E commerce Shipping Prediction

1.Introduction

1.1 Project Overview

This project focuses on developing a machine learning model to predict shipping times in the eCommerce industry. By analyzing historical shipping data, the model aims to forecast delivery dates with greater accuracy. This will help businesses streamline their logistics, improve customer satisfaction, and optimize their supply chain management.

1.2 Objectives

The primary objectives of this project are to:

- Develop a predictive model that accurately forecasts shipping times based on various factors such as order details, shipping routes, and historical data.
- Identify key determinants that impact shipping durations and use these insights to enhance operational efficiency.
- Provide actionable insights to eCommerce businesses to improve delivery reliability and customer service.

2.Project Initialization and Planning Phase

The Project Initialization and Planning Phase sets the foundation for our e-commerce shipping prediction project. During this phase, we define the project scope, objectives, and deliverables. Key activities include assembling the project team, establishing timelines, and identifying necessary resources. We also outline the data collection plan, ensuring we gather all relevant data from internal and external sources. This phase ensures we have a clear roadmap and all necessary elements in place to proceed efficiently with the project.

2.1 Define Problem Statement

In the fast-paced world of e-commerce, timely delivery is crucial for keeping customers happy and loyal. However, accurately predicting shipping times can be tricky due to various factors like order volumes, shipping distances, weather conditions, and traffic. Right now, our company struggles to provide reliable shipping time estimates, leading to frustrated customers and operational hiccups. We need to address this problem of inaccurate shipping predictions, which not only erodes customer trust but also ramps up costs due to delays and the need for expedited shipping. By developing a machine learning model to predict shipping times more accurately, we aim to boost customer satisfaction, streamline logistics, and improve overall efficiency.

Ecommerce Shopping prediction Problem Statement Report: Click Here

2.2Project Proposal (Proposed Solution)

To address the challenge of inaccurate shipping time predictions, we propose developing a machine learning model that leverages data from various sources. We'll collect and integrate historical shipping records, customer locations, and real-time weather and traffic information. The model will be trained and fine-tuned to capture the diverse factors influencing shipping times. By optimizing the model and

evaluating its performance with key metrics like accuracy, we aim to significantly improve our shipping time estimates. This solution will enhance customer satisfaction, streamline logistics, and boost operational efficiency, ultimately providing more reliable delivery times and reducing costs associated with delays.

Ecommerce Shopping prediction Project Proposal Report: Click Here

2.3 Initial Project Planning

In the initial project planning phase, we set the foundation for our e-commerce shipping prediction initiative. We define the project scope, objectives, and deliverables, assemble the project team, and assign roles. We establish key milestones and timelines to keep us on track, and identify the necessary resources, including data sources and development tools. Our data collection plan outlines how we will gather historical shipping records, customer data, and real-time weather and traffic information. This phase ensures we have a clear roadmap and all the essentials in place to move forward efficiently and successfully.

Ecommerce Shopping prediction Project Planning Report: Click Here

3.Data Collection and Preprocessing Phase

In the Data Collection and Preprocessing Phase, we gather and prepare all necessary data to build our e-commerce shipping prediction model. We start by collecting historical shipping records, customer locations, and real-time data on weather and traffic from internal and external sources. This data is then cleaned to remove any inaccuracies, inconsistencies, or duplicates. We preprocess the data by normalizing and transforming it into a suitable format for machine learning models. This phase is crucial for ensuring that the data we use is accurate, reliable, and ready for effective modeling, setting the stage for developing a robust predictive system.

3.1 Data Collection Plan, Raw Data Sources Identified, Data Quality Report

In this phase, we gather essential data from various sources, including internal company databases, Kaggle datasets, weather and traffic APIs, and geolocation services. We collect historical shipping records, customer locations, and real-time data on weather and traffic. To ensure our data's accuracy and reliability, we address quality issues such as missing values, duplicates, and inconsistencies by implementing validation rules, standardization, and deduplication techniques. This thorough approach ensures our data is clean, consistent, and ready for building a robust e-commerce shipping prediction model.

Ecommerce Shopping prediction Data Collection Report: Click Here

3.2 Data Quality Report

The Data Quality Report assesses the integrity of our collected data from internal databases, Kaggle, weather and traffic APIs, and geolocation services. We've identified issues like missing shipping dates and inconsistent addresses in our internal data, which we'll fix with validation rules and standardization. Kaggle datasets have duplicates and missing values, addressed through deduplication and imputation. Weather data can be incomplete or delayed, so we'll supplement with extra sources and use caching. Traffic data inconsistencies and API limits will be managed by cross-checking and optimizing API usage. For geolocation data, inaccuracies and missing locations will be corrected with better collection and

validation methods. This report helps ensure our data is reliable and ready for building an effective predictive model.

Ecommerce Shopping prediction Data Quality Report: Click Here

3.3 Data Exploration and Preprocessing

In the Data Exploration and Preprocessing phase, we get to know our data by exploring its structure and quality. We look for patterns, trends, and issues like missing values or outliers, and generate summary statistics to understand the data better. We then clean the data by handling missing values, fixing errors, and removing duplicates. Standardizing, normalizing, and encoding variables ensure consistency, and we perform feature engineering to create useful attributes. This thorough process prepares the data, making it clean and ready for building an accurate predictive model.

Ecommerce Shopping prediction Data Exploration and Preprocessing Report: Click Here

4.Model Development Phase

In the Model Development Phase, we focus on creating and training machine learning models to predict shipping times. We start by selecting suitable algorithms based on our data and objectives. We then split the data into training and validation sets to ensure our models are properly tested. During training, we fine-tune the models and adjust hyperparameters to enhance their performance. We also experiment with different model architectures and techniques to find the best fit. By systematically developing and refining our models, we aim to achieve high accuracy and reliability in our shipping time predictions.

4.1 Feature Selection Report

The Feature Selection Report highlights the process of choosing the most important attributes for our predictive model. We evaluate various features to determine which ones significantly impact shipping time predictions. By analyzing their relevance and performance, we select the key features that enhance the model's accuracy and efficiency. This report ensures that we focus on the most valuable data, simplifying the model while improving its effectiveness and making it more reliable for predicting shipping times.

Ecommerce Shopping prediction Feature Selection Report: Click Here

4.2 Model Selection Report

The Model Selection Report summarizes our choice of the best machine learning model for predicting shipping times. We compare various models based on their performance metrics, such as accuracy and F1 Score, to find the most effective one. The report details each model's strengths and weaknesses and explains why the final choice stands out in terms of reliability and accuracy. This helps ensure that our selected model will deliver the most accurate shipping time predictions, ultimately enhancing our e-commerce operations.

Ecommerce Shopping prediction Model Selection Report: Click Here

4.3 Initial Model Training Code, Model Validation and Evaluation Report

The Initial Model Training Code and Validation Report showcase the early stages of our machine learning model development. We provide a snapshot of the code used to train the model and the performance metrics we've gathered. The report includes a summary of how well the model is performing during initial validation, highlighting key metrics like accuracy and F1 Score. This gives a clear view of the model's effectiveness and areas for improvement, ensuring we're on track to achieve reliable and accurate

shipping time predictions.

Ecommerce Shopping prediction Model Development Phase Template: Click Here

5. Model Optimization and Tuning Phase

In the Model Optimization and Tuning Phase, we refine our machine learning model to enhance its performance. We start by fine-tuning hyperparameters and optimizing the model's code to ensure it runs efficiently. We compare different performance metrics, such as accuracy and F1 Score, to evaluate how well each model performs. This phase is crucial for selecting the best model by balancing accuracy, efficiency, and resource use, ultimately leading to more precise and reliable shipping time predictions.

5.1 Hyperparameter Tuning Documentation

The Hyperparameter Tuning Documentation captures the process of refining our model's hyperparameters to boost performance. It details the various settings we adjusted, such as learning rates and regularization parameters, and the techniques used, like grid search or random search. This document shows how these adjustments improved our model's accuracy and efficiency. By documenting these tweaks, we ensure that our final model is finely tuned for the best possible shipping time predictions.

5.2 Performance Metrics Comparison Report

The Performance Metrics Comparison Report provides a clear overview of how different models stack up against each other. It highlights key metrics like accuracy and F1 Score, showing which models performed best in predicting shipping times. This report helps us understand the strengths and weaknesses of each model, guiding us to select the one that delivers the most reliable and accurate predictions.

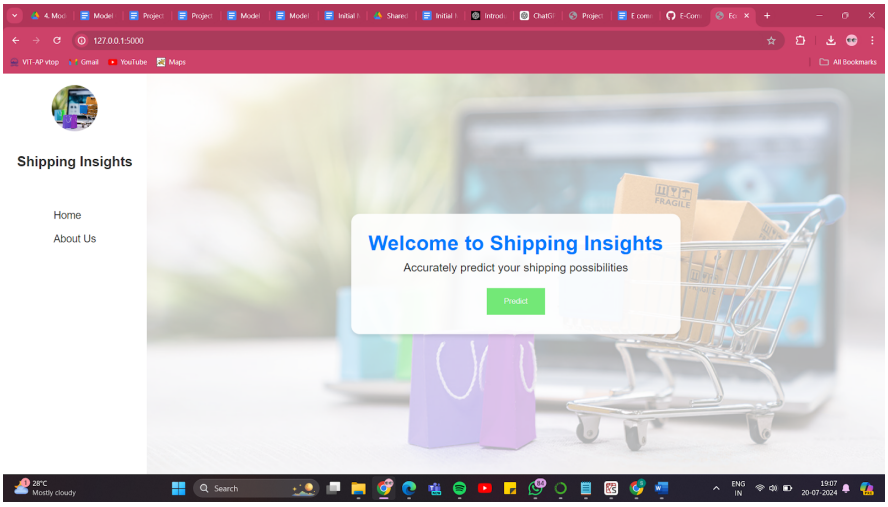
5.3 Final Model Selection Justification

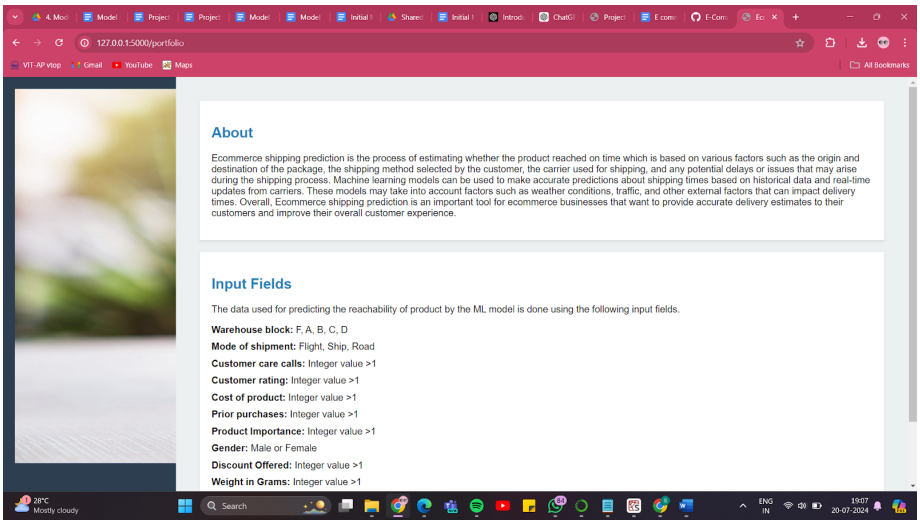
The Final Model Selection Justification explains why we chose our top-performing model. It highlights how the model excelled in key metrics like accuracy and F1 Score, and how it balances performance with efficiency. This rationale ensures that our final choice is not only the most reliable but also the most practical for delivering accurate shipping time predictions.

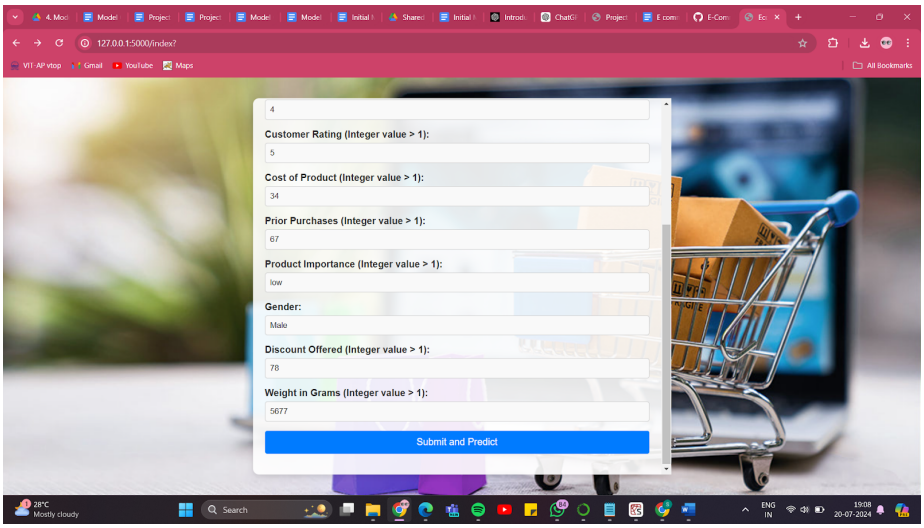
Ecommerce Shopping prediction Model Optimization and Tuning Phase Report: Click Here

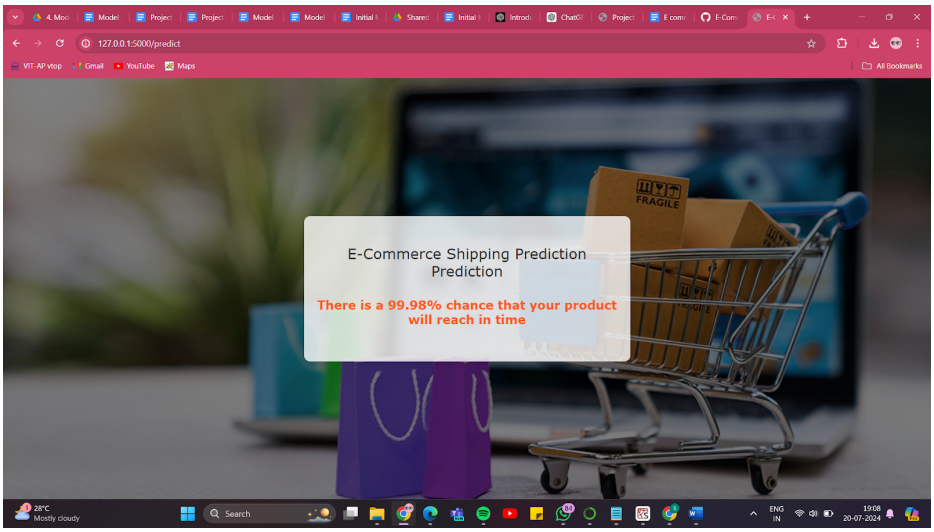
6.Results

6.1 Output Screenshots









7. Advantages & Disadvantages

Advantages

1. Improved Customer Satisfaction: Accurate shipping predictions can enhance customer experience by providing reliable delivery estimates, reducing uncertainty and increasing trust.
2. Operational Efficiency: Businesses can optimize their logistics and supply chain operations by anticipating shipping times, leading to cost savings and better resource allocation.
3. Data-Driven Decisions: Insights gained from the predictive model can inform strategic decisions, helping businesses to identify and address bottlenecks in their shipping processes.
4. Competitive Advantage: Offering precise delivery estimates can differentiate a business from its competitors, attracting more customers and retaining existing ones.

Disadvantages

1. Data Quality and Availability: The accuracy of the predictive model depends heavily on the quality and completeness of the historical data, which may not always be available or reliable.
2. Complexity: Developing and maintaining a sophisticated machine learning model requires specialized knowledge and expertise, which can be resource-intensive.
3. Dynamic Variables: Shipping times can be affected by unpredictable factors such as weather, political events, and transportation issues, which the model might not always account for accurately.
4. Implementation Costs: Integrating the predictive model into existing systems and processes may involve significant initial costs and ongoing maintenance expenses.

8. Conclusion

This project highlights the potential of machine learning to revolutionize shipping predictions in the eCommerce sector. By leveraging historical data, businesses can achieve more accurate delivery estimates, leading to enhanced customer satisfaction and operational efficiency. Despite the challenges, the benefits of implementing such predictive models make it a valuable investment for any eCommerce company aiming to improve its logistics and competitive edge.

9.Future Scope

The future scope of this project includes enhancing the model's accuracy by incorporating real-time data and external factors such as weather and traffic conditions. Additionally, expanding the model to predict other logistics aspects, like inventory management and supply chain disruptions, can further optimize operations. Integration with advanced technologies like IoT and blockchain can also provide more robust and secure predictive capabilities.

10.Appendix

10.1 Source Code

```
import pickle
from flask import Flask , request, render_template
app = Flask(__name__)
model = pickle.load(open("XGBoo_70.pkl","rb"))
```

```
@app.route('/')
def indput():
    return render_template('index.html')
@app.route('/index', methods=['GET','POST'])
def input():
    return render_template('inner-page.html')
@app.route('/portfolio', methods=['GET','POST'])
def portfolio():
    return render_template('portfolio-details.html')
@app.route('/predict',methods = ['GET','POST'])
def admin():
    Warehouse_block=(request.form["Warehouse_block"])
    Mode_of_Shipment=(request.form["Mode_of_Shipment"])
    Customer_care_calls=int(request.form["Customer_care_calls"])
    Customer_rating=int(request.form["Customer_rating"])
    Cost_of_the_Product = int(request.form["Cost_of_the_Product"])
    Prior_purchases = int(request.form["Prior_purchases"])
    Product_importance = (request.form["Product_importance"])
    Gender = (request.form["Gender"])
    Discount_offered = int(request.form["Discount_offered"])
    Weight_in_gms = int(request.form["Weight_in_gms"])

    # Convert categorical features to numeric codes if necessary
    Warehouse_block_mapping = {"A": 0, "B": 1, "C": 2, "D": 3, "F": 4}
    Mode_of_Shipment_mapping = {"Flight": 0, "Ship": 1, "Road": 2}
    Product_importance_mapping = {"low": 0, "medium": 1, "high": 2}
    Gender_mapping = {"Male": 0, "Female": 1}
    Warehouse_block = Warehouse_block_mapping[Warehouse_block]
    Mode_of_Shipment = Mode_of_Shipment_mapping[Mode_of_Shipment]
    Product_importance = Product_importance_mapping[Product_importance]
    Gender = Gender_mapping[Gender]


preds=[[Warehouse_block,Mode_of_Shipment,Customer_care_calls,Customer_rating,Cost_of_the_Produ
ct,
        Prior_purchases,Product_importance,Gender,Discount_offered,Weight_in_gms]]
    xx=model.predict(preds)
    prob=model.predict_proba(preds)[0]
    n_reach = prob[0]
    reach = prob[1]
    prediction_text = 'There is a {0:.2f}% chance that your product will reach in time'.format(reach * 100)
```

```
    print(prediction_text)
    print(xx)
    return render_template("result.html", prediction_text=prediction_text)
    # print('There is a {0:.2f} % chance that your product will reach in time'.format(reach*100))
    # print(xx)
    # return render_template("index3.html",Prediction_result='There is a {0:.2f} chance that your product will reach in time'.format(reach*100))
if __name__ == '__main__':
    app.run(debug='True')
    app.run(port=4000)
```

10.2 GitHub & Project Demo Link

GitHub link: https://github.com/Nihitha-S/E-Commerce-Shipping-Prediction

Project Demo Link:

https://drive.google.com/file/d/1RE_wS00sUEJyDY4jGDZ4NCDxO54vKitk/view?usp=sharing