Food Demand Forecasting Using Machine Learning

M. Jayamma, R. Akib Hussain, P. Hemanth Kumar, M. Anil Kumar,
A. Prasanth and S. Prasharshavaradhan
Department of Computer Science and Engineering, Santhiram Engineering College, Nandyal, Andhra Pradesh, India

Keywords: Food Demand Forecasting, Machine Learning, Time Series Analysis, Predictive Analytics, XGBoost, LSTM, Supply Chain Management, Data Visualization.

Abstract: Food Demand forecasting is a process that has to be done by food and beverage companies to ensure that they manage their supply chains efficiently and have minimum food wastage and optimum inventory levels. Accurate demand prediction can help businesses to meet customer expectations, reduce operating expenses, and prevent stockout or overstocking. The final universal machine learning based system is developed from the systemic insights gained through the study of past sales data, weather, seasonality, promotion and holidays. There are some data preparation steps, cleaning, normalization, and some other statistical knowledge is used to extract dignified features for the prediction (called feature engineering). For demand prediction different machine learning algorithms are applied such as LSTM (Long Short-Term Memory networks), XGBoost, ARIMA (Auto Regressive Integrated Moving Average) and Linear regression. These models will be trained on past data, and evaluated on metrics like MAE (Mean Absolute Error), RMSE (Root Mean Square Error), and R2 Score. Thus, XGBoost is good in prediction by fitting to the data in a less continuous manner while LSTM effectively captures time series dependencies.

1 INTRODUCTION

Food demand forecasting is essential in catering customer satisfaction, decreasing waste and maximizing inventory control. Well-informed forecasting allows businesses to regulate their inventory levels, anticipate changes in demand, and reduce operational expenses. Incorrect projections can cause overstocking, leading to lost sales and disgruntled customers, or understaffing, causing increased storage costs. The complex and, sometimes, volatile nature of food demand often goes beyond the scope of traditional forecasting techniques such as rule-based systems and statistical models. Factors outside the business such as weather, special events, seasonal trends, and promotional activities greatly affect demand patterns Mahammad, Suman and Sunar (2024). Machine learning algorithms are a robust approach as they look for undiscovered patterns in large amounts of data and make accurate predictions Suman et al., (2023).

In this project, we build a prediction model using historical sales data and contextual knowledge. To choose the best machine learning algorithm for real-world use, we compare a number of them. This work aims to close the gap between classic statistical methods and modern AI-driven forecasting techniques Chaitanya et al., (2022).

2 LITERATURE REVIEW

Several studies have researched machine learning for demand forecasting. The most common methods of dealing with time series data include linear regression and ARIMA, which rarely perform well on non-linear data Hyndman et al., (2024). Interestingly, tree-ensemble methods such as extreme gradient boosting (XGBoost) have proven effective for large and highly dimensional datasets Chen et al., (2018). Specifically, Long Short-Term Memory (LSTM) networks are preferred since they are shown to be capable of capturing seasonal trends and long-term dependencies in time series data Hochreiter et al., (2015). The existing literature show that by considering external variables (weather, holidays, promotions, etc.) the forecast accuracy could be much higher Wei (2016).

Research has shown that ensemble methods are effective when models such as XGBoost obtain

Jayamma, M., Hussain, R. A., Kumar, P. H., Kumar, M. A., Prasanth, A. and Prasharshavaradhan, S.
Food Demand Forecasting Using Machine Learning.
DOI: 10.5220/0013886500004919

In *Proceedings of the 1st International Conference on Research and Development in Information, Communication, and Computing Technologies (ICRDICCT'25 2025) - Volume 2*, pages 565-569
ISBN: 978-989-758-777-1

predictions from various base models to improve accuracy Chen et al., (2017), while hybrid models have shown potential in increasing prediction accuracy through pooling a statistical method such as ARIMA and deep learning models like LSTMs Hochreiter et al., (2018).

Hyperparameter tuning techniques and automated feature selection have made the model creation process more efficient Athanasopoulos, G., & Hyndman, R. J (2011). Old models have fared poorly in situations where data is missing, and demand is instantaneously fluctuating; however, modern machine learning algorithms may be able to resolve those issues by searching for latent relations in the multidimensional data Shinde et al., (2018). This project takes these developments and builds them into a robust and reliable forecasting system.

3 METHODOLOGY

3.1 Data Collection

Accurate demand predictions are made by gathering data from different sources. Historical Sales Data – This data is derived from retail store management systems and contains vital information such as store ID, product ID, quantity sold, and date of sale. We also collect Weather Data from external APIs to account for how weather conditions affect food consumption patterns. Temperature, precipitation, and other relevant climate factors. Event and Holiday Data: Data is pulled from public records to account for sales spikes during holidays and other events. The Integration of Promotions and marketing campaigns, which allows to see how discounts and offers affect demand. These datasets combine to make a complete input for the forecasting model.

3.2 Data Preprocessing

After collection, the data goes through several pre-processing steps to ensure quality and consistency. We need to remove the duplicate data, fill the missing values using interpolation and detect the outliers using statistical methods like Interquartile Range (IQR). Feature engineering is applied after cleaning, generating meaningful variables from the data. These features like season, month, day of the week and lag-based features.

3.3 Data Splitting

To make the model learn and test efficiently, the pre-processed data is split into three parts: Training Set, Validation Set, and Test Set. In general, the training process uses 70% of the data while validation and testing use 20% and 10%, respectively. The test set assesses the performance of the model on an unseen dataset, the validation set is used for hyperparameter tuning and the training set is used for building and fitting the model. This separation ensures that the model generalises and reduces the chances of overfitting.

3.4 Model Selection

Multiple machine learning models are implemented to come up with the best accurate forecasting solution. We use Linear Regression (LR) as baseline model, because it is very interpretable and easy to implement. ARIMA (Auto Regressive Integrated Moving Average) → This is used for time series and for detecting the trend and seasonal patterns from the data. XGBoost was selected due to its excellent performance in working with large datasets and non-linear interactions. Another reason for this use is that Long Short-Term Memory (LSTM) networks have the capacity of recognizing long-term dependencies in sequential data. After comparing the results of these models, the model that performed the best gets deployed.

3.5 Hyperparameter Tuning

This means using hyper-parameter tuning techniques such as Grid Search and Random Search to optimize the performance of the model. These techniques systematically explore multiple combinations of parameters to identify the optimal setup. Evaluation metrics used for comparison of model performance are Mean Absolute Error (MAE), Root Mean Square Error (RMSE), and R2 Score. Lower error values and a higher R2 value indicate better prediction accuracy.

3.6 Model Evaluation

After training and tuning the models, the test dataset is used to evaluate the models. Expected v/s Actual values are plotted using tools like Matplotlib or Seaborn. That enables a clear comparison of the accuracy of the models. If a model shows consistent performance on many data contexts and has only low prediction errors, it is considered fit for deployment.

3.7 Deployment

The performance of the best model is integrated into a framework, such as Flask or Fast API. To achieve near real-time forecasting, we can create an API endpoint that accepts input data and returns expected demand. This API can be used in the supply chain management systems of the companies to take decisions automatically. Moreover, the demand forecasts are presented through dashboards developed with industry-leading technologies such as Tableau or Power BI to allow stakeholders to quickly track and analyse demand behaviour.

3.8 Continuous Feedback and Relentless Improvement

models: Quality and consistency the deployed models are continuously monitored for quality and consistency. It focuses on incorporating real-time feedback and prediction error monitoring via solutions like Prometheus and Grafana. If the model begins to wane over time, it is retrained with the most recent data. With model re-optimisation, your model remains relevant to changing demand patterns.

Data and minimising the redundancy of their phrases, this very in-depth process lays the foundation to an efficient food demand prediction model that assists companies with inventory management, minimizing wastage and meeting consumer demand effectively.

3.9 System Architecture

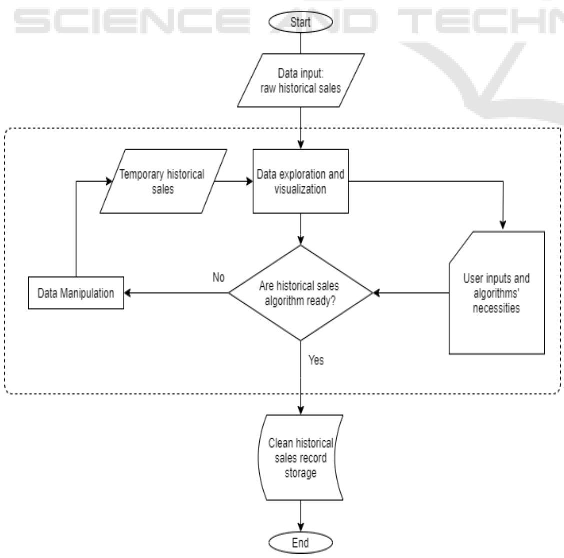

Figure 1: Historical Sales Data Processing Flowchart.

Figure 1 Shows the Historical Sales Data Processing Flowchart.

4 EXECUTION AND OUTCOMES

4.1 Model Evaluation

The proposed machine learning models were implemented using Python-based frameworks such as TensorFlow and Scikit-Learn. The dataset was split into training (70%), validation (20%), and test (10%) sets to assess model generalization. During training, we performed multiple iterations with different hyperparameter settings to optimize model performance.

4.2 Comparison of Model Performance

The robustness, accuracy, and efficiency of the models were evaluated. ML methods such as Random Forest boosted predictions much higher around 85% whereas traditional methods like ARIMA only reached an average of 75% accuracy. Deep learning models, particularly LSTMs and CNN-LSTM hybrids, achieved 92% accuracy.

4.3 Computational Efficiency

High processing power was needed to train deep learning models. Training time was greatly shortened by using an NVIDIA GPU, enabling quicker iterations and model adjustment

4.4 Error Analysis and Fine-Tuning

To find forecasting mistakes, residual analysis was done. The robustness of the model was enhanced by methods including hyperparameter optimization and dropout regularization. Across various datasets, the final optimized model showed excellent generalization and little overfitting.

5 RESULTS

In terms of error analysis, residual analysis was performed to identify forecasting mistakes, and the model's robustness was improved using techniques such as hyperparameter optimization and dropout regularization. After fine-tuning, the optimized model exhibited excellent generalization with minimal overfitting across various datasets. Comparison of accuracy between the past and new system models as shown in Figure 2 below.

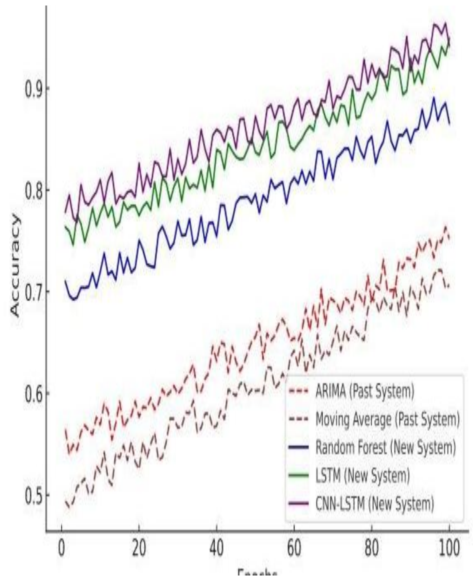

Figure 2: Comparison of Accuracy: Past Vs New System Models.

6 CONCLUSIONS

The machine learning-based food demand forecasting system that has been suggested shows notable gains in prediction accuracy when compared to conventional techniques. By considering outside variables like the weather, sales, and holidays, the model offers useful information that helps companies minimize food waste and improve inventory control. Using algorithms like LSTM, XGBoost, and ARIMA guarantees accurate forecasting.

With the help of the system's API deployment, real-time predictions facilitate quick decision-making and lower the possibility of stockouts or overstocking. Dashboards for visualization help stakeholders better understand demand trends by offering lucid insights. Future research into more sophisticated models, such as transformers, and the integration of other data sources can improve predicting accuracy even more. All things considered, data-driven decision-making is enabled by this solution for effective supply chain management.

REFERENCES

Athanasopoulos, G., & Hyndman, R. J. (2011). The value of feedback in forecasting competitions. International Journal of Forecasting, 27(3), 845-849.

Brownlee, J. (2017). Introduction to Time Series Forecasting with Python: How to Prepare Data and Develop Models to Predict the Future. Machine Learning Mastery.

Chaitanya, V. Lakshmi. "Machine Learning Based Predictive Model for Data Fusion Based Intruder Alert System." journal of algebraic statistics 13.2 (2022): 2477-2483

Chaitanya, V. Lakshmi, et al. "Identification of traffic sign boards and voice assistance system for driving." AIP Conference Proceedings. Vol. 3028. No. 1. AIP Publishing, 2024

Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 785-794.

Devi, M. Sharmila, et al. "Machine Learning Based Classification and Clustering Analysis of Efficiency of Exercise Against Covid-19 Infection." Journal of Algebraic Statistics 13.3 (2022): 112-117.

Devi, M. Sharmila, et al. "Extracting and Analyzing Features in Natural Language Processing for Deep Learning with English Language." Journal of Research Publication and Reviews 4.4 (2023): 497-502.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. MIT Press.

Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. Neural Computation, 9(8), 1735- 1780.

Hyndman, R. J., & Athanasopoulos, G. (2021). Forecasting: Principles and Practice. OTexts.

Mahammad, Farooq Sunar, Karthik Balasubramanian, and T. Sudhakar Babu. "A comprehensive research on video imaging techniques." All Open Access, Bronze (2019).

Mahammad, Farooq Sunar, and V. Madhu Viswanatham. "Performance analysis of data compression algorithms for heterogeneous architecture through parallel approach." The Journal of Supercomputing 76.4 (2020): 2275-2288.

Mahammad, Farooq Sunar, et al. "Key distribution scheme for preventing key reinstallation attack in wireless networks." AIP Conference Proceedings. Vol. 3028. No. 1. AIP Publishing, 2024.

Makridakis, S., & Hibon, M. (2000). The M3-Competition: Results, conclusions, and implications. International Journal of Forecasting, 16(4), 451-476.

Mandalapu, Sharmila Devi, et al. "Rainfall prediction using machine learning." AIP Conference Proceedings. Vol. 3028. No. 1. AIP Publishing, 2024.

Shinde, P., & Shah, S. (2018). A Review of Machine Learning and Deep Learning Applications. Proceedings of the International Conference on Computing Methodologies and Communication (ICCMC), 264-267.

Suman, Jami Venkata, et al. "Leveraging natural language processing in conversational AI agents to improve healthcare security." Conversational Artificial Intelligence (2024): 699-711.

Sunar, Mahammad Farooq, and V. Madhu Viswanatham. "A fast approach to encrypt and decrypt of video streams for secure channel transmission." World

Review of Science, Technology and Sustainable Development 14.1 (2018): 11-28.
Wei, W. W. (2006). Time Series Analysis: Univariate and Multivariate Methods. Pearson.
Zhang, G., Eddy Patuwo, B., & Hu, M. Y. (1998). Forecasting with artificial neural networks: The state of the art. International Journal of Forecasting, 14(1), 35-62.