Predicting Warehouse Location of online shopping platforms with Machine Learning Algorithm – A Case Study

Hrithik T H[1], K Deepa[1], S.V. Tresa Sangeetha[2]
[1]Department of Electrical and Electronics Engineering,
Amrita School of Engineering, Bengaluru, Amrita Vishwa Vidyapeetham, India.
[2]Department of Engineering, University of Technology and Applied Sciences - AlMusannah, Sultanate of Oman
BL.EN.U4EEE21013@bl.students.amrita.edu[1], k_deepa@blr.amrita.edu[1]

2024 International Conference on E-mobility, Power Control and Smart Systems (ICEMPS) | 979-8-3503-9439-9/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICEMPS60684.2024.10559337

Abstract – Online shopping has turned into the most convenient way to purchase products as it is more user friendly and the availability of the product is vast and also the range of products that the customers can select is abundant. Due to increasing customers in online shopping platforms, it is necessary for efficient warehouse location selection strategies to meet the increasing demand of the customers. By ensuring that all products are available and cutting down on delivery times, warehouse location optimization can improve customer satisfaction, which in turn boosts sales on e-commerce platforms. Using a dataset containing the required parameters, a Machine Learning algorithm can be deployed to optimize the warehouse location of virtual shopping platforms by analysing the total number of orders in each location. Various algorithms, such as KNN, SVM, and Random Forest algorithm, can be used to predict whether there is a need for a warehouse in a particular location. This study aims to determine which algorithm is best suited for this model.

Keywords – Support Vector Machine, warehouse location, e-commerce.

I. Introduction

In this busy world online shopping has become the latest trend in shopping due to the ease of shopping, wider range of products etc. In recent years as the demand for online shopping has increased rapidly, there are a lot of challenges to overcome in order to satisfy the customer's needs and maintain a user-friendly interface. Some of the main factors affecting customer satisfaction are, delivery time and availability of products. Prediction and optimization of warehouse location is vital for reducing the delivery time and delivery. The multi-phase tactical warehouse location-allocation problem involves both quantitative as well as qualitative criteria for making decisions. Organizations that want to prosper in the international marketplace through enhanced supply chain performance must look into the inter-functional elements that influence logistic system optimization. Careful observation reveals that choosing a strategy warehouse location has grown more difficult as there are more options and competing factors [17].

Previous researches have categorised warehouse optimization problems as, understanding fundamental technical layout of a warehouse, The typical organizational and operational structure of a warehousing company, the methods for organizing and managing warehouse activities [8]. Storage Location Assignment can be done with the Multi-objective Optimization utilizing Flower Pollination Algorithm (MOFPA) [16]. A warehouse can utilize the lean stocking process to ensure that no item is overstocked, which ensures better space utilization in the warehouse [4]. By implementing batch picking on picker blocking and determining a suitable batch formation can increase the order picking efficiency [14]. To cut costs overall, the e-commerce sector must optimize vehicle delivery routes based on time slots [5]. If operational handling is not done efficiently, the high frequency of delivery activity from hub to the customer might cause an increase in delivery cost. In metropolitan locations, the Heterogeneous Fleet Vehicle Routing Problem with Time Window model can be used to optimize the last mile delivery route [13]. When there is a time window constraint, the challenge of determining which routes are optimal for a fleet of vehicles to service a group of consumers can be resolved using the ant colony algorithm with k-means clustering [2,3]. For delivery, new tactics and technology are evolving. Order processing times are shortened. The formulation and resolution of routing and inventory routing problems are made more challenging by these new features. So, to solve these problems new method needs to be used [9].

One well-known NP-hard combinatorial optimization issue with multiple variations is the challenge of determining which routes are optimal for a fleet of vehicles to service a group of consumers. CVRPTW (vehicle routing with time windows) can be handled by applying an adaptive reinforcement learning technique based on encoder-decoder deep Q-network. While the decoder is a fully linked neural network, the encoder is based on the attention mechanism [12]. By building a multi-objective mixed-integer nonlinear programming model, it is possible to solve a heterogeneous problem of finding an optimal route for reducing greenhouse gas emission that also takes into account capacity, time-varying speed, and soft time windows all at once. This model not only takes into account minimizing the overall cost of delivery, but also builds a high priority on product delivery from the perspective of the clients in order to increase client satisfaction [15]. Different algorithms such as Linear Regression, SVM, Decision Tree, Random Forest, Ridge, and Lasso Regressor can be deployed for prediction [10,11,1]. The use of machine learning provides a faster and more precise method for predicting [6,7,18].

Previous research has employed various techniques and algorithms such as Genetic algorithms, Ant colony optimization to realign the delivery route and rearrange the products in the warehouse, resulting in quicker product delivery and broad availability. There have also been studies where mathematical models were used. However, these studies needed more data to produce accurate results. Using multiple regression methods, the model aims to precisely and effectively predict whether there is a need for a warehouse in a particular location in order to minimize delivery cost and time thus ensuring customer satisfaction.

With an introduction, there are six sections in this paper. "The proposed system's workings" is covered in Section 2. The suggested system's operation is shown in block diagram form in this section, along with concise descriptions. All of the machine learning models that were employed in this work are detailed in the third section, "Machine learning models".

Analyses and Results make up the fourth section. This section includes a discussion and analysis of the outputs from each machine learning model. Inferences from this work are discussed in the fifth section, "Conclusion." All the cited papers are listed in the "References" section, which concludes this paper.

II. Working on the proposed system

This section gives a quick rundown of how the proposed system operates. Dataset was obtained by analysing ‘Dunzo’ sales and warehouse location at different parts of ‘Bangalore’. The dataset comprises vital information crucial for streamlining delivery operations and improving customer satisfaction. Each entry includes an ‘Order ID’, uniquely identifying each customer's order, along with the ‘Pin code of the delivery location’, pinpointing the exact destination for delivery logistics planning. Additionally, it provides insights into the ‘Total number of orders received on the warehouse from where the customers product was delivered’, aiding in workload management and demand forecasting. The dataset also records the ‘Number of complaints about delivery Time or Product Availability at the customers location. Furthermore, it includes the Distance of Delivery from a Particular Warehouse to the Customer's Location, Lastly, it indicates the Number of Other Alternative Warehouses Available Near the Customer's Location, offering flexibility in routing decisions and optimizing resource allocation based on geographic distribution. This comprehensive dataset facilitates in-depth analysis, allowing businesses to improve the overall delivery experience for customers. This data was recorded for 100 working days.

The main parameters for predicting whether a warehouse is needed in a particular location are distance for delivery, Total orders received on that specific warehouse location and Number of other alternative warehouse that are available near the customers location.

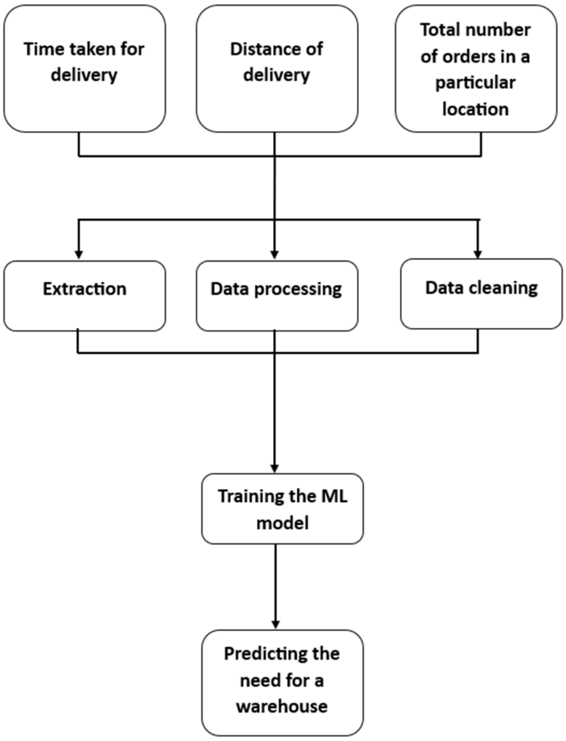


Figure 1 Block diagram of the proposed model

A block schematic of the developed model and its operation is shown in Figure 1. The number of orders at the particular location, the delivery location, and the delivery time are the model's inputs. To eliminate any data that might not be relevant to the model, the dataset must be pre-processed. The procedure doesn't proceed until all required data extraction has taken place. The machine learning model is then trained and used to determine whether a warehouse is required in a specific area.

III. Machine Learning models

The dataset consists of 7 columns. The last column, "New warehouse needed", shows whether a new warehouse is need in a particular location.

A. KNN Regressor

The KNN algorithm performs the functions of a classifier and a regressor. By averaging the target value, a K-Nearest Neighbors (KNN) regressor forecasts the value of a new data point. The new point's distance from every point in the training dataset is determined, and after that, the K nearest neighbors is chosen, and their average is determined. A KNN regressor has been deployed because the objective of this work is prediction.

With an R-squared value of 0.96, the model is well suited for predicting whether a warehouse will be required. Additionally, the MSE and RMSE scores are extremely low, indicating a significantly smaller difference between the predicted and actual values.

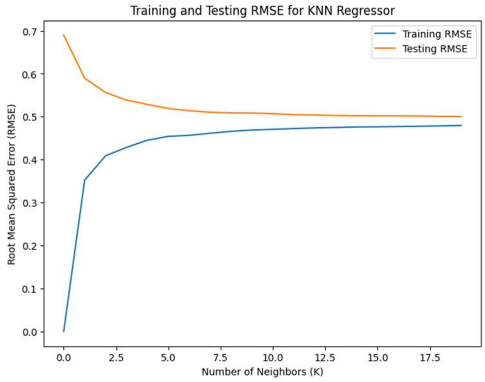


Figure 4: KNN regressor development and appraising loss

Figure 4 displays the development and appraising losses that the KNN regression algorithm attained. Root Mean Squared Error (RMSE) was the selected error for the plot, and the graph was drawn for K in the range of 0 to 20. It is observed that when the number of neighbours increases, the testing losses reduce.

B. Support Vector Regression (SVR)

SVR is a machine learning approach that is based on the Support Vector Machine (SVM) concept but is intended for use in regression and prediction applications. Support Vector Regression (SVR) finds a line that best fits data points, minimizing errors within a margin, and focusing on closest data points, using a kernel function for complex data, adjusted by parameters for optimal prediction performance. To determine whether a warehouse is necessary, SVR was used.

The very low R-R-squared error of -0.42 indicates a poor correlation between the data. Conversely, a low MSE and

MAE suggest less variation between the expected and actual values.

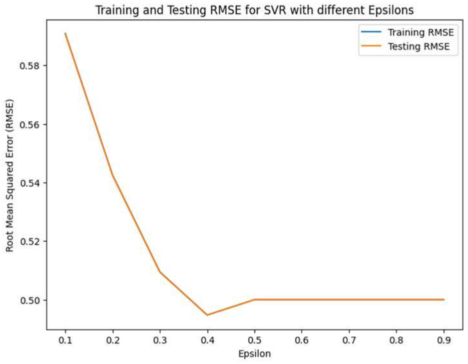


Figure 6 Development and appraising losses for SVR

The SVR algorithm's testing and training losses are displayed in Figure 6. RMSE was chosen as the selected error and it was plotted for epsilon values ranging from 0 to 0.9.

C. *Decision Tree Regressor (DT)*

The DT technique is implemented in areas associated with regression and classification. Since the application requires regression, the Decision Tree Regressor was implemented. Based on input features, a Decision Tree Regressor divides the feature space into segments, each of which predicts the target variable. By selecting the best feature, iteratively dividing the data lowers the variance of the target variable within each group. This procedure continues until a halting criterion is met, such as a maximum depth or a minimum number of samples in a leaf node. The method goes through the tree from the root to a leaf node and outputs the average target value of that node in order to predict the target value for a new data point.

The R-R-squared error is -0.68, as the figure shows. A model that underfits or overfits and is inappropriate for the application can be determined by a negative R-squared error.

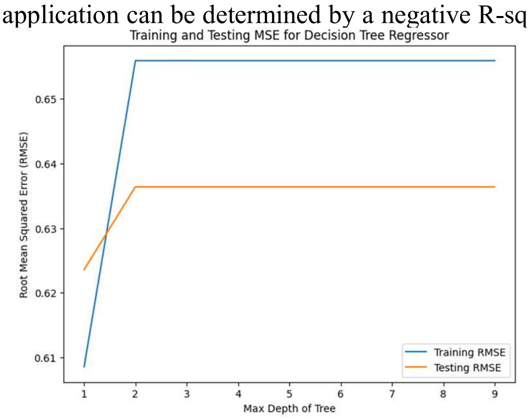


Figure 8 Development and appraising loss of DT algorithm

The DT algorithm's development and appraising loss plot is displayed in Figure 8. A tree's maximum depth, which spans from 0 to 10, is interpreted on the X-axis, and the RMSE values that correspond to these depths are interpreted on the Y-axis.

D. *Random Forest Regressor*

A Random Forest Regressor uses random feature and data subsets to build multiple decision trees. To get the final result, the regressor averages the independent predictions made by each tree about the target variable.

Utilizing averaging, the Random Forest method prevents overfitting and improves prediction precision. Prediction mistakes are decreased by using several decision trees throughout the dataset's subsamples.

The application is perfectly fitted by the model, as evidenced by the R-squared value of 0.99. There is no difference between the expected and actual numbers because the MSE and MAE values were equally found to be zero.

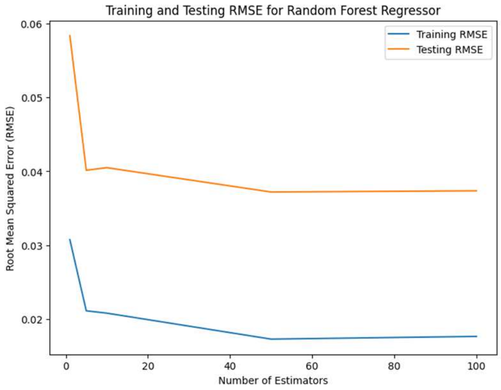


Figure 10 Development and appraising losses for random forest regressor

The development and appraising losses for the Random Forest regressor are displayed in Figure 10. The y-axis displayed the root mean squared error (RMSE), while the x-axis displayed the number of estimators.

E. *Gradient Boost Regression*

It's a machine learning model with regression and classification applications. The Gradient Boosting Regressor constructs decision trees one after another, with each tree aiming to address the mistakes of the preceding ones by focusing on the residuals during its fitting process. It optimizes predictions by iteratively adjusting their direction and magnitude to minimize the loss function. Finally, it combines predictions from all trees to produce the final output.

It was discovered that the MSE and MAE were very low, and the R-squared error was 0.99. As a result, the model fits the requirement for predicting the need for a warehouse.

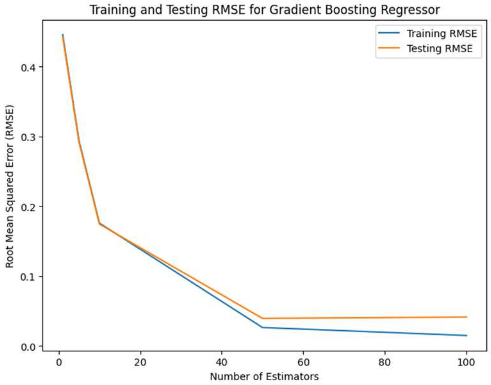
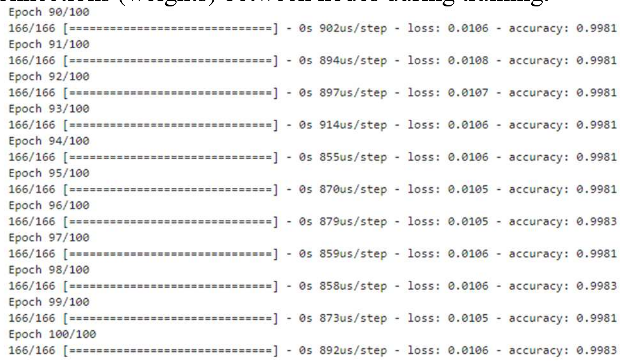


Figure 12 Development and appraising losses for Gradient Boost regressor

The development and appraising losses for gradient boost algorithms are displayed in Figure 12. The graph shows the relationship between a number of estimators and the Root Mean Squared Error (RMSE). It was discovered that as the number of estimators increased, the RMSE values decreased.

F. *Artificial Neural Network (ANN)*

A model based on the neuron networks of the human brain is called an ANN. An ANN's layers consist of an input layer, numerous hidden layers, and an output layer. . Information flows through the network from input nodes, passing through hidden layers where transformations occur, to produce output predictions. ANNs use mathematical operations to adjust the connections (weights) between nodes during training.



```
Epoch 90/100
166/166 [==============================] - 0s 902us/step - loss: 0.0106 - accuracy: 0.9981
Epoch 91/100
166/166 [==============================] - 0s 894us/step - loss: 0.0108 - accuracy: 0.9981
Epoch 92/100
166/166 [==============================] - 0s 897us/step - loss: 0.0107 - accuracy: 0.9981
Epoch 93/100
166/166 [==============================] - 0s 914us/step - loss: 0.0106 - accuracy: 0.9981
Epoch 94/100
166/166 [==============================] - 0s 855us/step - loss: 0.0106 - accuracy: 0.9981
Epoch 95/100
166/166 [==============================] - 0s 870us/step - loss: 0.0105 - accuracy: 0.9981
Epoch 96/100
166/166 [==============================] - 0s 879us/step - loss: 0.0105 - accuracy: 0.9983
Epoch 97/100
166/166 [==============================] - 0s 859us/step - loss: 0.0106 - accuracy: 0.9981
Epoch 98/100
166/166 [==============================] - 0s 858us/step - loss: 0.0106 - accuracy: 0.9983
Epoch 99/100
166/166 [==============================] - 0s 873us/step - loss: 0.0105 - accuracy: 0.9981
Epoch 100/100
166/166 [==============================] - 0s 892us/step - loss: 0.0106 - accuracy: 0.9983
```

Figure 13 Training of ANN model

The training of the ANN algorithm is shown in Figure 13. Fifty epochs were used to train the algorithm. Along with the training loss for every period, the accuracy is shown. The accuracy was discovered to be between 95% and 99%.

The results showed that the MAE was 0.002 and the MSE was 0.001, both extremely low values. As a result, there is less of a difference between the expected and actual values.

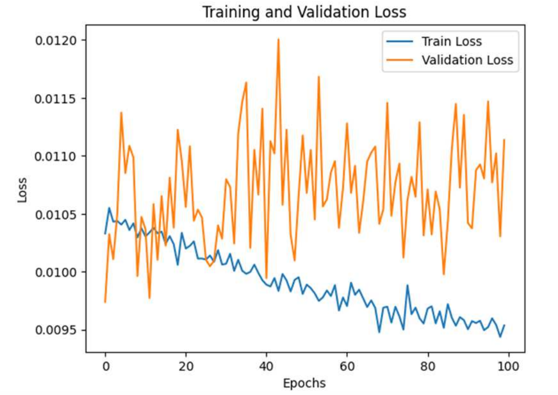
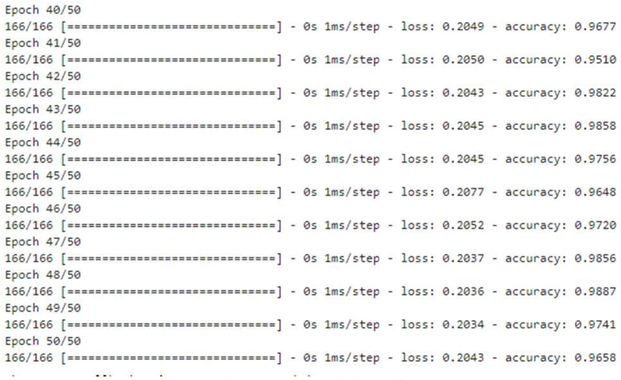


Figure 15 Training and Validation losses for ANN model

The plot of the development and appraising losses for ANN algorithms is displayed in Figure 15. For 100 epochs, the development and appraising history was recorded. Plotted against the number of plots, the Mean Absolute Error (MAE) was the chosen error for this particular plot.

G. *Long Short Term Memory (LSTM)*

As the dataset includes an ID number in its column the data is sequential and neural networks like LSTM can be used. RNNs of the LSTM type are primarily employed in sequence prediction applications. This system is comprised of a cell, forget gate, a gate for input, a gate for output. LSTM networks maintain a cell state to capture long-term dependencies in data. They use gates to control the flow of information, enabling them to selectively update, forget, and output information.



```
Epoch 40/50
166/166 [==============================] - 0s 1ms/step - loss: 0.2049 - accuracy: 0.9677
Epoch 41/50
166/166 [==============================] - 0s 1ms/step - loss: 0.2050 - accuracy: 0.9510
Epoch 42/50
166/166 [==============================] - 0s 1ms/step - loss: 0.2043 - accuracy: 0.9822
Epoch 43/50
166/166 [==============================] - 0s 1ms/step - loss: 0.2045 - accuracy: 0.9858
Epoch 44/50
166/166 [==============================] - 0s 1ms/step - loss: 0.2045 - accuracy: 0.9756
Epoch 45/50
166/166 [==============================] - 0s 1ms/step - loss: 0.2077 - accuracy: 0.9648
Epoch 46/50
166/166 [==============================] - 0s 1ms/step - loss: 0.2052 - accuracy: 0.9720
Epoch 47/50
166/166 [==============================] - 0s 1ms/step - loss: 0.2037 - accuracy: 0.9856
Epoch 48/50
166/166 [==============================] - 0s 1ms/step - loss: 0.2036 - accuracy: 0.9887
Epoch 49/50
166/166 [==============================] - 0s 1ms/step - loss: 0.2034 - accuracy: 0.9741
Epoch 50/50
166/166 [==============================] - 0s 1ms/step - loss: 0.2043 - accuracy: 0.9658
```

Figure 16 Training of LSTM

As seen in Figure 16, the LSTM algorithm was trained over 50 epochs. The accuracy of the LSTM algorithm ranged from 96 to 98%.

The MSE was calculated as 0.075, and the MAE as 0.20. The errors are very small in magnitude, and a R – R-squared value of 0.68 shows that the model is not that suitable for this application.

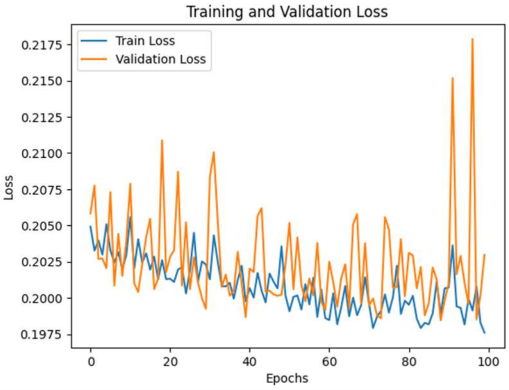


Figure 18 development and appraising loss of LSTM model.

The training and validation losses for the LSTM model are displayed in Figure 18. For 100 epochs, the train and test's history were recorded.

IV. Result and Analysis

Using Python code and the sci-kit-learn and TensorFlow libraries, a variety of Machine Learning (ML) algorithms were implemented, and the corresponding performance metrics were noted. A Minmax scaler from scikit-learn was used to scale the data to a range (0,1). Plots against specific parameters were created, and the errors were examined. The errors obtained for each Machine Learning (ML) algorithm in the proposed model are compared in Table 1. The two best-fit techniques for estimating the need for a warehouse are evident from the table: random forest and gradient boost. Their error values are the lowest, almost at zero. Additionally, their R-R-squared ratings are nearly equivalent to one, demonstrating their excellent application fit. Additionally. Artificial Neural Networks (ANN) offer a high R-squared score and lower MAE and MSE values. Because of its greater complexity and longer computation time, ANN is not chosen over random forest or gradient boost, despite being a good fit as well. While random forest and gradient boosts often take less than 10 seconds, the computation duration of an ANN is approximately one minute for 100 epochs.

TABLE 1: COMPARATIVE ANALYSIS OF PARAMETERS

Model	MSE	R^2	MAE	Accuracy
KNN	0.009	0.96	0.050	---
SVR	0.348	-0.429	0.438	---
Decision Tree	0.405	-0.682	0.405	---
Random Forest	0.001	0.994	0.002	---
Gradient Boost	0.001	0.992	0.003	---
ANN	0.001	0.995	0.002	95 – 99%
LSTM	0.075	0.685	0.202	96 - 98%

Both the KNN and LSTM algorithms are appropriate for this application because of their low MSE and MAE scores. Because KNN has a lower MAE and MSE value than LSTM, it is a preferable option in this case. Their R-R-squared values for KNN are 0.96 and 0.68, respectively, indicating that they offer an acceptable enough correlation among the data. Because SVR has a negative R-squared value of -0.429 and has a relatively high MSE and MAE, it is not a good fit and is not recommended for the application. The Decision tree is the least preferred machine learning model because of its negative R-squared values, which indicate that the model overfits or underfits. It is not a preferred option for estimating the demand for a warehouse because it also has the greatest MSE and MAE values.

V. CONCLUSION

The purpose of this work was to determine whether a new warehouse is required in a specific area by applying several machine-learning models to a dataset including data relevant to online shopping delivery. The two best models for this application are the random forest regressor and the gradient boost regressor, according to an analysis of all the performance characteristics of the deployed models. With the highest R-squared values, they provide the lowest error (MSE and MAE) values. The least recommended models for this type of application, however, are SVR and decision trees, as they have very high error value and low R - R-squared value. While ANN and LTSM have longer computation times than the other ML models, they are still suitable for the application. These models are useful for regression and classification, and they also have a high accuracy score. KNN is a suitable fit.

Machine learning offers predictions with greater accuracy and efficiency, it can be used to determine whether a new warehouse is necessary in a given location. These kinds of systems are becoming more and more necessary to satisfy the growing number of customers. To further improve accuracy in the future, advanced algorithms like deep learning models can be incorporated. These systems have a very broad scope and will be crucial to the growth of the online retail sector in the years to come.

REFERENCES

[1] T. Ganguly, P. B. Pati, K. Deepa, T. Singh and T. Özer, "Machine Learning based Comparative Analysis of Cervical Cancer Risk Classifications Algorithms," 2023 International Conference on Advances in Computing, Communication and Applied Informatics (ACCAI), Chennai, India, 2023, pp. 1-7, doi: 10.1109/ACCAI58221.2023.10200617.

[2] Revanna, Jai Keerthy Chowlur, and Nushwan Yousif B. Al-Nakash. "Vehicle routing problem with time window constrain using KMeans clustering to obtain the closest customer." Global Journal of Computer Science and Technology 22.D1 (2022): 25-37.

[3] Le, Thi Diem Chau, et al. "Clustering algorithm for a vehicle routing problem with time windows." Transport 37.1 (2022): 17-27.

[4] Baro, Manuel & Valdiviezo Castillo, Cinthia & Amaya-Toral, Rosa. (2023). Optimization of a Government Medical Warehouse Using Lean Logistics Methodology. 4. 1-17. 10.37745/bjmas.2022.0084.

[5] G. Kasuri Abhilashani, M. I. D. Ranathunga and A. N. Wijayanayake, "Minimising Last-Mile Delivery Cost and Vehicle Usage Through an Optimised Delivery Network Considering Customer-Preferred Time Windows," 2023 International Research Conference on Smart Computing and Systems Engineering (SCSE), Kelaniya, Sri Lanka, 2023, pp. 1-7.

[6] N. Mahesh, P. B. Pati, K. Deepa and S. Yanan, "Body Fat Prediction using Various Regression Techniques," 2023 International Conference on Advances in Computing, Communication and Applied Informatics (ACCAI), Chennai, India, 2023, pp. 1-5, doi: 10.1109/ACCAI58221.2023.10200647.

[7] S. Sinha and S. R, "An Educational based Intelligent Student Stress Prediction using ML," 2022 3rd International Conference for Emerging Technology (INCET), Belgaum, India, 2022, pp. 1-7, doi: 10.1109/INCET54531.2022.9824636.

[8] Karasek, Jan. (2013). An Overview of Warehouse Optimization. International Journal of Advances in Telecommunications, Electrotechnics, Signals and Systems. 2. 111-117. 10.11601/ijates.v2i3.61.

[9] Archetti, Claudia, and Luca Bertazzi. "Recent challenges in Routing and Inventory Routing: E - commerce and last - mile delivery." Networks 77.2 (2021): 255-268.

[10] R. T. Reddy, P. Basa Pati, K. Deepa and S. T. Sangeetha, "Flight Delay Prediction Using Machine Learning," 2023 IEEE 8th International Conference for Convergence in Technology (I2CT), Lonavla, India, 2023, pp. 1-5, doi: 10.1109/I2CT57861.2023.10126220.

[11] S. J. Selladurai, N. Srivastava and P. B. Pati, "Machine learning analysis of shear stress over a symmetrically stenosed arterial wall under the impact of magnetic field," 2023 2nd International Conference on Vision Towards Emerging Trends in Communication and Networking Technologies (ViTECoN), Vellore, India, 2023, pp. 1-5, doi: 10.1109/ViTECoN58111.2023.10157351.

[12] Gupta, Abhinav, Supratim Ghosh, and Anulekha Dhara. "Deep reinforcement learning algorithm for fast solutions to vehicle routing problem with time-windows." 5th Joint International Conference on Data Science & Management of Data (9th ACM IKDD CODS and 27th COMAD). 2022.

[13] Ayu, Talitha. "Optimizing the heterogeneous fleet vehicle routing problem with time window on urban last mile delivery." IOP Conference Series: Earth and Environmental Science. Vol. 830. No. 1. IOP Publishing, 2021.

[14] Hong, Soondo; Johnson, Andrew L.; and Peters, Brett A., "Analysis of Picker Blocking in Narrow-aisle Batch Picking" (2010). 11th IMHRC Proceedings (Milwaukee, Wisconsin. USA – 2010). 27.

[15] Xiang, Yifei & Zhou, Yongquan & Huang, Huajuan & Luo, Qifang. (2022). Multi-Objective Chimp Optimization Algorithm Based on Genetic Operators for Green Heterogeneous Vehicle Routing Problem. SSRN Electronic Journal. 10.2139/ssrn.4138981.

[16] C. Rungjaroenporn and R. Setthawong, "Multiobjective Optimization Using Flower Pollination Algorithm for Storage Location Assignment at Lazada Thailand Warehouse," 2021 13th International Conference on Knowledge and Smart Technology (KST), Bangsaen, Chonburi, Thailand, 2021, pp. 135-140, doi: 10.1109/KST51265.2021.9415772.

[17] Karmaker, Chitra & Saha, Mou. (2015). Optimization of warehouse location through fuzzy multi-criteria decision making methods. Decision Science Letters. 4. 315-334. 10.5267/j.dsl.2015.4.005.

[18] A. Sowmya and A. S. Pillai, "Human Fall Detection with Wearable Sensors Using ML Algorithms," 2021 2nd International Conference on Smart Electronics and Communication (ICOSEC), Trichy, India, 2021, pp. 1092-1095, doi: 10.1109/ICOSEC51865.2021.9591912.