International Journal of Innovative Research in Engineering and Management (IJIREM)
ISSN(Online): 2350-0557, Volume-12, Issue-1, February 2025
https://doi.org/10.55524/ijirem.2025.12.1.4
Article ID IJIRE-1372, Pages 28-33
www.ijirem.org

Clickthrough Rate Prediction with Tree-based Machine Learning Models

Blessington Naveen Palaparthi[1], and Dr. S. Akthar[2]

[1]Department of Computer Science and Engineering, Acharya Nagarjuna University, Guntur, India
[2]Department of Computer Science, Government College for Women (A), Guntur, India

Received: 26 December 2024 Revised: 10 January 2025 Accepted: 24 January 2025

ABSTRACT- This paper introduces an intuitive approach to clickthrough rate (CTR) prediction, a learning problem that has been extensively studied over the past several years. As digital marketing continues to grow rapidly into a multi-billion-dollar industry, this study aims to find the most effective machine learning model to enhance the CTR of marketing emails by comparing various tree-based models. Key steps in this research include data collection, feature extraction, and CTR prediction through the evaluation of different models. The statistical results prove that the CatBoost model, with optimized feature selection, achieves near-perfect data fitting, indicating its efficiency.

Keywords- Click Through Rate (CTR), Contextual Advertisements, Email Marketing, Web advertisements, Machine Learning, Decision tree, Logistic Regression, CatBoost Regression, Random Forest Regressor, Extra Trees Regressor, LightGBM, Gradient Boosting

I. INTRODUCTION

Digital marketing is a rapidly expanding industry, valued at approximately $667 billion USD in 2024 and projected to reach $786.2 billion USD by 2026. It offers innovative methods for businesses to engage with potential customers and promote their brands through various online channels, including websites, search engine marketing, search engine optimization, email marketing, organic and paid social media, among others. Channels are classified as direct or indirect based on how they reach the audience. Email marketing, considered a direct channel, can effectively reach both existing and new customers with creatively crafted content. It boasts extensive storytelling capabilities through images, text, and device-specific features. Additionally, email marketing provides robust tracking mechanisms to monitor recipient interactions, such as email open events, clicks on internal URLs or images, and even unsubscribe actions.

Clicking on a URL enables marketers to figure out the most relevant advertisement for each user, thereby enhancing the user experience. The click-through rate (CTR), which measures the ratio of clicks to impressions, is a crucial metric for assessing the commercial value of a campaign. Machine learning has significantly contributed to predicting user activity and behavior. A paper presented by [1] highlights the importance and potential real-world advancements of these techniques. Extensive research has been conducted on CTR predictions using various machine learning models, with each study showing that specific algorithms or their combinations produced optimal results for the datasets used. Chen et al. [2] used a stacking-based fusion model combining Logistic Regression (LR) and Gradient Boosting Decision Trees (GBDT), along with a BP neural network model for deep learning prediction. Their experimental results on a real dataset showed that the deep-learning-based BP neural network model outperformed other models. Richardson et al. [3] employed LR and Multiple Additive Regression Trees (MART) to predict CTR, finding that the LR model outperformed the MART model. Chapelle et al. [4] developed an LR-based machine learning framework specifically for display ad click-through rate prediction. Despite its simplicity, the LR model struggles with non-linear features. To address this, He et al. [5] from Facebook introduced a model that combines GBDT and LR to predict Facebook ad click-through rates effectively.

Yin, Ning et al. [6] introduced a novel model called Coupled Logistic Regression (CLR) for accurate and efficient CTR prediction. CLR leverages all features from ads, users, and context, including their nonlinear interactions, by seamlessly integrating conjunction information using a factorization machine. Experimental results on real-world datasets showed that the CLR model ensures both accuracy and efficiency in large-scale CTR prediction problems.

Zhang, W., Han, Y., Yi, B., et al. [7] concluded that, based on results from the public experimental dataset, the Interactive Attention Rate Estimation Model (IARM) surpasses other recent prediction models in terms of the assessment metrics AUC and LOSS, demonstrating superior accuracy.

Çakmak, Tülin et al. [8] employed the Extreme Gradient Boosting (XGBoost) algorithm for predicting hotel clicks. Their results showed that XGBoost achieved the highest R-Squared values across all metrics used in the study. Additionally, the study aimed to develop sequential models using various architectures of recurrent neural networks for click prediction.

Xia, Zhen et al. [9] conducted experiments using the online ads CTR prediction datasets provided by Huawei and Avazu on the Kaggle platform. Their results showed that the PCSN and PCSNL models outperform traditional CTR prediction models and other deep learning models.

Effendi, Muhammad Junaid et al. [10] aimed to improve the click-through rate (CTR) of contextual advertisements using Linear Regression. The statistical results from their dynamic technique showed high efficiency, with the Linear Regression model fitting the data almost perfectly through optimized feature selection. This research introduced a novel method for predicting CTR for online advertisements using Linear Regression, incorporating a dynamically added feature known as the keyword. Although this approach slightly reduced efficiency, it provided a distinct perspective on calculating CTR. The study found an accuracy of 83%, which could increase to 95% by removing the keyword feature, significantly improving model fit. However, since the research focuses on contextual advertisements, CTR is also dependent on keywords. These findings can guide future research to combine and develop new techniques to enhance the performance of online advertisement serving.

The data used in CTR prediction tasks typically includes multiple features, and the method of extracting key features can significantly affect the accuracy of CTR predictions. However, many click-through rate prediction models often overlook the importance of distinctive features. To address this, Effendi, Muhammad Junaid et al. [10] proposed a model for advertising click-through rate prediction that emphasizes feature importance.

In this paper, we explore a novel approach to examining various tree-based models for CTR predictions. Our goal is to identify the best algorithm and key features that uniquely determine email click-through rates using datasets from Kaggle.

II. MACHINE LEARNING MODELS FOR CTR PREDICTION

Machine learning models are divided into three major categories:

A. Classification

The Classification Module is a supervised machine learning module designed to categorize elements into groups. The objective of classification is to predict the categorical class labels that are not only discrete but also without any specific order. Common use cases include predicting whether a customer will default (Yes or No), deciding customer churn (whether a customer will leave or stay), and diagnosing diseases (positive or negative).

B. Regression

The Regression Module is a supervised machine learning module used to estimate the relationships between a dependent variable and one or more independent variables. The dependent variables are referred to as target or outcome variable whereas the independent variables are referred to as features, predictors, or covariates.

C. Clustering

The Clustering Module is an unsupervised machine learning module that groups a set of objects so that those within the same group (or cluster) are more like each other than to those in distinct groups.

Regression models are ideal for CTR prediction because they use a specific set of input parameters for training, enabling accurate predictions. The regression models are broadly categorized into two types: linear models and tree-based models.

III. TYPES OF MODELS

Linear regression and tree-based regression models are both used to predict a continuous outcome, but they differ significantly in their approach and application.

A. Linear Regression

The Linear regression model presumes a linear relationship between the variables, showing that changes in the features or independent variable(s) lead to proportional changes in the target or dependent variable.

The linear regression model is represented by an equation as shown:

(Y = a + bX) where (Y) is the target or dependent variable, (X) is the feature or independent variable. (a) is the intercept and predicts where the regression line will cross the y-axis, and (b) is the slope, which predicts the change in Y for every unit of change in X.

The coefficients (a) and (b) are straightforward to interpret, reflecting the impact of each predictor on the outcome. This approach is best suited for data where the relationship between variables is linear.

B. Tree-Based Regression

Tree-based regression models, such as decision trees, divide the data into subsets based on the values of input features. Each internal node represents a decision based on a feature, each branch stands for the outcome of that decision, and each leaf node represents a predicted value. The model generates a series of if-else rules to make predictions. For example, “if (X > 10), then (Y = 20); else if (X > 5), then (Y = 15); else (Y = 10)”.

The model does not presume a linear connection between the variables. This behavior enables it to identify and capture intricate, non-linear relationships. It is easy to interpret and visualize, particularly with smaller trees. Each path from the root to a leaf is a decision rule. This approach is more flexible and better suited for handling non-linear relationships compared to linear regression. Key Differences in Models are explained below:

- Flexibility: Tree-based models are more adaptable and can capture non-linear relationships, while linear regression is mostly limited to linear relationships.
- Interpretability: Both tree-based and linear models are interpretable, but in diverse ways. While linear regression offers a clear mathematical relationship between variables, the tree-based models present a set of decision rules.
- Robustness: Linear regression tends to be more robust against small variations in the data. In contrast, tree-based models can be more sensitive to changes in data and often require methods like pruning or the use of ensemble techniques to improve their stability.

IV. TREE-BASED MODELS FOR CTR PREDICTION

A. CatBoost Regressor

CatBoost is a powerful machine learning algorithm that has gained significant popularity due to its exceptional performance in handling categorical features and effectively modeling click-through data. It employs gradient boosting on decision trees to address various regression, classification, and ranking problems, making it particularly suitable for click-through rate (CTR) prediction tasks. Some

unique advantages of using CatBoost Regressor for CTR modeling are as given below:

- Automatic Encoding: CatBoost automatically encodes categorical features, reducing the need for manual preprocessing and enhancing model performance. It uses methods like one-hot encoding, target encoding, and feature combinations.
- Ordered Boosting: This novel technique reduces overfitting and variance by introducing a random permutation of the training data at each iteration. This approach allows the model to learn from both historical and future data, avoiding repeated splits for the same feature.
- Symmetric Decision Trees: CatBoost builds symmetric decision trees, ensuring that all leaves at the same level have the same depth. This improves interpretability and reduces computational complexity.
- Support for Various Loss Functions: CatBoost supports several loss functions suitable for CTR prediction, such as log loss, cross-entropy, and pairwise ranking. These functions capture the probabilistic nature of CTR problems and optimize the model accordingly.
- Analytical Tools: CatBoost offers tools for analyzing and evaluating the model, including feature importance, SHAP values, partial dependence plots, and model calibration. These tools help understand the model's predictions and identify the most influential features and interactions.

B. Random Forest Regressor

Random forest regression is a supervised learning algorithm that runs on labelled data. Known for its simplicity and high accuracy, it is widely used for regression problems, such as predicting continuous outcomes. The algorithm operates by constructing several decision trees, each one developed using randomly chosen subsets of the data. It then aggregates the outputs of these trees to make overall predictions for new data points. This approach allows it to handle larger datasets and capture more complex relationships than individual decision trees.

Random forest regression is applied to various business problems, including predicting future prices or costs, forecasting revenue, and comparing performance. As both a supervised learning algorithm and an ensemble method, it learns the mappings between inputs and outputs during training. The Ensemble algorithms integrate several machine learning models to produce more accurate predictions than any individual model could achieve alone. In the case of random forest, it combines multiple decision trees to form its final decision. This algorithm can be used for both regression tasks (predicting continuous outputs, like prices) and classification tasks (predicting categorical or discrete outputs).

C. Extra Trees Regressor

The Extra Trees Regressor (short for extremely randomized trees) is an ensemble supervised machine learning method, which uses decision trees. Like the random forests algorithm, Extra Trees creates multiple decision trees, but it samples randomly without replacement for each tree, resulting in unique datasets for each tree. Moreover, a certain number of features are chosen at random for each tree from the entire set of features. The unique characteristic feature of Extra Trees is its random selection of splitting values for features. Instead of using Gini or entropy to calculate a locally optimal split, the algorithm randomly chooses a split value, leading to more diversified and uncorrelated trees.

D. Light Gradient Boosting Machine (LightGBM)

LightGBM is an open-source, distributed, high-performance framework for gradient boosting (GBDT, GBRT, GBM, or MART). Developed by Microsoft's team led by Guolin Ke, it was introduced in a 2017 paper titled "LightGBM: A Highly Efficient Gradient Boosting Decision Tree." Designed for efficiency and scalability, LightGBM model became a popular choice for machine learning tasks involving large datasets. It is renowned for its speed and accuracy, achieved through innovative techniques such as histogram-based learning and leaf-wise tree growth. LightGBM is highly adaptable and can be utilized for a variety of machine learning applications, such as:

- Classification: Used for binary and multi-class classification problems, such as spam detection, image classification, and sentiment analysis.
- Regression: Ideal for regression tasks such as forecasting house prices, predicting trends in stock market, and customer lifetime value.
- Ranking: Applied in ranking tasks, such as search engine result ranking and recommendation systems.
- Anomaly Detection: Useful for detecting anomalies in datasets, which is beneficial for fraud detection and network security.

LightGBM's framework leverages gradient boosting, employing a sequence of decision trees to construct a strong predictive model. It prioritizes efficiency through a leaf-wise tree growth strategy and histogram-based algorithms. The leaf-wise method constructs trees by choosing the node that maximizes loss reduction, allowing for deeper trees and improved accuracy, though it may lead to overfitting with smaller datasets. The histogram-based approach converts continuous features into histograms, lowering computational complexity and memory usage, which is particularly beneficial for large-scale datasets. These techniques, combined with parallel and distributed computing capabilities, enable LightGBM to achieve fast training speeds, reduced memory consumption, and high scalability, making it ideal for various machine learning tasks.

In their experiments on online ads CTR prediction datasets, Xia, Zhen et al. [9] found that the Product & Cross supported Stacking Network with LightGBM (PCSNL) outperformed traditional CTR prediction models and deep learning models.

E. Gradient Boosting Regressor

Gradient Boosting is a robust boosting algorithm and a type of ensemble method that merges multiple weak learners into a strong learner. Each new model is trained to minimize the loss function, such as mean squared error or cross-entropy, of the earlier model using gradient descent. At each step, the algorithm computes the gradient of the loss function with respect to the current ensemble's predictions. It then trains a new weak model to reduce this gradient. The new model's predictions are added to the ensemble, and this process continues until a stopping criterion is reached.

V. RESULTS AND OBSERVATIONS

Interpreting the machine learning algorithm performance is the key component of the CTR prediction. Training data set is segregated into three buckets like train, test and validate. Training data is 60% of the data set, while validation and verification take 20% each.

A. Mean Absolute Error (MAE)

Mean Absolute Error quantifies the average magnitude of errors in a set of predictions, disregarding their direction. It represents the mean absolute difference between predicted and actual values. The Mean Absolute Error (MAE) is computed using the following formula

$$MAE = \frac{1}{n}\sum_{i=1}^{n} | Y_i - \hat{Y_i} | \quad (2)$$

B. Mean Squared Error (MSE)

Mean Squared Error (MSE) is a cost function that determines the average of the squared differences between predicted values and actual values. For example, in a regression model used to predict house prices, the Mean Squared Error (MSE) quantifies the average of the squared differences between the actual prices and the predicted prices. For instance, if the model predicts a house price to be $450,000, the squared error is the square of the difference between this prediction and the actual price. MSE averages these squared errors across all predictions. This metric emphasizes larger errors, which can be particularly important in scenarios like financial forecasting, where large errors can have significant consequences.

$$MSE = \frac{1}{n}\sum_{i=1}^{n} (Y_i - \hat{Y_i})^2 \quad (3)$$

C. Root Mean Squared Error (RMSE)

Root Mean Squared Error (RMSE) is a cost function that measures the square root of the mean squared error ensuring the error scale matches the target scale. In the context of predicting house prices, RMSE translates the error metric back to the price scale, making it easier to interpret the average error in terms of actual values. For example, if the RMSE is $20,000, it shows that the typical prediction error is around $20,000. Opting for RMSE over Mean Squared Error (MSE) can be beneficial, especially for practical applications and interpretability.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n} (Y_i - \hat{Y_i})^2} \quad (4)$$

D. R-Squared

R-Squared measures the proportion of variance in the dependent variable that can be forecast by the independent variables. It shows how closely the predictions match with the actual data. A high R-Squared value (close to 1) suggests that the model can accurately predict the actual values.

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(\hat{y}_i - y_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y}_i)^2} \quad (5)$$

E. Root Mean Squared Logarithmic Error

Root Mean Squared Logarithmic Error (RMSLE) is a measure to compute the square root of the average squared logarithmic differences between predicted and actual values. It is determined by taking the square root of the mean of these squared logarithmic errors.

$$RMSLE = \sqrt{1n\sum ni=1(log(pi+1) - log(ai+1))2} \quad (6)$$

Where: 'n' is the total number of observations in the (public/private) data set, pi is the prediction of target, and ai is the actual target for i.

F. Mean Absolute Percentage Error (MAPE)

Mean Absolute Percentage Error (MAPE) expresses the error as a percentage of the actual values, making it an easily understandable metric. For instance, if a house is worth $500,000 and the predicted value is $450,000, then it can be inferred that the error is 10%. This percentage-based approach makes MAPE highly interpretable, especially when explaining model performance to non-technical stakeholders.

$$MAPE = \frac{100}{n}\sum_{i=1}^{n} \left| \frac{Y_i - \hat{Y_i}}{Y_i} \right| \quad (7)$$

VI. CTR DATASETS

Both linear and tree-based models were trained on the datasets, using various key features that uniquely classify the CTR. See table 1.

Table 1: Top 4 algorithms for CTR Prediction

Tree Model	CatBoost Regressor	Random Forest	Extra Trees	LightGBM
MAE	0.029	0.032	0.031	0.032
MSE	0.004	0.004	0.004	0.004
RMSE	0.058	0.061	0.062	0.062
R2	0.5486	0.4755	0.4744	0.4723
RMSLE	0.0476	0.0511	0.0514	0.051
MAPE	2.0906	2.9901	2.4354	2.4208

The CatBoost Regressor emerged as the top-performing algorithm. The tabulated results above compare the top four algorithms. Other models included in the analysis are Gradient Boosting Regressor, Extreme Gradient Boosting, K Neighbours Regressor, Decision Tree Regressor, and AdaBoost Regressor, listed in descending order of performance.

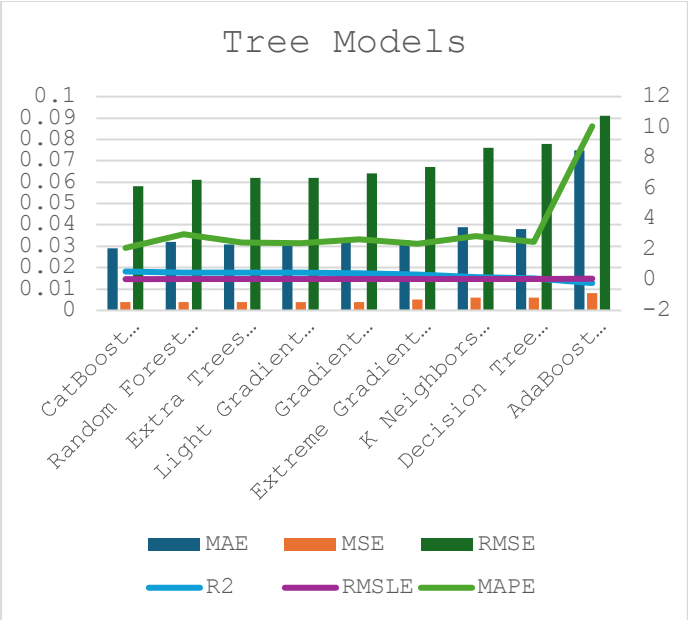

Figure 1: Algorithm Performance

The clustered column graph in Figure 1 showcases the MAE, MSE, and RMSE metrics on the column, while the secondary axis highlights the R2, RMSLE, and MAPE values. This visual representation effectively distinguishes each metric, clearly indicating the strongest and weakest algorithms for CTR predictions. In addition to identifying the best Machine Learning algorithm for CTR prediction, the study also aimed to determine the most influential features in the training process.

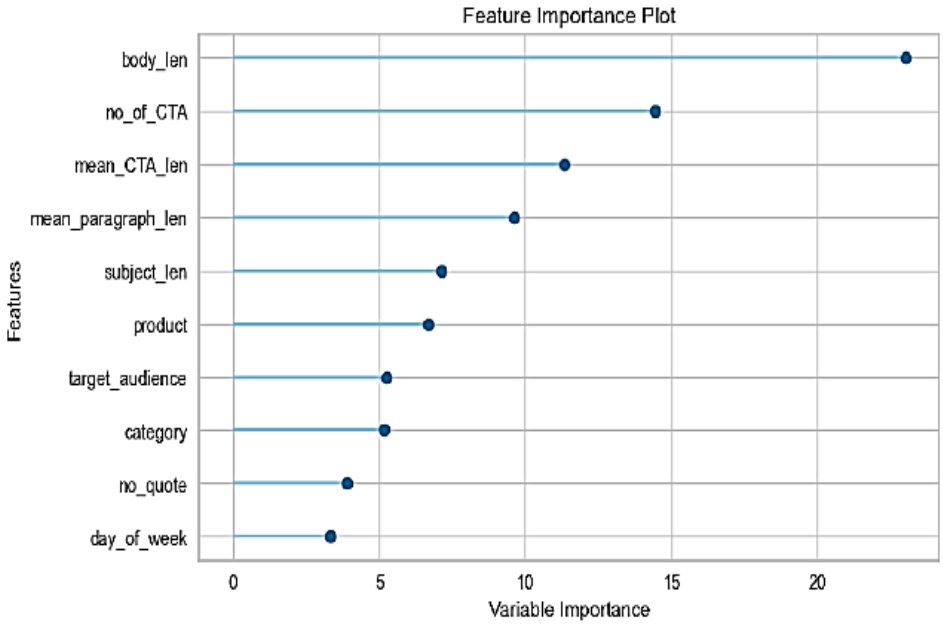

Figure 2: Feature Importance

In Figure 2 above, the results indicate that the features: body_len (number of characters in an email body), no_of_CTA (number of Call To Actions in an email), and mean_CTA_len (average number of characters in a CTA) are the most significant predictors of CTR.

VII. CONCLUSION

This study highlights the modeling and implementation of various tree-based machine learning algorithms for predicting CTR in email marketing. Features are identified based on various aspects of a marketing email and extracted to develop the model. Supervised machine learning algorithms, including CatBoost Regressor, Random Forest Regressor, Extra Trees Regressor, Light Gradient Boosting Machine, Gradient Boosting Regressor, Extreme Gradient Boosting, K Neighbors Regressor, Decision Tree Regressor, and AdaBoost Regressor, are used to construct the models. Experimental results show that CatBoost Regressor outperforms all other models in predicting CTR, with email body length emerging as a key feature. For future research, the model can be enhanced by incorporating more relevant features and larger datasets. Additionally, integrating the built models to create ensemble and fusion models could further improve prediction accuracy.

CONFLICTS OF INTEREST

The authors declare that they have no conflicts of interest.

REFERENCES

[1] I. H. Sarker, "Machine Learning: Algorithms, Real-World Applications and Research Directions," *SN Comput. Sci.*, vol. 2, no. 3, p. 160, Mar. 2021. Available from https://doi.org/10.1007/s42979-021-00592-x.

[2] L. Chen, "Research on advertising click-through rate prediction model based on Taobao big data," *Highlights in Science, Engineering and Technology*, vol. 56, pp. 179–187, 2023. Available from: https://doi.org/10.54097/hset.v56i.10102

[3] M. Richardson, E. Dominowska, and R. Ragno, "Predicting clicks: estimating the click-through rate for new ads," in *Proc. 16th Int. Conf. World Wide Web*, pp. 521–530, 2007. Available from: https://doi.org/10.1145/1242572.1242643

[4] O. Chapelle, E. Manavoglu, and R. Rosales, "Simple and scalable response prediction for display advertising," *ACM Trans. Intell. Syst. Technol. (TIST)*, vol. 5, no. 4, pp. 1–34, 2014. Available from: https://doi.org/10.1145/2532128

[5] X. He et al., "Practical lessons from predicting clicks on ads at Facebook," in *Proc. 8th Int. Workshop Data Mining Online Advertising*, pp. 1–9, 2014. Available from: https://doi.org/10.1145/2648584.2648589

[6] N. Yin, H. Li, and H. Su, "CLR: Coupled logistic regression model for CTR prediction," in *ACM Turing Celebration Conference*, 2017. Available from: https://doi.org/10.1145/3063955.3063976

[7] W. Zhang, Y. Han, B. Yi, et al., "Click-through rate prediction model integrating user interest and multi-head attention mechanism," *J. Big Data*, vol. 10, p. 11, 2023. Available from: https://doi.org/10.1186/s40537-023-00688-6

[8] T. Çakmak et al., "Accurate prediction of advertisement clicks based on impression and click-through rate using extreme gradient boosting," 2019. Available from: https://shorturl.at/jX2AN

[9] Z. Xia, S. Mao, J. Bai, X. Geng, and Y. Liu, "A novel integrated network with LightGBM for click-through rate prediction," 2021. Available from: https://shorturl.at/cnVf2

[10] M. J. Effendi and S. A. Ali, "Click-through rate prediction for contextual advertisement using linear regression," *ArXiv*, vol. abs/1701.08744, 2017. Available from: https://shorturl.at/kc1Tp

[11] S. Kathuria, A. Kumar, S. Gupta, and A. Mishra, "Predicting the click-through rate using machine learning methodologies," *JETIR*, vol. 6, no. 3, 2019. Available from: https://www.jetir.org/view?paper=JETIR1903971

[12] H. B. McMahan et al., "Ad click prediction: A view from the trenches," in *Proc. KDD'13*, 2013. Available from: https://doi.org/10.1145/2487575.2488200

[13] L. Shi and B. Li, "Predict the click-through rate and average cost per click for keywords using machine learning methodologies," in *Proc. Int. Conf. Ind. Eng. Oper. Manag.*, 2016. Available from: https://shorturl.at/yN180

[14] P. A. Clemenshia and M. S. Vijaya, "Click-through rate prediction for display advertisement," *Int. J. Comput. Appl.*, vol. 136, no. 1, Feb. 2016. Available from: https://shorturl.at/Mew99

[15] L. Breiman, J. H. Friedman, C. J. Stone, and R. A. Olshen, *Classification and Regression Trees*, CRC Press, 1984. Available from: https://doi.org/10.1201/9781315139470

[16] A. Kolan, M. Dasika, K. Sreevani, and H. Jayasree, "Click-through rate prediction using decision tree," 2020. Available from: https://shorturl.at/7CcyN

[17] Z. Wang, Q. Yu, C. Shen, and W. Hu, "Feature selection in click-through rate prediction based on gradient boosting," 2016. Available from doi: https://shorturl.at/JiU7I.

[18] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discov. Data Min.*, San Francisco, CA, USA, pp. 785–794, 2016. Available from: https://doi.org/10.1145/2939672.2939785

[19] C. Yang, "A click-through rate prediction algorithm based on real-time user feedback," *J. Comput. Appl.*, vol. 37, no. 10, pp. 2866–2870, 2017. Available from: https://shorturl.at/mFucB

[20] A.-Y. Zhou, M.-Q. Zhou, and X.-Q. Gong, "Computational advertising: A data-centric integrated application for the web," *J. Comput. Sci.*, vol. 34, no. 10, pp. 1805–1819, 2011. Available from:

[21] Y. Juan, Y. Zhuang, W. S. Chin, et al., "Field-aware factorization machines for CTR prediction," in *Proc. 10th ACM Conf. Recommender Syst*, pp. 43–50, 2016. Available from: https://doi.org/10.1145/2959100.2959134

[22] S. Rendle, "Factorization machines," in *Proc. 2010 IEEE Int. Conf. Data Min.*, IEEE, pp. 995–1000, 2010. Available from: https://doi.org/10.1109/ICDM.2010.127

[23] J. Pan et al., "Field-weighted factorization machines for click-through rate prediction in display advertising," in *Proc. 2018 World Wide Web Conf.*, pp. 1349–1357, 2018. Available from: https://doi.org/10.1145/3178876.3186040

[24] H. Guo, R. Tang, Y. Ye, et al., "DeepFM: A factorization-machine based neural network for CTR prediction," *arXiv preprint arXiv:1703.04247*, 2017. Available from: https://shorturl.at/XZX1C

[25] X. Wei, C. Yu, C. Tian, et al., "Internet CTR prediction model on big data platform," *Proc. Comput. Eng. Des.*, vol. 38, no. 9, pp. 2504–2508, 2017. Available from: https://doi.org/10.16208/j.issn1000-7024.2017.09.038

[26] X. He and X. Guo, "Research on advertising click-through rate prediction model based on feature optimization," *J. East China Norm. Univ. (Nat. Sci. Ed.)*, no. 4, pp. 147–155, 2020. Available from: https://shorturl.at/wzxjK

[27] S. Li, Z. Cui, and Y. Pei, "A dual adaptive interaction click-through rate prediction based on attention logarithmic interaction network," *Entropy*, vol. 24, no. 12, p. 1831, 2022. Available from: https://doi.org/10.3390/e24121831

[28] F. Zhou, "A model for predicting the click-through rate of advertisements based on feature importance," *Proc. Comput. Knowl. Technol.*, vol. 16, no. 36, pp. 12–14, 2020. Available from: https://shorturl.at/QCmgf.

[29] J. Zhang et al., "Multi-scale and multi-channel neural network for click-through rate prediction," *Neurocomputing*, vol. 480, pp. 157–168, 2022. Available from: https://doi.org/10.1016/j.neucom.2022.01.035

[30] X. Y. Jiang et al., "A prediction model for advertising click-through rate with dynamic feature importance extraction," *Proc. Small Microcomput. Syst.*, vol. 43, no. 5, pp. 976–984, 2022. Available from: https://shorturl.at/ddq1x.

[31] T. Chen et al., *XGBoost: Extreme Gradient Boosting*, R package version 0.4-2, 2015. Available from:

[32] A. O. Sykes, "An introduction to regression analysis," 1993. Available from: https://shorturl.at/T5nSv

[33] B. Efron, "Estimation and accuracy after model selection," *J. Am. Stat. Assoc.*, vol. 109, no. 507, pp. 991–1007, 2014. Available from: https://shorturl.at/03p1y

[34] B. D. Klein and D. F. Rossin, "Data quality in linear regression models," *Informing Sci. J.*, vol. 2, pp. 33–43, 1999. Available from: https://shorturl.at/gxEyv

[35] P. Usharani, "Fast algorithms for mining association rules in data mining," *Int. J. Sci. Technol. Res.*, vol. 2, pp. 13–24, 2013. Available from: https://shorturl.at/tcyzO