type']=le.fit_transform(df_train['outlet_type'])
df_train
X=df_train.drop('item_outlet_sales',axis=1)
Y=df_train['item_outlet_sales']
from sklearn.model_selection import train_test_split
X_train,X_test,Y_train,Y_test=train_test_split(X,Y,
random_state=101,test_size=0.2)
X_train
Y_train
X.describe()
from sklearn.preprocessing import StandardScaler
sc=StandardScaler()
X_train_std=sc.fit_transform(X_train)
```

```
X_test_std=sc.transform(X_test)
X_train_std
X_test_std
Y_train
Y_test
import joblib
joblib.dump(sc,r'C:\Users\sasit\OneDrive\Desktop\pro\Model\sc.sav')
from sklearn.linear_model import LinearRegression
lr=LinearRegression()
lr.fit(X_train_std,Y_train)
Y_pred_lr=lr.predict(X_test_std)
Y_pred_lr
from sklearn.metrics import r2_score, mean_squared_error,
mean_absolute_error
print(r2_score(Y_test,Y_pred_lr))
print(np.sqrt(mean_squared_error(Y_test,Y_pred_lr)))
print(mean_absolute_error(Y_test,Y_pred_lr))
from sklearn.ensemble import RandomForestRegressor
rf=RandomForestRegressor()
rf.fit(X_train,Y_train)
Y_pred_rf=rf.predict(X_test)
print(r2_score(Y_test,Y_pred_rf))
print(np.sqrt(mean_squared_error(Y_test,Y_pred_rf)))
print(mean_absolute_error(Y_test,Y_pred_rf))
joblib.dump(rf,r'C:\Users\sasit\OneDrive\Desktop\pro\Model\rf.sav')
from sklearn.model_selection import RepeatedStratifiedKFold
from sklearn.model_selection import GridSearchCV
```

```
model= RandomForestRegressor()
n_estimators = [10,100,1000]
max_depth=range(1,31)
min_samples_leaf=np.linspace(0.1, 1.0)
max_features=["auto","sqrt","log2"]
min_samples_split=np.linspace(0.1, 1.0,10)
grid = dict(n_estimators=n_estimators)
#cv = RepeatedStratifiedKFold(n_splits=5, n_repeat=3, random_state=101)
grid_search_forest = GridSearchCV(estimator=model, param_grid=grid,
n_jobs=-1, scoring='r2',error_score=0,verbose=2,cv=2)
grid_search_forest.fit(X_train_std, Y_train)
print(f"Best: {grid_search_forest.best_score_:.3f} using
{grid_search_forest.best_params_}")
means = grid_search_forest.cv_results_['mean_test_score']
stds = grid_search_forest.cv_results_['std_test_score']
params = grid_search_forest.cv_results_['params']
for mean, stdev, param in zip(means, stds, params):
    print(f"{mean:.3f}({stdev:.3f}) with: {param}")
grid_search_forest.best_score_
```

8.2 ANNEXURE (OUTPUT)



Sales Forecast Prediction

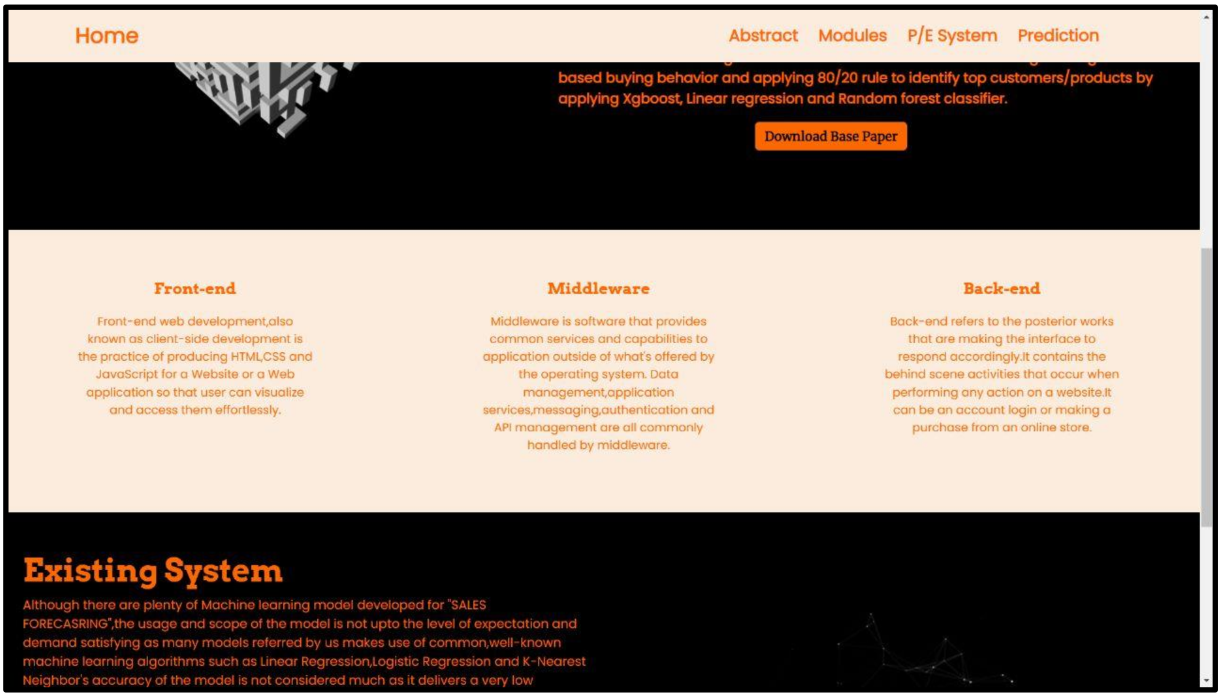
A machine learning model where it takes the sales history database as the input and learns the insights and recognizes the sales pattern and allows user to provide input and predicts the future sale of a product based on the user inputs.

FORECAST AND ACTUAL SALES (IN LAKHS)

YEAR	FORECASTED	ACTUAL
2014-15	2.80	3.02
2015-16	4.50	5.00
2016-17	6.75	6.66
2017-18	8.25	8.20

Abstract

Business intelligence is one of the demanded skills in information technology, as it is the current stipulation, we have developed model where it predicts the future sales pattern of a product by imparting the purchase history between a time period to the model. This project focuses on analyzing and visualizing the regional sales of products. The underlying algorithm is based on the linear regression and the random forest classifier. Segmenting consumer-based buying behavior and applying 80/20 rule to identify top customers/products by applying Xgboost, Linear regression and Random forest classifier.



based buying behavior and applying 80/20 rule to identify top customers/products by applying Xgboost, Linear regression and Random forest classifier.

Front-end

Front-end web development,also known as client-side development is the practice of producing HTML,CSS and JavaScript for a Website or a Web application so that user can visualize and access them effortlessly.

Middleware

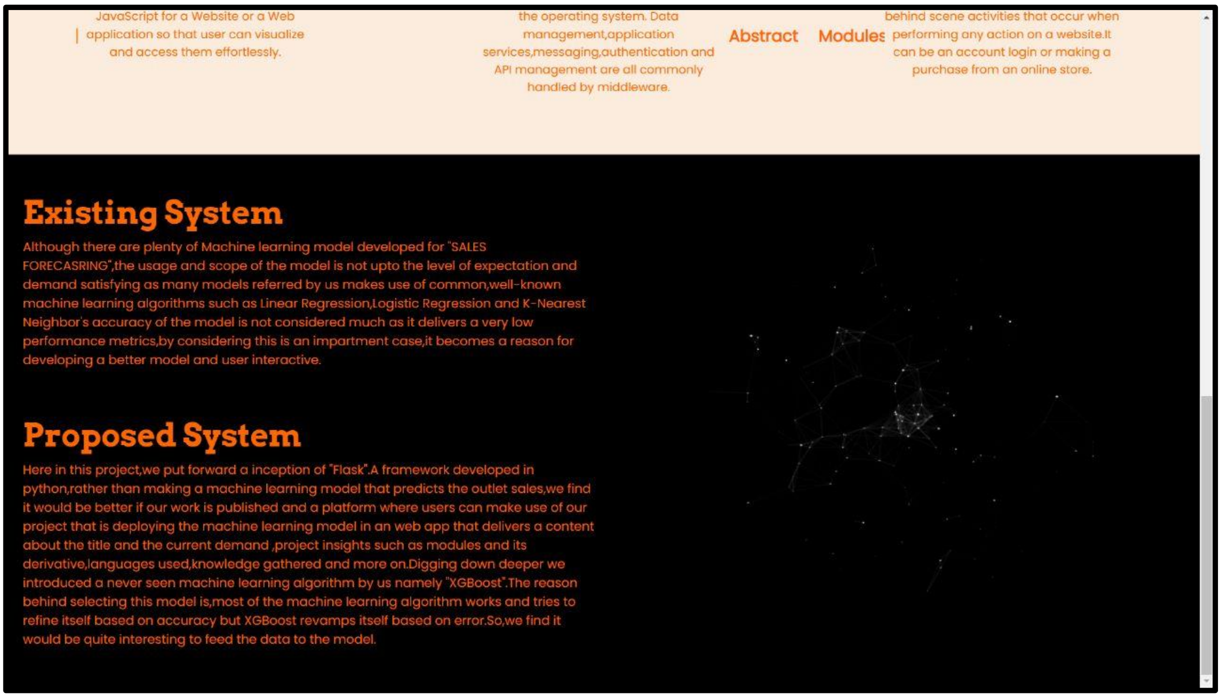
Middleware is software that provides common services and capabilities to application outside of what's offered by the operating system. Data management,application services,messaging,authentication and API management are all commonly handled by middleware.

Back-end

Back-end refers to the posterior works that are making the interface to respond accordingly.It contains the behind scene activities that occur when performing any action on a website.It can be an account login or making a purchase from an online store.

Existing System

Although there are plenty of Machine learning model developed for "SALES FORECASRING",the usage and scope of the model is not upto the level of expectation and demand satisfying as many models referred by us makes use of common,well-known machine learning algorithms such as Linear Regression,Logistic Regression and K-Nearest Neighbor's accuracy of the model is not considered much as it delivers a very low



JavaScript for a Website or a Web application so that user can visualize and access them effortlessly.

the operating system. Data management,application services,messaging,authentication and API management are all commonly handled by middleware.

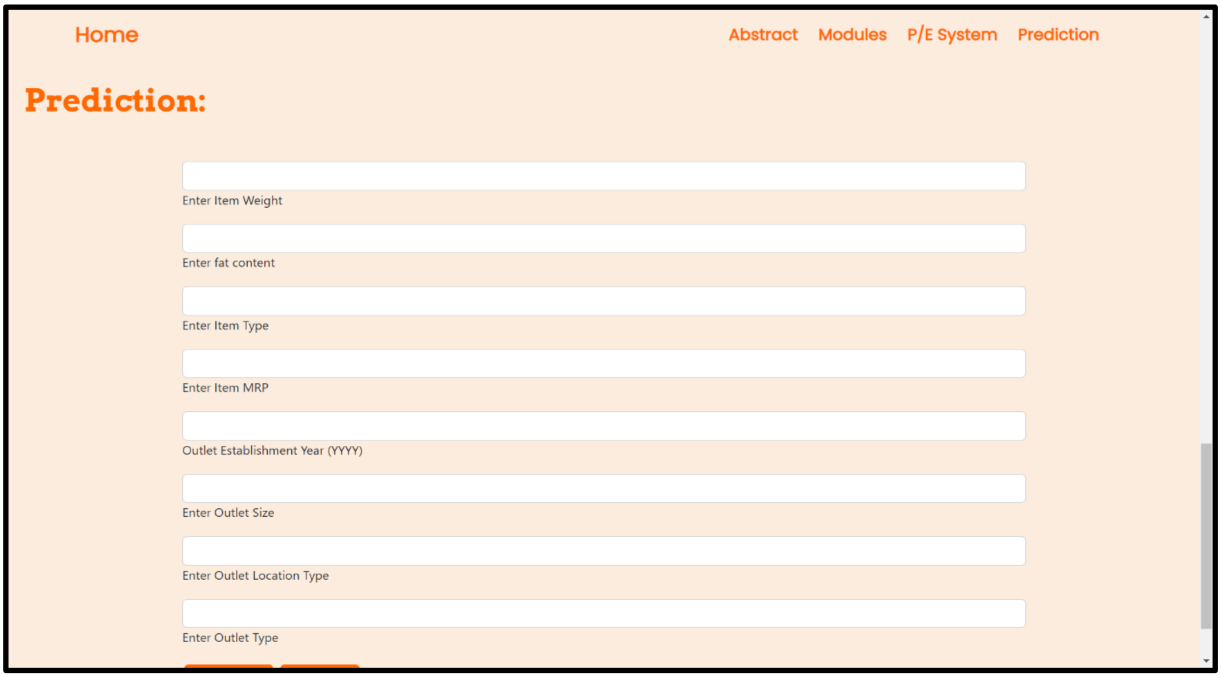
behind scene activities that occur when performing any action on a website.It can be an account login or making a purchase from an online store.

Existing System

Although there are plenty of Machine learning model developed for "SALES FORECASRING",the usage and scope of the model is not upto the level of expectation and demand satisfying as many models referred by us makes use of common,well-known machine learning algorithms such as Linear Regression,Logistic Regression and K-Nearest Neighbor's accuracy of the model is not considered much as it delivers a very low performance metrics,by considering this is an impartment case,it becomes a reason for developing a better model and user interactive.

Proposed System

Here in this project,we put forward a inception of "Flask".A framework developed in python,rather than making a machine learning model that predicts the outlet sales,we find it would be better if our work is published and a platform where users can make use of our project that is deploying the machine learning model in an web app that delivers a content about the title and the current demand ,project insights such as modules and its derivative,languages used,knowledge gathered and more on.Digging down deeper we introduced a never seen machine learning algorithm by us namely "XGBoost".The reason behind selecting this model is,most of the machine learning algorithm works and tries to refine itself based on accuracy but XGBoost revamps itself based on error.So,we find it would be quite interesting to feed the data to the model.



Prediction:

Enter Item Weight

Enter fat content

Enter Item Type

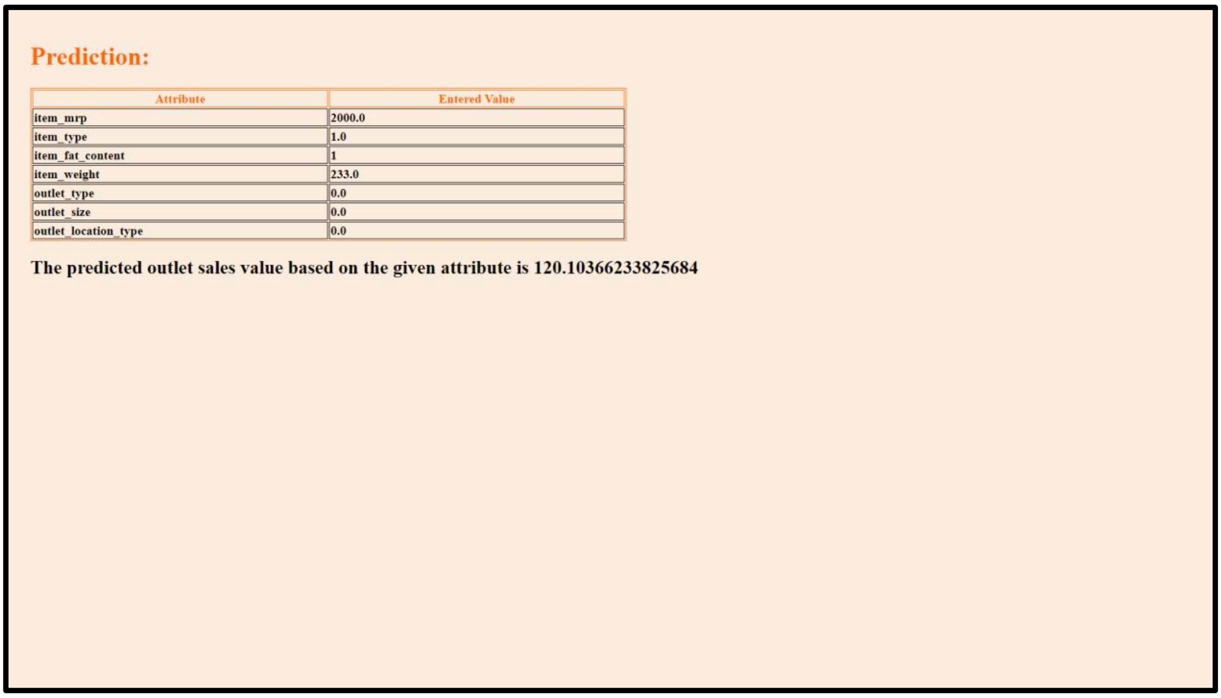
Enter Item MRP

Outlet Establishment Year (YYYY)

Enter Outlet Size

Enter Outlet Location Type

Enter Outlet Type



Prediction:

Attribute	Entered Value
item_mrp	2000.0
item_type	1.0
item_fat_content	1
item_weight	233.0
outlet_type	0.0
outlet_size	0.0
outlet_location_type	0.0

The predicted outlet sales value based on the given attribute is 120.10366233825684