Back Order Prediction

Revision No:0.1

Last date of revision:29/01/2024

Back Order Prediction

Document Version Control

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Version | Description | Author |
| 29/01/2024 | 0.01 | First Draft | Thomas Antony |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# General Description

ERP systems generate a lot of data (mainly

structured) and also contain a lot of historical data; if this data can be properly utilized, a

predictive model to forecast backorders and plan accordingly can be constructed.

Based on past data from inventories, supply chain, and sales, classify the products as

going into backorder (Yes or No)

## Product Description

Back Order Prediction application for helping merchants in predicting the order that will be going to back order

## Problem statements:

Inventory management

product merchandising

Customer satisfaction

## Technical Requirements

* IDE
* API

## Data Requirements

Historical data of orders that have gone to back order .This data will generally be tabular data generated by ERP systems

## Tools used

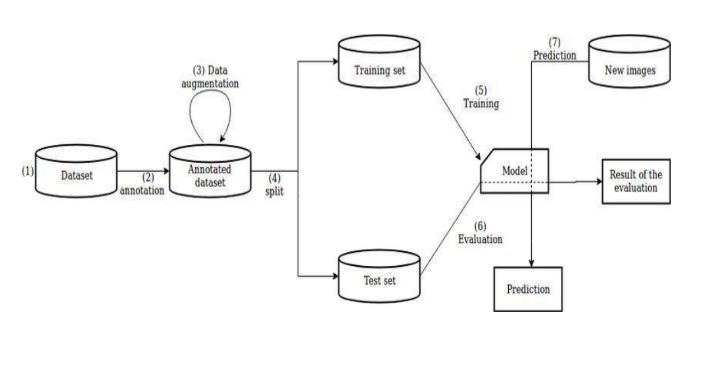
* Python
* Numpy
* Pandas
* Scikit learn
* Flask
* VS code (IDE)
* git
* mlflow
* dvc

## Hard ware requirements:

PC

# Design Details

## Model Training and Evaluation



## Process Flow

As the problem statement is a classification task specifically binary classification we will use Supervise Machine Learning models

Predict new data

Take necessary actions

Training and Testing on dataset

ML model for binary classification

Data Ingestion from database/local file directory

## Data and Model Versioning

## DVC

The data must be version control to identify data drifts that could potentially influence model performance and DVC help us to keep versions of the data we are only tracking the hash key of the current version and the cache and other files are not being tracked

## MLFLOW

MLFLOW help us to log the model , evaluation metrics and parameters used so that we can track different model versions and ml flow and also help us to compare different models and to select the best model .This will also help us to track model drift

## Event Log

The system should log every event so that the user will know what process is running internally and where has the systems produced errors

**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. Developer can choose **logging** method. You **can** choose database logging**/** File

logging as **well**.

4. System **should** not hang even **after using so many** loggings. Logging **just because**

we can easily debug issues so logging **is** mandatory to do.

## 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.

We have created a custom exception function for extra ease of exception handling

## **Performance**

The Back Order prediction gives a quite a good accuracy but as there was quite the class imbalance the precision for predicting back order are low but has a higher recall meaning the predictions we are making are covering most of the back order predictions but not all

### Reusability

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

### 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.

### Resource Utilization

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

**Conclusion**

The back order prediction application performs quite well and clients can use it for inventory management, customer satisfaction and product merchandising etc