OPTIMIZING XGBOOST FOR CUSTOMER CHURN PREDICTION IN E-COMMERCE USING GRIDSEARCHCV AND STRATIFIED CROSS VALIDATION

Dwi Hartanti[1*], Aprilisa Arum Sari[2], Sri Lestari[3], Dyah Purwaningsih[4]
Universitas Duta Bangsa Surakarta [1,2,3,4]
*Correspondence Email : **dwihartanti@udb.ac.id**

ABSTRACT

Customer churn prediction is important for e-commerce companies because identifying customers who are likely to leave can help companies develop more targeted retention strategies. This study compares baseline and optimized XGBoost models for customer churn prediction using GridSearchCV and Stratified Five-Fold Cross-Validation. The experiment used an e-commerce churn dataset containing 5,630 customer records, 18 predictor variables, and one binary target variable. Data preprocessing included median imputation, Min-Max scaling for numerical features, Ordinal Encoding for categorical features, and removal of the CustomerID identifier. The data were divided into training and testing sets using an 80:20 stratified split. For model optimization, preprocessing, SMOTE, and XGBoost were integrated into a pipeline, with SMOTE applied only to the training portion of each cross-validation fold. GridSearchCV evaluated 216 hyperparameter combinations, with the best configuration achieving a mean cross-validation F1-Score of 0.8747 ± 0.0155. On the independent test set, the baseline model achieved 99.11% Accuracy, 97.37% Precision, 97.37% Recall, 97.37% F1-Score, and 99.92% ROC-AUC, while the optimized model achieved 98.85% Accuracy, 97.84% Precision, 95.26% Recall, 96.53% F1-Score, and 99.74% ROC-AUC. The optimized model slightly improved Precision but did not outperform the baseline on the other evaluation metrics. These findings show that hyperparameter optimization does not necessarily improve overall test-set performance and should therefore be assessed using an independent test set.

KEYWORDS

Customer Churn Prediction, E-Commerce, GridSearchCV, Hyperparameter Optimization, Stratified Cross Validation, XGBoost

This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International

INTRODUCTION

The rapid expansion of e-commerce has fundamentally transformed consumer purchasing behavior, creating increasingly competitive digital marketplaces (N. Chen & Hu, 2023). In this environment, customer retention has become a critical business objective because retaining existing customers is considerably more cost-effective than acquiring new ones (Maan & Maan, 2023). Despite continuous investments in promotional campaigns and loyalty programs, many e-commerce companies continue to experience customer churn, resulting in revenue loss, increased marketing expenditure, and reduced long-term profitability (Azhar et al., 2025; Maan & Maan, 2023). Consequently, accurately identifying customers who are likely to discontinue using an e-commerce platform has become an essential component of customer relationship management and data-driven business decision making (Rahman & Khan, 2023).

Recent advances in machine learning have significantly improved the capability of predictive analytics for customer churn prediction (Manzoor et al., 2024; Mouli et al., 2024). Traditional statistical models are often limited in capturing complex nonlinear relationships among demographic, behavioral, and transactional customer attributes. In contrast, machine learning algorithms are capable of automatically learning hidden patterns from historical data, resulting in more accurate and robust customer churn prediction models (Imani et al., 2025). Consequently, various supervised learning algorithms, including logistic regression, decision tree, random forest, support vector machine, artificial neural networks, and extreme gradient boosting (XGBoost), have been widely adopted for customer churn prediction in various business domains (Liang, 2023; Manzoor et al., 2024).

Among these algorithms, XGBoost has attracted considerable attention because of its high predictive accuracy, computational efficiency, and robustness against overfitting through regularization mechanisms (T. Chen & Guestrin, 2016). The gradient boosting framework enables XGBoost to iteratively construct decision trees that correct errors produced by previous trees, allowing the model to effectively capture complex nonlinear interactions among customer attributes(Gunasekara et al., 2024; Ouf et al., 2024). Furthermore, XGBoost can naturally accommodate heterogeneous feature types and has demonstrated superior predictive performance in numerous classification problems involving structured tabular datasets (Brito et al., 2024; Gala et al., 2024). Therefore, XGBoost was selected as the classification algorithm in this study because its predictive capability, computational efficiency, and robustness make it well suited for customer churn prediction using structured e-commerce data (Andespa et al., 2025; Brito et al., 2024) Despite the widespread adoption of XGBoost and hyperparameter optimization techniques in customer churn prediction, empirical evidence regarding whether hyperparameter optimization consistently provides meaningful improvements over the default XGBoost configuration remains limited (Ouf et al., 2024).

Although XGBoost consistently achieves competitive predictive performance, its effectiveness largely depends on the selection of appropriate hyperparameter values (Imani et al., 2025). Hyperparameters such as learning rate, tree depth, number of estimators, subsampling ratio, and column sampling ratio directly influence model complexity, learning capability, and generalization performance (Fazli & Mohammadi, 2026; Peng et al., 2023). Previous studies have shown that hyperparameter optimization techniques, including GridSearchCV, Random Search, and Bayesian Optimization, may improve classification accuracy. However, the reported improvements are often inconsistent across

datasets, and relatively few studies have explicitly compared baseline and optimized XGBoost models under identical experimental settings using multiple evaluation metrics. Consequently, the actual contribution of hyperparameter optimization to customer churn prediction remains insufficiently understood (Hakyemez & Adar, 2024; Ouf et al., 2024).

To address these limitations, this study develops a customer churn prediction model based on XGBoost and investigates the effectiveness of hyperparameter optimization using GridSearchCV combined with Stratified Five-Fold Cross Validation. Unlike previous studies that primarily emphasize optimized model performance, this research performs a comprehensive comparison between the baseline XGBoost model and the optimized model using multiple evaluation metrics, including Accuracy, Precision, Recall, F1-Score, ROC-AUC, and Confusion Matrix. This comparative evaluation provides a more objective assessment of whether hyperparameter optimization offers meaningful predictive improvements over the default XGBoost configuration.

The objective of this study is to evaluate the predictive performance of baseline and optimized XGBoost models for customer churn prediction using an e-commerce dataset. Furthermore, this study aims to examine the effect of hyperparameter optimization on predictive performance. Additionally, this study provides empirical evidence on the performance differences between baseline and GridSearchCV-optimized XGBoost models.

RESEARCH METHOD

This study employed a quantitative experimental approach to develop and evaluate an XGBoost based customer churn prediction model for e-commerce. The overall research workflow is presented in Figure 1.

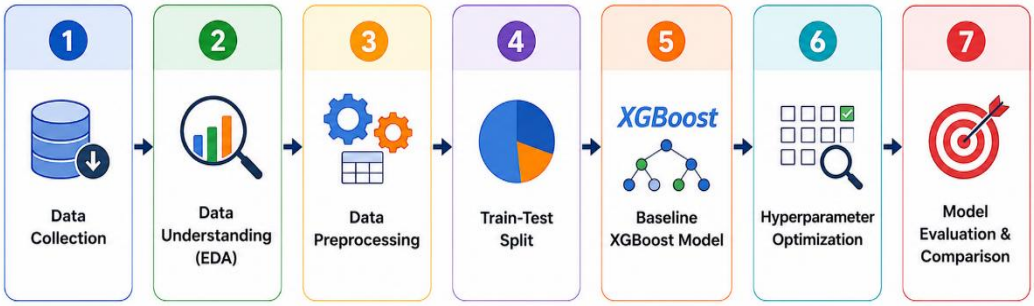


Figure 1. Research Workflow

Figure 1 illustrates the research workflow used in this study. The process begins with the E-Commerce Customer Churn Dataset obtained from Kaggle. The dataset was examined to understand its structure, feature characteristics, missing values, and class distribution.

The next stage is data preprocessing, which includes identifying numerical and categorical features, handling missing values, applying Min-Max scaling to numerical features, and applying Ordinal Encoding to categorical features. The CustomerID attribute was removed because it functions only as a customer identifier and does not provide predictive information. The data were then divided into training and testing sets using an 80:20 stratified split with a random state of 42. During model training, the preprocessing transformations were fitted using the training data, while the testing set was kept separate and used only for final model evaluation.

A baseline XGBoost model was then developed as a reference model without hyperparameter optimization. The optimized model was developed using GridSearchCV with Stratified Five-Fold Cross-Validation. The modeling pipeline integrated

preprocessing, SMOTE, and XGBoost, with SMOTE applied only to the training portion of each cross-validation fold while the validation fold remained unchanged. This procedure was used to reduce the risk of data leakage. GridSearchCV evaluated 216 hyperparameter combinations through five-fold cross-validation, resulting in 1,080 model fits. The configuration with the highest mean F1-Score was selected as the final optimized model.

Finally, the baseline and optimized XGBoost models were evaluated using the independent test set based on Accuracy, Precision, Recall, F1-Score, ROC-AUC, and Confusion Matrix. The performance of both models was then compared to assess the effect of hyperparameter optimization on customer churn prediction.

RESULT AND DISCUSSION

3.1. Dataset Description

The experiments were conducted using the E-Commerce Customer Churn Dataset obtained from Kaggle. The dataset contains 5,630 customer records and 20 attributes, including one customer identifier (CustomerID) and one binary target variable (Churn). Since CustomerID serves only as a unique identifier and does not provide predictive information, it was removed during data preprocessing. After removing this identifier, the dataset contained 18 predictor variables and one binary target variable representing customer churn. The predictor variables represent customer characteristics, purchasing behavior, service usage, and transaction-related information. These variables were used as input features to develop a supervised machine learning model for customer churn prediction.

Table 1. Dataset Characteristics

Characteristic	Description
Dataset Source	Kaggle – E-Commerce Customer Churn Dataset
Number of Records	5.630
Initial Attributes	20
Identifier	CustomerID
Predictor Variables	18
Target Variable	Churn
Target Classes	Non-Churn (0), Churn (1)

3.2.Exploratory Data Analysis

Exploratory Data Analysis (EDA) was conducted to examine the characteristics of the dataset before model development. The analysis focused on the target class distribution, missing values, and general feature characteristics to understand the data before preprocessing.

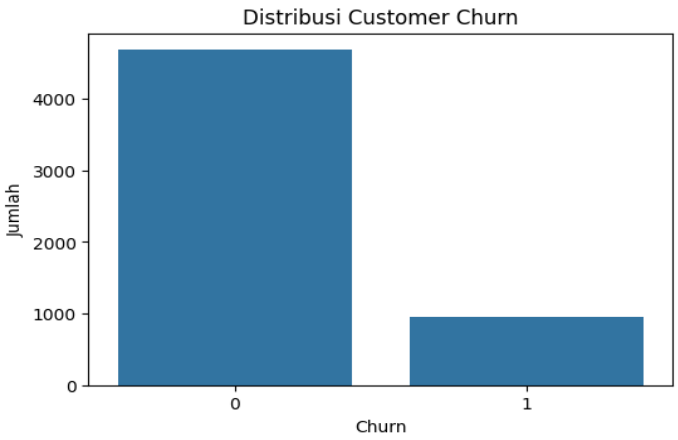


Figure 2. Distribution of Customer Churn Classes

As shown in Figure 2, the dataset consists of 4,682 non-churn customers (83.16%) and 948 churn customers (16.84%). The distribution indicates an imbalanced target variable, with the non-churn class considerably larger than the churn class. To address this imbalance, SMOTE was incorporated into the modeling pipeline during the training and cross-validation process, as described in the research methodology.

Table 2. Customer Churn Distribution

Class	Number of Records	Percentage
Non-Churn	4.682	83.16%
Churn	948	16.84%

Besides the class distribution, several predictor variables contain missing values, as summarized in Table 3. These missing values were handled during the preprocessing stage using appropriate imputation methods based on the feature type.

Table 3. Missing Values

Feature	Missing Values
DaySinceLastOrder	307
OrderAmountHikeFromlastYear	265
Tenure	264
OrderCount	258
CouponUsed	256
HourSpendOnApp	255
WarehouseToHome	251

3.3 Data Preprocessing

Data preprocessing was performed to prepare the dataset for machine learning modeling. The preprocessing procedures included handling missing values, transforming numerical and categorical features, and removing the CustomerID attribute. For numerical features, missing values were handled using median imputation, followed by Min-Max scaling. For categorical features, missing values were handled before applying Ordinal

Encoding. The CustomerID attribute was removed because it functions only as a customer identifier and does not provide predictive information.

The preprocessing transformations were implemented using a ColumnTransformer and were fitted using the training data during model development. After removing CustomerID, the dataset contained 18 predictor variables and one binary target variable (Churn). For the optimized model, SMOTE was integrated into the GridSearchCV pipeline and applied only to the training portion of each cross-validation fold, while the validation and testing data remained unchanged.preprocessing was performed to prepare the dataset for machine learning modeling. The preprocessing included handling missing values, transforming numerical and categorical features, and removing the CustomerID attribute. These steps were carried out to make the data suitable for the XGBoost classification model.

Table 4. Data Preprocessing Procedures

Step	Description
Numerical Missing Values	Median Imputation
Categorical Missing Values	Imputation
Numerical Features	Min-Max Scaling
Categorical Features	Ordinal Encoding
Identifier Removal	Remove CustomerID
Predictor Variables	18 Features
Target Variable	Churn

3.4 Train-Test Split

After the predictor and target variables were defined, the dataset was divided into training and testing sets using an 80:20 ratio. Stratified sampling was applied to preserve the original proportion of churn and non-churn customers. As a result, 4,504 records were assigned to the training set and 1,126 records to the testing set. The testing set was kept separate and used only for final model evaluation.

Table 5. Training and Testing Dataset Partition

Dataset	Records	Percentage
Training Set	4,504	80%
Testing Set	1,126	20%
Total	**5,630**	**100%**

The training set was used for model development, including baseline modeling and hyperparameter optimization. The testing set was kept separate and was used only for the final evaluation of the baseline and optimized XGBoost models.

3.5 Baseline XGBoost Model

A baseline XGBoost classifier was developed as a reference model for evaluating the effect of hyperparameter optimization. The baseline model used the XGBoost classifier with default hyperparameter settings and a random state of 42. The model used the same preprocessing and class-balancing procedures as the optimized model to ensure a fair comparison. The testing set was kept separate and used only for final evaluation.

The baseline model was evaluated using the testing set. Its performance was then compared with the optimized XGBoost model to assess whether hyperparameter optimization improved the prediction of customer churn.

Table 6. Baseline XGBoost Configuration

Parameter	Value
Algorithm	XGBoost Classifier
Objective Function	binary:logistic
Evaluation Metric	logloss
Random State	42
Hyperparameters	Default

The baseline model was evaluated using the testing set, and its results were used as the benchmark for comparison with the optimized XGBoost model.

3.6 Hyperparameter Optimization

Hyperparameter optimization was performed using GridSearchCV combined with Stratified Five-Fold Cross-Validation to identify an effective configuration of the XGBoost classifier. The optimization was implemented using a machine learning pipeline integrating preprocessing, SMOTE, and XGBoost. During each cross-validation iteration, preprocessing transformations and SMOTE were applied only to the training portion of the corresponding fold, while the validation portion remained unchanged. This procedure was used to reduce the risk of data leakage and ensure a consistent evaluation of each hyperparameter configuration.

Six XGBoost hyperparameters were optimized: n_estimators, learning_rate, max_depth, min_child_weight, subsample, and colsample_bytree. The search space consisted of 216 possible hyperparameter combinations. Each combination was evaluated using Stratified Five-Fold Cross-Validation, resulting in a total of 1,080 model fits. F1-Score was selected as the optimization criterion because the dataset has an imbalanced class distribution, making a balance between Precision and Recall important for churn detection. The hyperparameter search space is presented in Table 7.

Table 7. Hyperparameter Search Space

Hyperparameter	Candidate Values
n_estimators	100, 200, 300
learning_rate	0.05, 0.1, 0.2
max_depth	4, 6, 8
min_child_weight	1, 3
subsample	0.8, 1.0
colsample_bytree	0.8, 1.0

GridSearchCV identified the hyperparameter configuration that obtained the highest mean F1-Score across the five validation folds. The selected configuration was subsequently used to construct the optimized XGBoost model. The selected hyperparameter values are presented Table 8.

Table 8. Best Hyperparameter Configuration

Hyperparameter	Best Value
n_estimators	300
learning_rate	0.2

max_depth	8
min_child_weight	1
subsample	1.0
colsample_bytree	0.8

The selected configuration achieved a mean F1-Score of 0.8747 with a standard deviation of 0.0155 across the five folds.

3.7. Model Evaluation and Comparison

The baseline and optimized XGBoost models were evaluated using the independent testing dataset containing 1,126 observations. The evaluation used Accuracy, Precision, Recall, F1-Score, and ROC-AUC. Confusion matrices were also generated to provide a detailed comparison between the predicted and actual churn classes. Table 9 presents the performance comparison between the baseline and optimized XGBoost models.

Table 9. Performance Comparison between Baseline and Optimized XGBoost Models

Metric	Baseline XGBoost	Optimized XGBoost
Accuracy	99.11%	98.85%
Precision	97.37%	97.84%
Recall	97.37%	95.26%
F1-Score	97.37%	96.53%
ROC-AUC	99.92%	99.74%

As shown in Table 9, both models achieved high predictive performance on the independent testing dataset, with Accuracy above 98% and ROC-AUC above 99%. The optimized XGBoost model improved Precision from 97.37% to 97.84%. However, Accuracy, Recall, and ROC-AUC decreased slightly, while the F1-Score decreased from 97.37% to 96.53%. These results indicate that hyperparameter optimization improved Precision but did not lead to improvements across all evaluation metrics.

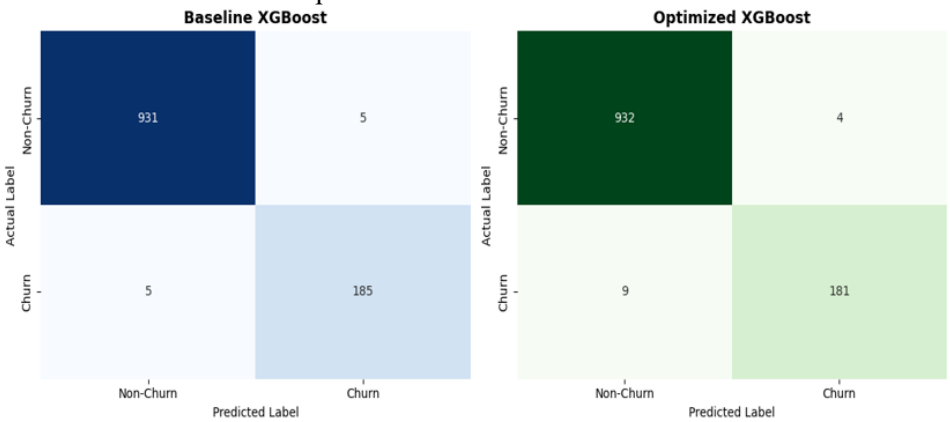


Figure 3. Confusion Matrix Comparison between Baseline and Optimized XGBoost Models

Figure 3 shows the confusion matrices of the baseline and optimized XGBoost models. The baseline model produced 931 true negatives (TN), 5 false positives (FP), 5 false negatives (FN), and 185 true positives (TP). In comparison, the optimized model produced 932 TN, 4 FP, 9 FN, and 181 TP. The optimized model correctly classified one additional non-churn customer and produced one fewer false positive. However, the number of false negatives increased from 5 to 9, while the number of correctly identified churn customers decreased from 185 to 181.

The increase in false negatives indicates that the optimized model missed more customers who actually churned. This is important in churn prediction because false negatives represent churn customers who are not identified by the model and may therefore be missed by retention efforts. Under the experimental setting used in this study, the optimized model improved Precision slightly but did not improve the identification of churn customers compared with the baseline model.

3.8. Discussion

The results show that both baseline and optimized XGBoost models achieved high performance for e-commerce customer churn prediction. The baseline model achieved 99.11% Accuracy, 97.37% Precision, 97.37% Recall, 97.37% F1-Score, and 99.92% ROC-AUC. Meanwhile, the optimized model achieved 98.85% Accuracy, 97.84% Precision, 95.26% Recall, 96.53% F1-Score, and 99.74% ROC-AUC. The optimized model provided a small improvement in Precision, increasing from 97.37% to 97.84%. However, Accuracy, Recall, F1-Score, and ROC-AUC were slightly lower than those of the baseline model. The confusion matrix also shows that false negatives increased from 5 to 9 after optimization, while false positives decreased from 5 to 4. This indicates that the optimized model produced fewer false-positive predictions but missed more customers who actually churned.

The optimized model was selected through GridSearchCV using Stratified Five-Fold Cross-Validation. A total of 216 hyperparameter combinations were evaluated, resulting in 1,080 model fits. The selected configuration achieved a mean F1-Score of 0.8747 with a standard deviation of 0.0155 across the five folds. However, the final test results show that the optimized configuration did not improve overall performance compared with the baseline model. From a practical perspective, both models can support e-commerce companies in identifying customers who may churn. However, the optimized model should not automatically be considered better than the baseline. Model selection should also consider the business objective, particularly the importance of identifying churn customers, customer value, and the cost of retention activities.

CONCLUSION

This study evaluated baseline and optimized XGBoost models for e-commerce customer churn prediction using GridSearchCV and Stratified Five-Fold Cross-Validation. Both models achieved high predictive performance on the independent testing set. The baseline model achieved 99.11% Accuracy, 97.37% Precision, 97.37% Recall, 97.37% F1-Score, and 99.92% ROC-AUC, while the optimized model achieved 98.85% Accuracy, 97.84% Precision, 95.26% Recall, 96.53% F1-Score, and 99.74% ROC-AUC. The optimized model improved Precision slightly and reduced false positives from 5 to 4, but false negatives increased from 5 to 9.

These findings indicate that, under the experimental setting of this study, GridSearchCV-based hyperparameter optimization did not improve all evaluation metrics compared with the baseline model. Nevertheless, both models demonstrated strong performance for identifying potential churn customers. The XGBoost model can be used as a decision-support component to help e-commerce companies identify customers at risk of churn and support targeted retention strategies. Future research can incorporate explainable artificial intelligence methods and evaluate model performance using real-world customer retention outcomes.

ACKNOWLEDGMENT

This research was funded by the Internal Research Grant of Universitas Duta Bangsa Surakarta for the Fiscal Year 2026, under contract number 152/UDB/A.34-SK/VII/2026, dated July 14, 2026. The authors express sincere gratitude to the Institute for Research and Community Service (LPPM) of Universitas Duta Bangsa Surakarta for the institutional support provided throughout this research.

REFERENCES

Andespa, R., Sadik, K., Suhaeni, C., & Soleh, A. M. (2025). Evaluating Random Forest and Xgboost for Bank Customer Churn Prediction on Imbalanced Data Using Smote and Smote-Enn. *Media Statistika*, *18*(1), 25–36. https://doi.org/10.14710/medstat.18.1.25-36

Azhar, M., Saikhu, A., & Buliali, J. L. (2025). Optimization of Predictive Ensemble Model for Customer Churn Through the Integration of SMOTE-ENN and GridSearchCV. *2025 International Conference on Data Science and Its Applications (ICoDSA)*, 1216–1220. https://doi.org/10.1109/ICoDSA67155.2025.11157631

Brito, J. B. G., Bucco, G. B., Heldt, R., Becker, J. L., Silveira, C. S., Luce, F. B., & Anzanello, M. J. (2024). A framework to improve churn prediction performance in retail banking. *Financial Innovation*, *10*(1). https://doi.org/10.1186/s40854-023-00558-3

Chen, N., & Hu, M. (2023). Frontiers in Service Science: Data-Driven Revenue Management: The Interplay of Data, Model, and Decisions. *Service Science*, *15*(2), 79–91. https://doi.org/10.1287/serv.2023.0322

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. https://doi.org/10.1145/2939672.2939785

Fazli, S., & Mohammadi, M. J. (2026). Leveraging gradient boosting machine learning models to predict customer churn in digital health platforms. *Discover Health Systems*, *5*(1). https://doi.org/10.1007/s44250-026-00387-y

Gala, D. M., Pawar, B., Band, G., Dua, P., Mohanty, B. R., & Patil, B. V. (2024). Enhancing Predictive Accuracy for Customer Churn in Digital Banking: A Multi-Model Analysis. *2024 3rd Edition of IEEE Delhi Section Flagship Conference (DELCON)*, 1–5. https://doi.org/10.1109/DELCON64804.2024.10866346

Gunasekara, N., Pfahringer, B., Gomes, H., & Bifet, A. (2024). Gradient boosted trees for evolving data streams. *Machine Learning*, *113*(5), 3325–3352. https://doi.org/10.1007/s10994-024-06517-y

Hakyemez, T. C., & Adar, O. (2024). *Testing the Efficacy of Hyperparameter Optimization Algorithms in Short-Term Load Forecasting*. http://arxiv.org/abs/2410.15047

Imani, M., Joudaki, M., Beikmohammadi, A., & Arabnia, H. R. (2025). Customer Churn Prediction: A Systematic Review of Recent Advances, Trends, and Challenges in Machine Learning and Deep Learning. *Machine Learning and Knowledge Extraction*, *7*(3), 1–38. https://doi.org/10.3390/make7030105

Liang, Z. (2023). Predict Customer Churn based on Machine Learning Algorithms. *Highlights in Business, Economics and Management*, *10*, 270–275. https://doi.org/10.54097/hbem.v10i.8051

Maan, J., & Maan, H. (2023). *Customer Churn Prediction Model using Explainable Machine Learning*. *11*(1), 33–38. http://arxiv.org/abs/2303.00960

Manzoor, A., Atif Qureshi, M., Kidney, E., & Longo, L. (2024). A Review on Machine Learning Methods for Customer Churn Prediction and Recommendations for Business Practitioners. *IEEE Access*, *12*, 70434–70463.

https://doi.org/10.1109/ACCESS.2024.3402092

Mouli, K. C., Raghavendran, C. V., Bharadwaj, V. Y., Vybhavi, G. Y., Sravani, C., Vafaeva, K. M., Deorari, R., & Hussein, L. (2024). An analysis on classification models for customer churn prediction. *Cogent Engineering*, *11*(1). https://doi.org/10.1080/23311916.2024.2378877

Ouf, S., Mahmoud, K. T., & Abdel-Fattah, M. A. (2024). A proposed hybrid framework to improve the accuracy of customer churn prediction in telecom industry. *Journal of Big Data*, *11*(1). https://doi.org/10.1186/s40537-024-00922-9

Peng, K., Peng, Y., & Li, W. (2023). Research on customer churn prediction and model interpretability analysis. *PLOS ONE*, *18*(12), e0289724. https://doi.org/10.1371/journal.pone.0289724

Rahman, M. M., & Khan, M. K. (2023). Mechanisms By Which Ai-Enabled Crm Systems Influence Customer Retention and Overall Business Performance: a Systematic Literature Review of Empirical Findings. *International Journal of Business and Economics Insights*, *03*(01), 31–67. https://doi.org/10.63125/qqe2bm11