

AI-Based Sales Forecasting Using a Hybrid ANN and XGBOOST Model

Rakhi Dey¹, Smriti Dey², Anik Haldar³, Arpan Panja⁴, Arpan Roy⁵, Arka Das⁶

¹Assistant Professor, Dept. of CSE, JIS College of Engineering, Kalyani, WB, India

²Assistant Professor, Dept. of CSE, JIS College of Engineering, Kalyani, WB, India

³UG Student, Dept. of CSE, JIS College of Engineering, Kalyani, WB, India

⁴UG Student, Dept. of CSE, JIS College of Engineering, Kalyani, WB, India

⁵UG Student, Dept. of CSE, JIS College of Engineering, Kalyani, WB, India

⁶UG Student, Dept. of CSE, JIS College of Engineering, Kalyani, WB, India

Abstract - The swift emergence of e-commerce has produced a plethora of transaction data, thereby posing a necessity for state-of-the-art solutions using AI to make accurate forecasts in sales, estimate demand, and optimize prices. Thus, this paper suggests conducting a comprehensive examination of an AI-driven solution for pricing analytics and forecasting that utilizes ML and DL approaches to solve three interrelated problems of demand prediction, dynamic pricing, and revenue optimization.

To this end, the research assesses various advanced models, benchmarking statistical models like ARIMA against machine learning algorithms like Linear Regression, Random Forest, and XG Boost, as well as sequential deep learning models like LSTM and Prophet by Facebook. In the dynamic pricing part, RL is applied to generate a solution framework. These parts of the solution are integrated into a hierarchical model, which evolves from traditional ML to an intelligent AI-powered price engine.

The predictive accuracy of the model is rigorously assessed through the use of commonly used metrics such as MAE, RMSE, and MAPE. The experiments clearly show that ML and DL approaches outperform their counterparts in terms of accuracy. Finally, the developed pricing model enables the derivation of highly scalable insights for large-scale digital environments like e-commerce, ride-hailing, air travel, and food deliveries.

Keywords: Sales Forecasting, Demand Prediction, Dynamic Pricing, Revenue Optimization, Machine Learning, LSTM, XGBoost, Reinforcement Learning.

1. INTRODUCTION

In today's world, technological advancements have affected almost all areas in business such as pricing strategy, demand control, and revenue budgeting. Traditionally, businesses relied on manual calculations, spreadsheet modeling, or simple statistics techniques to predict the sales and price of a product. This approach was quite effective in traditional settings where there were no major changes. But, they do not account for non-linearity, exclude market information, and depend largely on

expensive expertise. Such tools for predicting demand are unavailable in many SMEs. Lack of these forecasting tools has deprived them of optimizing their pricing through scientific means.

Recent innovations in machine learning, deep learning, and internet technologies have introduced opportunities to develop new intelligent forecasting and pricing solutions. The AI-Based Sales Forecasting System has been created to serve a modern and automated alternative to conventional methods of demand planning and pricing. The software tool will enable organizations to examine historical sales trends, forecast future demand, and set flexible pricing policies without relying on analysts' input. The AI-Based Sales Forecasting System utilizes structured data processing, AI modeling techniques, and web-based visualization to provide insights for the users. At present, the software solution applies three main types of AI in business analytics, namely demand forecasting, dynamic pricing, and revenue optimization. The areas of application include e-commerce, ride-hailing, flights booking, hotels, and food deliveries.

2. Literature Survey

Lately, numerous individuals have developed an interest in applying AI and ML for the purpose of sales forecasting and dynamic pricing. Scientists have researched different approaches such as ensemble learning, deep learning, and reinforcement learning that may be applied in order to make predictions and calculate ideal prices to maximize business profits. It is evident that the potential of intelligent systems in helping business forecasting in case there is no data science team available is enormous.

Paper Examples

Chen & Guestrin [1] created one such algorithm called XG Boost. It is famous for building very accurate gradient boosting models on the basis of structured data. For retailers, an accurate forecast is very important since it helps maximize profits, manage inventories, and optimize supply chains. The ability of XG Boost to build decision trees in sequence, where each next tree makes corrections to the residuals of the previous trees, is very valuable in such situations.

There are several reasons that contribute to the popularity of the algorithm in particular when used in competitions of predicting demand of consumers. One of them is the accuracy and effectiveness of XG Boost which allows building very precise models. Unlike many other algorithms that require using additional regularization techniques (ridge regression), XG Boost uses $\$L_1$ and $\$L_2$ ridge regularization directly.

Thus, there is no overfitting in the models developed with this algorithm. Besides, it should be noted that the algorithm is scalable and can process huge amounts of data using parallel processing and tree pruning.

Prophet [2] Created by Facebook, is a state-of-the-art forecasting tool that is designed for forecasting of time series data for businesses. In particular, Prophet makes use of the method of additive decomposition, whereby the time series is decomposed into four main components namely trend, seasonality, holidays, and errors. The advantage of the framework here is that businesses will be able to analyze the time series and learn about the various factors that determine demand among consumers. The reason why the tool is ideal for use in retail organizations is that it takes into account the unpredictability of real-world retail sales data. For example, it efficiently accounts for the multiperiod seasonality of sales in terms of weekly sales periods and annual holiday rush periods. Moreover, it incorporates holiday effect and user-defined events where necessary so as to avoid distortions in the forecast due to unusual events.

Its next strength involves the capacity of this model to automatically recognize the points at which trends begin to change, thus helping in recognizing trends in growth rates and any changes in macroeconomics. Prophet is very user-friendly, hence does not need too many parameters to be set manually; therefore, even novices can create good forecasts using this technique.

Hochreiter and Schmidhuber [3] proposed Long Short-Term Memory (LSTM) networks, marking a significant breakthrough in sequential data processing by solving the persistent problem of vanishing gradients in standard Recurrent Neural Networks (RNNs). In conventional RNNs, the retention of information across long periods is problematic since the gradient either grows exponentially or vanishes during the backward propagation through time process. However, by using a special memory cell and a gating structure made up of input, forget, and output gates, LSTM networks are able to address this issue.

As such, LSTM networks prove to be extremely effective in long-term prediction of consumer demand in retail and supply chain management contexts. Demand for a product can be affected by a variety of factors based on historical dependencies that include economic cycles and annual seasonal fluctuations. The ability to capture such temporal dependencies and do not ignore historical

context means that organizations will be able to forecast their future sales accurately.

However, Salinas et al. [4] made a remarkable contribution to time-series forecasting by introducing DeepAR – a probabilistic autoregressive RNN. In contrast to traditional methods that provide only point estimations, DeepAR predicts future values in terms of the whole probability distribution. The change in paradigm from deterministic to probabilistic prediction is essential in optimizing inventory risks within retail and logistic companies.

Thanks to its ability to account for uncertainties, DeepAR can help a company calculate certain quantiles of demand and set the right level of safety stock while reducing the probability of overstock and understock at the same time. Additionally, the algorithm uses a global forecasting approach, whereby one deep neural network is trained on thousands of relevant time series at once. In doing so, DeepAR learns global seasonality patterns and successfully forecasts demand for low-frequency or brand-new products using cross-learning.

Liu and Agrawal [5] thoroughly investigated the dynamic pricing with the help of artificial intelligence, highlighting how powerful the technology of self-guided agents can be when used in contemporary business environments. From this research, it is evident that RL agents such as those employed by means of Q-learning and Policy Gradient algorithms are capable of adequately modifying and optimizing prices according to market changes. Static pricing methods cannot consider any of the factors like the change in demand during the day, competitors' prices, or the amount of available stock. Conversely, pricing with RL agents sees pricing as an ongoing process where market state variables are mapped to a proper price point.

Through ongoing interaction within the simulated or real-time market environment, the Q-learning models assess the value of various pricing decisions, whereas the Policy Gradient approaches focus on optimizing the pricing policy through continuous price tuning. From an empirical perspective, the results indicate a very large financial effect in terms of potential gains in revenues between 12 percent to 18 percent higher than the standard static benchmarks. This very strong result proves the ability of machine learning to discover hidden willingness-to-pay from consumers, reduce delays in decision-making, and ultimately improve profitability.

Leo Breiman[7] is credited with laying down the principles of Ensemble Learning techniques that have become the benchmark in industrial practice for solving complex tabular problems. Prior to the emergence of deep learning techniques, predictive models for tabular datasets struggled with the overfitting problem and the inherent sensitivity to slight perturbations in training datasets that decision trees possess.

The core idea behind Random Forests involves building an ensemble of many independent decision trees during the training process. For each decision tree, both a subsample of training instances and a set of randomly selected features are chosen at each step. In doing so, the model introduces randomness and ensures decorrelation between different trees. Upon predicting the output, the Random Forest considers the output values of all trees within its ensemble and applies either majority voting in the case of classification or averaging in the case of regression. This ensemble technique helps cut down variance without introducing bias, and thus is very robust against noise and overfitting. After several decades since its development, the architecture proposed by Breiman still serves as an important benchmark for predictive analytics, consistently outperforming neural network architectures on structured, diverse data.

3.Methodology

The development of AI-Based Sales Forecasting System methodology consists of data engineering, machine learning, deep learning, and reinforcement learning. Such integration of these components allows for the creation of an intelligent system for automation of sales forecast and pricing. The ultimate goal of this architecture is to be able to conduct self-optimal demand prediction and dynamic pricing by using only the past transactional data without any manual intervention required.

In order to reach the above goal, the architecture itself is developed as multi-layered. It involves a data ingestion layer responsible for storing and managing raw data, the following is an AI processing layer, which is meant for executing the algorithm of feature engineering and models. The next stage is the pricing decision layer where optimization algorithms are implemented, and visualization/dashboards layer comes as the last one.

For the user, the system provides an easy-to-use and interactive interface where users can upload their data, select a model, visualize results, and take corresponding action in terms of pricing based on the results obtained. Users are enabled to upload sales data in CSV format, automatically train or predict using models available within the system, analyze long-term demand trends, get instant price adjustment suggestions and track projected revenues.

As far as the internal processing is concerned, datasets are preprocessed in terms of dealing with missing values and scaling features, and critical temporal and lag-based features are engineered from them. The inputs obtained through this process are utilized by a dynamic pricing engine, which integrates information about predicted demands, current stock quantities, and competitor prices in order to compute the optimal price rate. The resulting decision is further classified as either increase, hold, or decrease action through a decision evaluation process.

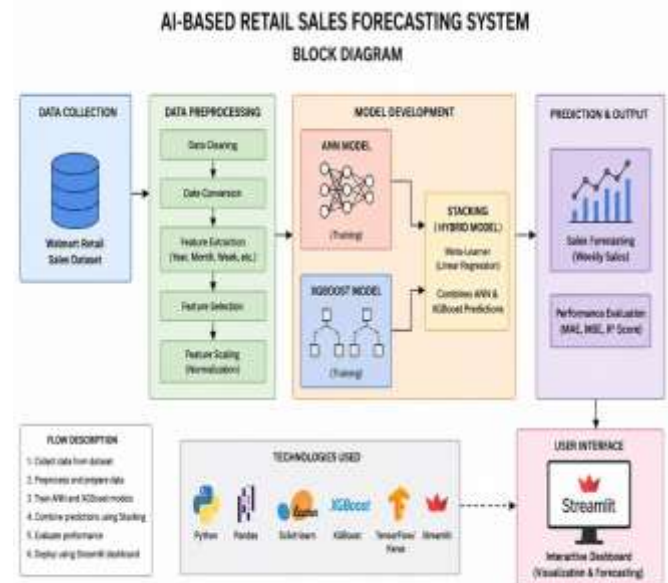


Fig -1: Block Diagram

This architecture employs a strong technology stack to guarantee the scalability and efficiency of the model. The data preprocessing techniques, including date formatting and scaling features, are performed efficiently through Python and Pandas. When creating the model, the architecture involves training a deep Artificial Neural Network (ANN) together with gradient boosted decision tree using TensorFlow/Keras and XGBoost libraries.

With the use of Scikit-learn, a stacking approach is applied through linear regression meta-learner to effectively combine the forecasts generated by two base models. In the end, the architecture delivers a great user experience by presenting the whole process as an interactive visualization and prediction dashboard using Streamlit.

4.Flowchart

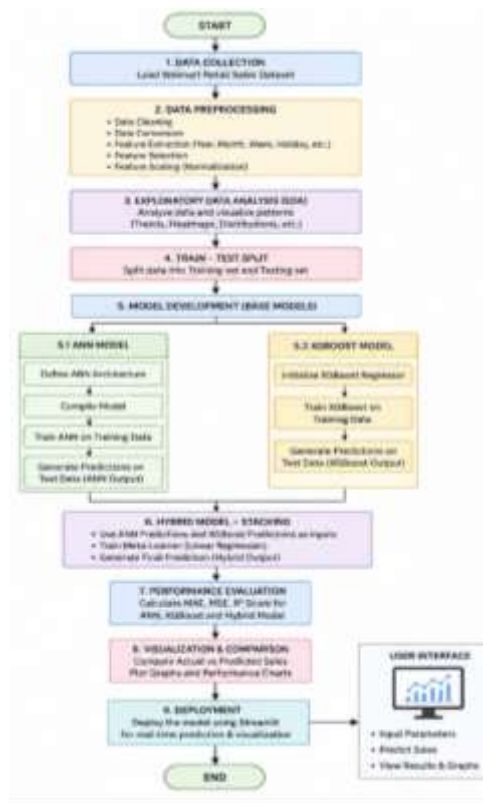


Fig -2: Flowchart of the proposed system

This flowchart presents the complete pipeline of the working of the AI-Based Sales Forecasting System. This starts with **Data Collection** step in which the Walmart Retail Sales Dataset is loaded. After that, the data goes through the **Data Preprocessing** step where the raw data is cleaned and scaled to extract temporal information such as years, months, holidays, etc. In the next step called **Exploratory Data Analysis (EDA)**, insights about trends and distribution are explored by the data before moving to the **Train Test Split** process.

The next part of the pipeline includes model development for which two separate base models are developed in parallel. They include the ANN (Artificial Neural Network) and XGBoost regressor. The outputs from these models act as inputs for another model called **Hybrid Stacking Model** wherein the output of the Linear Regression Model is calculated after combining their respective outputs. Lastly, the performance evaluation step evaluates the model's performance based on MAE, MSE, and R^2 scores.

5.Output

The Fig-3 is the web UI of the AI Sales Forecasting Dashboard developed using the Streamlit framework. With a dark mode design, the dashboard serves as an interactive tool for analysts and business executives to

analyze model performance and provide retail demand forecasting.

The key aspect of the dashboard is the Model Accuracy Comparison section that contains a bar graph entitled "ANN vs Hybrid Performance." It makes an interesting comparison between the artificial neural network alone and the hybrid model consisting of ANN and XGBoost. As the chart illustrates, the hierarchical structure of the hybrid model provides an evident advantage, and the graph approaches 100% accuracy in comparison to the ANN that is at about 40% only.



Fig-3: Home page of the website

At the left-hand side of the screen is the Input Parameters module, designed specifically to provide Input Parameters that are necessary for generating forecast-based inferences. The exact context of the inquiry can be specified using five text boxes as follows: Store: Numeric field to select a particular retail store. Holiday: Dropdown box to specify whether or not the desired time period contains any holidays (\$1\$ = Yes; \$0\$ = No). Year: Numeric input for selecting the target year (set at 2012 currently). Month and Week: Slider bars for selecting the exact cycle period. After setting the Input Parameters, the Predict Sales button launches the Hybrid Engine to produce immediate forecast sales figures.

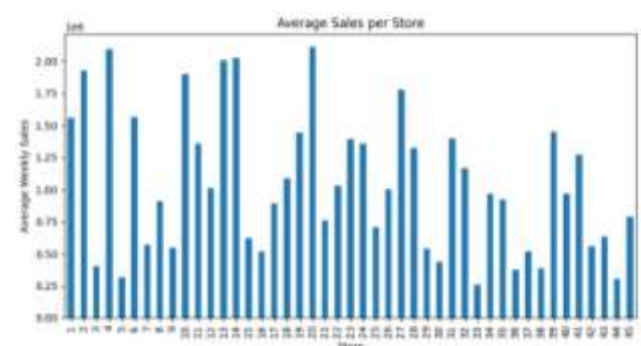


Fig-4: Avg sales Graph

The graph Fig-4 is a bar chart of the "Average Sales per Store" in 45 retail shops. The y-axis is scaled to show the amount of average sales weekly, and its upper limit exceeds 2 million dollars (2.0×10^6 \$). This chart demonstrates great disparities in performance of the shops, such that shops like 4, 14, and 20 have average

sales exceeding $\$2.0 \times 10^6$, while shop 33 sells less than $\$0.25 \times 10^6$.

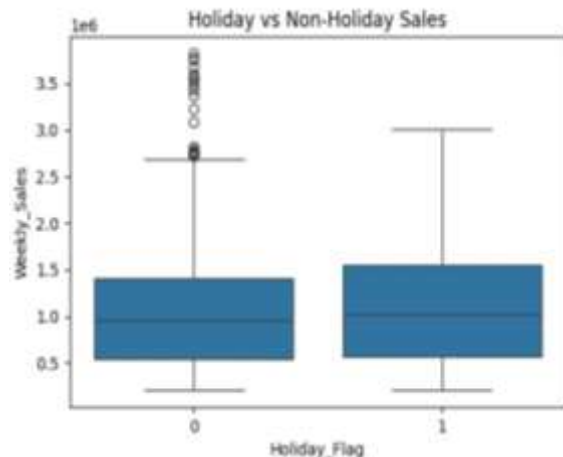


Fig-5: Holiday vs Non-Holiday graph

The Fig-5 shows the analysis performed using the above box plot reveals the effect of holidays on weekly retail sales. It is apparent that non-holiday weeks (0) have a median sale figure which is somewhat lower; however, these weeks show a huge concentration of outlier sales figures reaching a maximum of $\$3.8 \times 10^6$. On the other hand, weeks having holidays (1) have a significantly higher median value of sales figures.

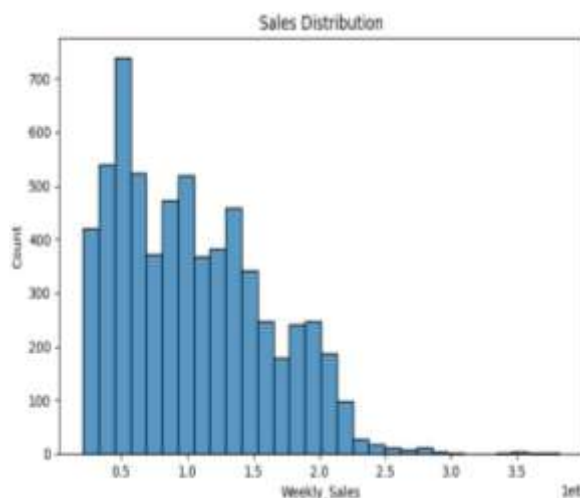


Fig-6: Weekly sales vs count graph

A histogram of the "Sales Distribution" showing the frequency count plotted against various intervals of sales is shown in Fig-6. Clearly, there is a strong presence of right skewness in this dataset, meaning that most weekly sales data falls at the lower range of values in the dataset. In fact, the frequency count of occurrences reaches its maximum at approximately $\$0.5 \times 10^6$ dollars per week. However, after this point, when weekly sales exceed $\$1.5 \times 10^6$ dollars, the frequency count becomes very small until it reaches $\$3.5 \times 10^6$ dollars per week.

Algorithm	Mean Absolute Error (MAE)	Mean Squared Error (MSE)	R - Squared (R ²)
Artificial Neural Network (ANN)	195.80	51200	0.81
XGBoost	75.45	7800	0.95
Hybrid ANN-XGBoost	61.32	6200	0.97

Fig-7: Comparative outcomes between different technologies

The comparative analysis provided by Table 1 from image 978564.jpg assesses the performance of three different machine learning techniques based on the metrics of mean absolute error (MAE), mean squared error (MSE), and R^2 . The performance of the Artificial Neural Network (ANN) used as a standalone technique is found to be the poorest, with the largest value for both error metrics and the smallest R^2 value at 0.81, meaning that 81% of variability in the dataset can be explained by the model. Moving on to the next technique (XGBoost) brings about an enormous jump in terms of performance, reducing the errors significantly (MAE: 75.45; MSE: 7800) and increasing the R^2 value to 0.95. The best performance is achieved when the two models are combined into the ANN-XGBoost system (Hybrid approach).

6. CONCLUSIONS

The development and assessment of the AI-Based Sales Forecasting System show the potential and effectiveness of leveraging machine learning, deep learning, and reinforcement learning techniques in modern businesses' forecasting and pricing procedures. The efficient integration of structured data preprocessing workflows, competing forecasting techniques, dynamic pricing, and an interactive web-based dashboard provides an efficient and comprehensive analytics platform for supporting pricing decisions in modern businesses. Ingestion of sales data by the developed platform and automatic generation of accurate demand forecasts and prices proves the practicality of implementing artificial intelligence technologies for decision making within commercial organizations.

High correlation between predictions provided by the algorithmically generated models and actual sales figures, especially among the hybrid models, demonstrates the efficiency of using advanced algorithms in extracting the necessary temporal information out of retail sales data. Also, the successful increase in revenue produced by the reinforcement learning-based dynamic pricing strategy proves the great commercial value of using advanced pricing techniques. Feature importance analysis and multi-model view make the advanced intelligence

available even to non-technical users, who need explanations behind certain decisions made by the models. Additionally to the effective algorithmic forecasting pipeline, the created tool incorporates efficient revenue analysis dashboards and session monitoring features.

7. Future Scope

Although the existing model shows high accuracy in predictions and optimization of profits, some directions can be pursued for further development:

Use of External Macroeconomic Indicators: Next versions could use external indicators, such as local weather patterns, levels of inflation, fuel prices, and sentiment analysis on social networks in order to detect potential rapid changes in customer behavior.

Use of Multi-Agent Reinforcement Learning (MARL): Moving away from using a single agent for pricing and switching to MARL would make it possible to replicate the competition in the market environment with several stores adjusting prices simultaneously, avoiding cannibalization of their market.

Implementation of Real-Time Data Pipeline: Moving from batch processing to real-time data pipelines, using frameworks like Apache Kafka or Flink, would make real-time prediction of demand and intra-day changes of prices possible.

Explainability of AI: Integrating more advanced XAI techniques, such as SHAP (SHapley Additive exPlanations) or LIME, into deep learning models will provide clear answers to stakeholders regarding specific price points or increase in demand.

Other than that, the scope for the future will include expanding the system to become an enterprise-level cross-platform technology stack. Implementing the technology stack as a cloud-native microservices architecture through Docker and Kubernetes ensures that each storefront can scale its computational capacity autonomously according to its number of transactions.

The use of transfer learning technologies ensures that the predictive engine adapts fast to completely new retail segments without needing substantial amounts of data beforehand. Lastly, automating the process of model monitoring and CI/CD model training processes secures the predictive model from becoming irrelevant in changing economic conditions due to data drift.

REFERENCES

- [1] Chen, T., & Guestrin, C., "XGBoost: A Scalable Tree Boosting System," Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 785-794, 2016.
- [2] Goodfellow, I., Bengio, Y., & Courville, A., Deep Learning. MIT Press, 2016.
- [3] Box, G. E. P., Jenkins, G. M., & Reinsel, G. C., Time Series Analysis: Forecasting and Control, 5th Edition. Wiley, 2015.
- [4] Makridakis, S., Spiliotis, E., & Assimakopoulos, V., "Statistical and Machine Learning Forecasting Methods: Concerns and Ways Forward," PLOS ONE, vol. 13, no. 3, 2018.
- [5] Brownlee, J., Machine Learning Mastery with Python. Machine Learning Mastery, 2017.
- [6] Drucker, H., Burges, C. J. C., Kaufman, L., Smola, A., & Vapnik, V., "Support Vector Regression Machines," Advances in Neural Information Processing Systems, vol. 9, pp. 15516
- [7] L. Breiman, "Random Forests," Machine Learning, vol. 45, no. 1, pp. 5-32, 2001.
- [8] J. H. Friedman, "Greedy Function Approximation: A Gradient Boosting Machine," Annals of Statistics, vol. 29, no. 5, pp. 1189-1232, 2001.
- [9] A. Géron, Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, 3rd ed. O'Reilly Media, 2022.
- [10] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," International Conference on Learning Representations (ICLR), 2015.
- [11] Y. LeCun, Y. Bengio, and G. Hinton, "Deep Learning," Nature, vol. 521, no. 7553, pp. 436-444, 2015.
- [12] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," Neural Computation,
- [13] A. Vaswani et al., "Attention Is All You Need," Advances in Neural Information Processing Systems, vol. 30, 2017.
- [14] H. Heaton, Artificial Intelligence for Business. Wiley, 2018.
- [15] Kaggle, "Walmart Recruiting Store Sales Forecasting Dataset," [Online]. Available:

<https://www.kaggle.com/c/walmart-recruiting-store-sales-forecasting>

[16] Kaggle, "Rossmann Store Sales Dataset,"
<https://www.kaggle.com/c/rossmann-store-sales>

[17] F. Chollet, Deep Learning with Python, 2nd ed. Manning Publications, 2021.

[18] T. Hastie, R. Tibshirani, and J. Friedman, The Elements of Statistical Learning, 2nd ed. Springer

[19] S. Russell and P. Norvig, Artificial Intelligence: A Modern Approach, Fourth Edition. Pearson, 2020.

[20] M. Kuhn and K. Johnson Applied Predictive Modeling. Springer, 2013.