The 14th IEEE International Conference on Dependable Systems, Services and Technologies, DESSERT'2024
11-13 October, 2024, Athens, Greece

The System of Dynamic Optimization Pricing by Machine Learning

Artem Konotopchyk
Department of Computer Engineering and Security
of Lutsk National Technical University
Lutsk, Ukraine
artemkonotopchik2003@gmail.com

Svitlana Lavrenchuk
Department of Computer Engineering and Security
of Lutsk National Technical University
Lutsk, Ukraine
LavrSveet@gmail.com

Pavlo Melnyk
Department of Computer Engineering and Security,
of Lutsk National Technical University
Lutsk, Ukraine
pashamelnik@gmail.com

Kateryna Melnyk
Department of Computer Engineering and Security,
of Lutsk National Technical University
Lutsk, Ukraine
ekaterinamelnik@gmail.com

Nataliia Khrystynets
Department of Computer Engineering and Security
of Lutsk National Technical University
Lutsk, Ukraine,
hrystynets.at.ua@gmail.com

Kateryna Bortnyk
Department of Computer Engineering and Security
of Lutsk National Technical University
Lutsk, Ukraine,
katerina.bortnyk@gmail.com

***Abstract* — Based on the real retail data collected over eight months, a number of models are developed to determine the most efficient machine learning algorithm. It is found that the KNeighbors Regressor model demonstrates the best performance for a small number of transactions, achieving a low MSE of 0.00091 and a high R2 of 0.72 in the validation set. For a large number of transactions, Random Forest Regressor and Decision Tree Regressor show the best ability to capture complex relationships and handle nonlinearties in the data, thanks to the ensemble learning technique. Whereas the Linear Regressor and Support Vector Regressor models demonstrated large deviations from the real price on the test data. The results of the study demonstrate the relevance of rtificial intelligence algorithms in dynamic pricing trategies, showing the ability to quickly process huge amounts of data, taking into account numerous factors and changes in market demand.**

Keywords* — *Dynamic Pricing, Machine Learning, K-Nearest Neighbors Regressor, Random Forest Regressor

I. Introduction

Dynamic pricing is the adjustment of prices to products or services in real time depending on current market conditions to maximise profits. There are three most common pricing methods: based on business costs, based on competitors' pricing decisions, and based on consumer demand [1]. The use of dynamic pricing will allow companies to automate price adjustments, respond to current market demand, use their resources rationally, and segment customers.

However, there is a need to monitor customer reactions to price changes, and there is a risk of a race to the bottom when prices are recklessly cut in response to competitors' demands. One way to overcome these shortcomings is to use dynamic pricing, which analyses data on demand, prices, inventory, customer digital footprints, and website activity [2].

There are two types of dynamic pricing solutions: rule-based and machine learning [3]. Rule-based solutions are characterised by a lack of flexibility to change the environment and respond to unusual or unpredictable events and the need to track duplicate rules when adding new ones or changing existing ones. Machine learning-based systems gain knowledge from data without direct programming and improve their performance as data increases.

Building a pricing model using machine learning provides the following opportunities:

- Based on cluster analysis, to segment customers, build behavioural models and identify customer personality groups [4].
- Take into account a large number of external (seasonality, weather, location) and internal factors affecting the price in real time [5].
- Take into account various performance indicators, such as margin, turnover, or profit maximisation, and inventory optimisation.
- Make forecasts of whether customers will accept the new price of a product or service [6].

.

II. State-of-the-Art

The technologies underlying urban transport platforms such as Uber, Lyft, and DiDi have become one of the most active research topics in computer science, with operations and dynamic pricing algorithms being studied [7].

Dynamic pricing systems are an innovative step in online marketing and are able to respond quickly and efficiently to market changes and anomalies. In [8, 9], the dynamic pricing problem is modelled as a Markov decision process using deep reinforcement learning. This model, combined with an improved reward system, showed good results in dynamic pricing, taking into account seasonality and the history of product prices.

In [10], three machine learning algorithms were tested: gradient boosting (GBM), random forest, and neural networks. It was shown that the GBM model is able to best account for complex relationships and handle nonlinearities and gives the lowest mean square error of 0.012 and the highest R-squared score of 0.92.

Study [11] attempts to predict purchase decisions based on adaptive or dynamic pricing strategies for individual products by integrating statistical and machine learning

models. It emphasises the importance of choosing the right purchase price, not just offering the cheapest option. Unlike previous studies, the paper clusters customers into groups to build the model, not just the history of transactions, which has a positive impact on the formation of purchase predictions.

A detailed overview of research in the field of dynamic pricing based on AI can be found in [12]. The literature review notes that there are no clearly defined algorithms for different use cases, which makes it difficult to compare their efficiency and accuracy. This is due to the different datasets used in each publication. On the other hand, although most of the publications found have the common feature of price discovery, they pursue different goals, such as price forecasting or simulation to determine the best market prices, or pricing strategy.

III. METODOLOGY

The research and development of a dynamic pricing system using machine learning is carried out as shown in Fig. 1.

Fig. 1 Flowchart of the study

A. Data selection and pre-processing

The study was conducted on the basis of real online retail transactions in the UK over an eight-month period [13]. This set contains such characteristics as product and user identifiers, transaction date, unit price and quantity, which allows us to understand the main trends and characterise customer behaviour, taking into account the time component. We created a new column 'MontlyDemand', which contains data on the number of products sold each month. This granularity allows us to better understand the demand for each product and track how it is affected by price changes during the month.

Efficient data preprocessing is essential for building accurate and reliable machine learning models. The following steps were taken to prepare the dataset for analysis:

- Rows with a missing CustomerID were removed to ensure accurate analysis of customer segmentation and behaviour.
- InvoiceDate was converted to date and time format to facilitate time-based analysis.
- Transactions with negative or zero quantities and unit prices have been filtered out, as they do not contribute to the effectiveness of the pricing strategy.
- Several new features have been created to take into account temporal aspects and demand patterns:
    - TotalPrice: Calculated as Quantity multiplied by UnitPrice, which represents the total cost of the transaction.
    - Year, Month, Day, DayOfWeek: Extracted from InvoiceDate to identify date relationships.
    - MonthlyDemand: Calculated as the total number of units sold for each product (StockCode) in each month.

Creating relevant features improves the predictive capabilities of models and increases their performance.

B. Selection of models

Transaction data differs from product to product. The number of records in the database, the frequency of purchases, and the market factors that influenced the creation of the database are crucial in deciding which model to use. In this case, we chose to run the analysis with several models, which allows us to generate a comparative analysis for each product separately. Each model is trained on the basis of each product separately, which leads to a high training speed, but requires a sufficient amount of initial data. This problem is solved in the proposed dataset by having a sufficient number of transaction records in the database for each commodity.

Linear models are particularly valued for their simplicity and interpretability. They assume a direct, proportional relationship between the inputs and the target variable, which makes them useful in situations where this assumption holds true. However, these models may be limited in their ability to capture complex, non-linear relationships in the data. From this category, linear regression has been used, which allows for a linear relationship between attributes and prices, offering a simple approach to forecasting.

Tree models are excellent at handling non-linear relationships and interactions between features by segmenting data into hierarchical branches based on decision rules. These models are powerful for capturing complex patterns in data, although some, like decision trees, can be prone to overfitting without proper hyperparameter tuning. The following tree-based models were selected [12]:

- Decision Tree Regressor - used to capture nonlinear relationships by dividing the data into separate segments based on feature values.
- Random Forest Regressor - used to improve forecasting accuracy and reduce overfitting by combining predictions from multiple decision trees.
- Gradient Boosting Regressor - used to gradually create an ensemble of trees, each of which corrects the errors of its predecessors, which leads to an increase in accuracy.
- AdaBoost Regressor - used to combine several weak models into a more powerful one by focusing on hard-to-predict cases.
- one by focusing on hard-to-predict cases. XGBoost Regressor - chosen for its efficiency and ability to handle large datasets, this model is based on gradient boosting with optimisations for speed and accuracy.

Distance and kernel-based models are effective at capturing relationships by taking into account the proximity of data points or transforming the feature space to better capture non-linear patterns. These models are particularly

useful when the relationship between features and the target variable is not obvious. The following models are included in this category:

- K-Nearest Neighbours (KNN) Regressor - used to predict prices based on the average of the nearest data points, offering a simple but effective method for local pattern recognition.
- Support Vector Regressor (SVR) - used to find the optimal boundary in the transformed feature space, which allows capturing complex, non-linear relationships between features and the target variable.

C. Training and evaluation of models

The dataset was split into training and test samples in an 80/20 ratio. Each trained model was evaluated on the test set using the metrics presented in Table 1:

- Mean Squared Error (MSE) - measures the average squared difference between predicted and actual prices. Lower values indicate better performance.
- Mean Absolute Error (MAE) - measures the average absolute difference between predicted and actual prices, providing a clearer measure of model accuracy.
- R-Squared - shows the proportion of the variance of the target variable explained by the model. Higher values (closer to 1) indicate better performance.

The performance of the models was evaluated and compared. The results demonstrate the highest performance of the KNeighborsRegressor model in terms of accuracy and predictive power. The results are for one type of commodity with a small number of transactions (94).

TABLE 1. PERFORMANCE COMPARISON OF MACHINE LEARNING ALGORITHMS

Model	MSE	MAE	R-squared
LinearRegression	0.0303	0.1241	0.0781
RandomForestRegressor	0.0124	0.0413	0.6234
GradientBoostingRegressor	0.0182	0.0653	0.4466
AdaBoostRegressor	0.0178	0.0496	0.4583
DecisionTreeRegressor	0.0142	0.0331	0.5667
KNeighborsRegressor	0.0091	0.0529	0.7227
SVR	0.0184	0.1089	0.4402
XGBRegressor	0.0142	0.0342	0.568

The results of this study emphasise the effectiveness of machine learning algorithms in complex dynamic environments, in particular the KNeighbors Regressor and Random Forest Regressor models. Table 1 shows that the KNeighborsRegressor model achieved the lowest mean squared error of 0.0091, which indicates the closest match between predicted and actual prices. In addition, the R-squared of 0.72 shows that the model explained 72% of the variance of the target variable, demonstrating its high predictive power.

Such models as Gradient Boosting Regressor and Ada Boosting Regressor showed an average result, which is explained by the specifics of these algorithms and the need for more training data than provided in the dataset.

The analysis of the results of the Linear Regressor and Support Vector Regressor models indicates their inability to detect complex relationships between variables affecting the price. Consequently, the forecasts obtained with these models will be inaccurate and inadequate for this dataset.

Figs. 2-4 show a comparison of model performance for a large number of transactions of the same commodity. The number of transactions in this case is 1727.

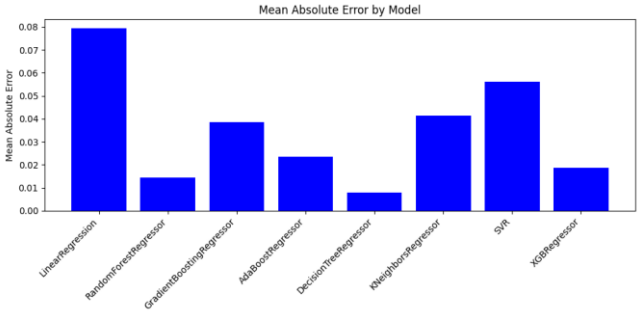

Fig. 2 The value of MAE when training models with a large volume of transactions

Fig. 3 The importance of MSE when training models with a large volume of transactions

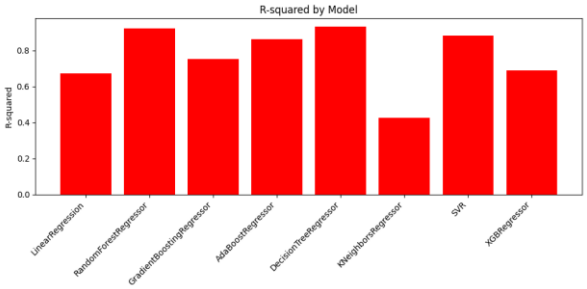

Fig. 4 R-squared value when training models with a large volume of transactions

The superior performance of the Random Forest Regressor and Decision Tree Regressor models can be attributed to its ability to capture complex relationships and handle nonlinearities in the data and to the ensemble learning technique. The hyperparameter tuning process

further optimised the performance of the models, resulting in the lowest MSE and highest R2 among the compared algorithms.

Fig. 5 shows a graph that visualises the difference between the actual and predicted values of the product price.

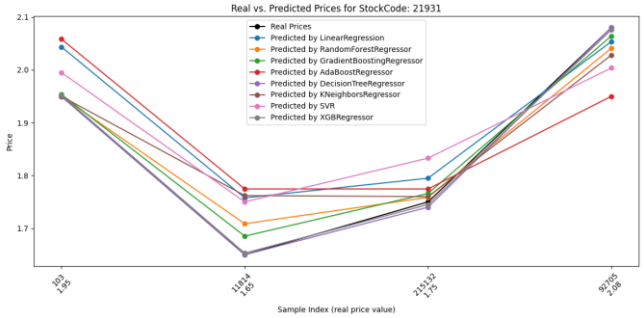

Fig. 5 Comparison of predicted values and the actual price of goods (black line)

D. Discussion of model evaluation results

The study revealed differences in model training errors depending on the following phenomena:

- the ambiguity of the advantage of one model over others for commodities with different numbers of transactions (training data) in the database used;
- dependence on the presence of intense fluctuations in the value of the target variable in the sample.

Each machine learning model used assumes a balanced training sample, however, in the real world where the database was formed, the frequency of purchases differs significantly between different products. This phenomenon has natural reasons, as different types of goods have different purposes and needs for customers. The difference between everyday goods and highly specialised goods has a significant impact on the saturation of the database with sufficient information.

It is also important to note that certain goods have sharp price changes, which affects model training. The results of testing models trained on commodities with a relatively stable price history (the difference between which did not exceed 10-20%) and those with price spikes differ.

In this case, it is advisable to draw a conclusion about which models should be used depending on these two features. The tests revealed that:

- In the case of a small amount of training data (from 50 to 150 training items), the KNeighbors Regressor and Decision Tree Regressor models demonstrated higher accuracy rates.
- In the case of a large amount of training data (500 or more training items), the accuracy of the Random Forest Regressor, Gradient Boosting Regressor, and XGB Regressor algorithms increases;
- In the presence of significant fluctuations between historical commodity prices, the accuracy of all tree models increases, while distance and kernel-based models show significant deviations from real data;
- In the absence of significant fluctuations between price values in the training sample, the accuracy of the models improves and the data has a more stable and linear relationship. Such models are KNeighbors Regressor and Support Vector Regressor. Also, in this case, the accuracy of the basic Linear Regressor improves, which is explained by the fact that it is easier to find a linear relationship between data with small differences.

In real-life situations, these phenomena can be combined, which makes it necessary to consider situations where these phenomena overlap. The testing revealed that in the case of high training data saturation and large differences between historical prices of commodities, the Gradient Boosting Regressor and XGB Regressor models have high accuracy rates. In the opposite conditions, with a small training sample and stable prices, Linear Regressor, KNeighbors Regressor, and Support Vector Regressor demonstrate higher accuracy rates.

As can be seen from Figs. 6-7, the obtained conclusions hold for other commodities with a large and small number of transactions. The MAE values for the eight models are presented.

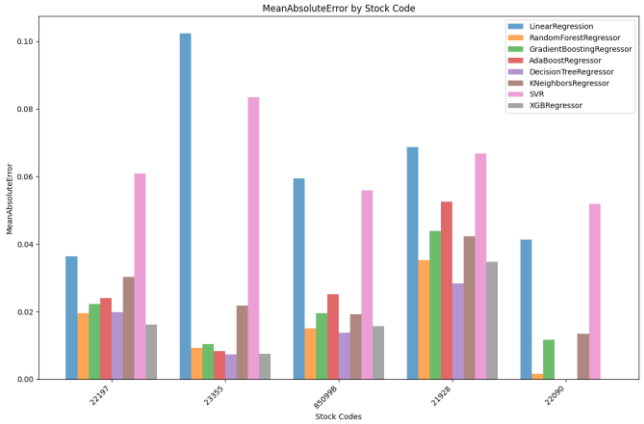

Fig. 6 MAE of models trained on a large data set for five randomly selected products

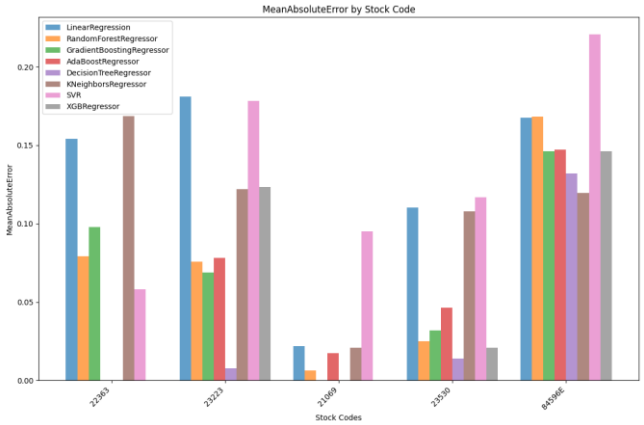

Fig. 7 MAE of models trained on a small sample of data for five randomly selected goods

IV. CONCLUSION AND FUTURE WORK

This study demonstrates the potential of using machine learning methods in dynamic pricing tasks. We have built and analysed 8 machine learning models, with an assessment of the performance of each model.

As a result of the study, a programme was developed that predicts the price of goods depending on such indicators as the quantity of goods, the date of purchase, and the monthly demand for goods.

It was found that KNeighbors Regressor is the most accurate model for a small number of transactions, and Random Forest Regressor and Decision Tree Regressor for a large number, while Linear Regressor and Support Vector Regressor models showed large deviations from the real price on the test data.

Future research on this topic could take into account other features to determine the relationship with price, data saturation, and data uniqueness or difference from each other. Also, the algorithms used can be tuned with different values of hyperparameters, which also affects the results in case of changing the number or type of training features or a different data set. Thus, creating a unified dataset for comparing machine learning algorithms remains relevant.

REFERENCES

[1] Shaulska, L., Pererva, P., Kosenko, O., Marchuk, L., & Hrechanyi, O. The pricing mechanism as a factor in the development of innovative activity management systems in the sphere of electronic commerce. *Bulletin of the National Technical University "Kharkiv Polytechnic Institute" (economic Sciences)*, no. 6, 2023, pp.100–106. Retrieved from http://es.khpi.edu.ua/article/view/307447.

[2] Seleznyova, O. O., Shmagina, V. V., & Yehorova-Hudkova, T. I. Positive and negative aspects of automating pricing in digital marketing. Marketing and Digital Technologies, no. 5(3), 2021, pp. 43-52.

[3] Vasyltsova S., Hasiuk M. (2023). Digitality of pricing methods for determining the optimal price. Bulletin of the National Technical University "Kharkiv Polytechnic Institute" (economic sciences), no. 5, 2023, pp. 71–74. https://doi.org/10.20998/2519-4461.2023.5.71.

[4] Kashwan, Kishana R.; Velu, C. M. Customer segmentation using clustering and data mining techniques. International Journal of Computer Theory and Engineering, 2013, 5.6: 856.

[5] Liman, V., Y. Ivanchuk, R. Yaroschuk. Dynamic pricing in the Ukrainian internet retail. Measuring and computing techniques in technological processes, No. 1, March 2024, p. 231-6, doi:10.31891/2219-9365-2024-77-30.

[6] Panchyshyn, T., Marets, O., & Gayevska, R. Analysis and modeling of customer behavior using Weibul beta distributions. Visnyk of the Lviv University. Series Economics. 2020. Issue 58. P. 99-109. DOI: http://dx.doi.org/10.30970/ves.2020.58.0.5809.

[7] Chen, M. K., & Sheldon, M. Dynamic pricing in a labor market: Surge pricing and flexible work on the Uber platform. 2016. Ec, 16, 455.

[8] Singh A. Introduction to Reinforcement Learning : Markov-Decision Process. *Medium*. URL: http://surl.li/oedddd (date of access: 07.08.2024).

[9] Liu, J., Zhang, Y., Wang, X., Deng, Y. and Wu, X., 2019. Dynamic pricing on e-commerce platform with deep reinforcement learning: A field experiment. arXiv preprint arXiv:1912.02572.

[10] El Youbi, Raouya, Fayçal Messaoudi, and Manal Loukili. "Machine Learning-driven Dynamic Pricing Strategies in E-Commerce." In 2023 14th International Conference on Information and Communication Systems (ICICS), pp. 1-5. IEEE, 2023.

[11] Sarkar, Malay, Eftekhar Hossain Ayon, Md Tuhin Mia, Rejon Kumar Ray, Md Salim Chowdhury, Bishnu Padh Ghosh, Md Al-Imran, MD Tanvir Islam, Maliha Tayaba, and Aisharyja Roy Puja. "Optimizing e-commerce profits: A comprehensive machine learning framework for dynamic pricing and predicting online purchases." Journal of Computer Science and Technology Studies 5, no. 4 (2023): 186-193.

[12] Tomitza, Christoph, Ulvi Ibrahimli, and Lukas-Valentin Herm. "AI-Based Methods of Dynamic Pricing in E-Commerce: A Systematization of Literature." 2024.

[13] Online Retail Data Set. Kaggle: Your Machine Learning and Data Science Community. URL: https://www.kaggle.com/datasets/vijayuv/onlineretail/data (date of access: 08.08.2024).