ADBA Chaos and Fractals
Cutting-Edge Scientific Solutions

e-ISSN: 3023-8951
RESEARCH ARTICLE
Vol.3 / No.1 / 2026 / pp.38-46
https://doi.org/10.69882/adba.chf.2026016

Ad-Click Prediction Enhanced by Nonlinear Dynamics-Inspired Feature Extraction and Ensemble Optimization

Ceyda Çağ[α,1], **Neslihan Akbulut**[α,2] **and Yusuf Çankırlı**[α,3]
[α]Hitit University, Faculty of Engineering and Natural Sciences, Department of Computer Engineering, 19030, Corum, Türkiye.

ABSTRACT The primary objective of this study is to enhance the predictive performance of machine learning models used for estimating Click-Through Rate (CTR), a key metric in digital advertising analytics. Beginning with a baseline Logistic Regression (LR) model applied to the "Click-Through Rate Prediction" dataset from Kaggle, the study systematically incorporates multiple optimization layers to improve forecasting accuracy. Inspired by nonlinear dynamics concepts, new feature representations were derived from temporal patterns and textual fields using TF-IDF and Word2Vec-based embeddings. Hyperparameter optimization techniques were then applied to refine model behavior, followed by the construction of ensemble architectures combining LR, XGBoost, Random Forest (RF), and Support Vector Machine (SVM) classifiers. Experimental results show that the optimized ensemble achieved the highest F1-score of 0.8694, yielding an improvement of approximately 12.7% over the baseline model. Overall, the study provides a comprehensive examination of feature extraction strategies, model optimization procedures, and ensemble fusion techniques, demonstrating the clear advantage of hybrid approaches in complex CTR prediction tasks.

KEYWORDS
Feature extraction
Ensemble learning
Machine learning
Hyperparameter optimization
Classification

INTRODUCTION

The digital environment is a virtual environment where people share their feelings and thoughts with each other. When this virtual environment was first established, people focused only on communicating with each other by sharing photos, videos and texts. As time passed, people started to use this medium not only as a source of communication, but also in many areas such as announcing their own brands, shopping, education, news and so on. Digital advertising emerged when people started shopping through this medium. Digital advertising is when people advertise using the digital environment to announce their brands, promote their products, and make sales. As time passes, people shop digitally, which has allowed digital advertising to increase and gain importance worldwide.

Digital advertising, which has spread globally, has become the focus of attention of all brands, sellers, institutions and users. Total advertising expenditures have increased worldwide according to the latest research. The global market size for digital advertising is expected to approach $700 billion in 2024 and exceed $830 billion by 2026. In 2023, a user spent approximately 6.5 hours a day interacting with ad content (Gangopadhyay *et al.* 2025) . Users are essential building blocks for engagement in the advertising industry. Digital advertising can now be seen while watching a video, playing a game, and using an app. In this way, it has gained direction in the digital advertising sector and it has become easier for brands to reach their target users.

Although many advantages emerge with the growth of digital advertising, some problems may also arise. In digital advertising, users' click-through rates have an important place in terms of evaluating many situations such as the user's requests, likes, complaints, and trends of the time. In digital advertising, predicting which ad will be clicked more has an important place. However, the click-through rate may not always reflect the truth. The placement of ads on the page is very important for this. Since users see the first ad they see more, they are more likely to click on that ad, which can make it difficult to find the real click-through rate of the ads. In addition, incorrect clicks made by users also make the click-through rate misleading. Advertising trends that change over time cause the click-through rate to change rapidly. All these factors prevent properly evaluating ad click-through rates, acting according to the user's wishes, and thus earning more.

Manuscript received: 18 December 2025,
Revised: 23 January 2026,
Accepted: 23 January 2026.

[1]234210020@ogrenci.hitit.edu.tr
[2]234210008@ogrenci.hitit.edu.tr
[3]234210024@ogrenci.hitit.edu.tr

The concept of CTR in digital advertising is a method used to evaluate the success of digital ads (AgencyAnalytics 2025). Click-through rate is an important problem in the field of digital marketing and online advertising. In our study, it is aimed to develop various hyperparameter optimizations, models, feature extraction, ensemble models and to help the click-through rate problem by combining these studies. With the increase in digital advertising, our study contributes to the development of many factors such as reaching the user more easily, interacting, allowing brands to introduce themselves, and using social platforms more effectively. Digital advertising is guided by user desires, and finding a click-through rate is important for advertisers, marketers, and campaign managers.

In this direction, the layout of the ads, which page they will appear on, which ad attracts more attention are learned and other services are offered to the user according to these details. User engagement with an effective digital ad is not limited to the first click or view; instead, it becomes more comprehensive. For example, a user who clicks on an ad for a particular book is taken not only to the product they are looking for, but also to other book recommendations personalized to their interests; This process aims to increase user engagement on the platform and encourage them to make additional purchases.In this way, in digital advertising, whatever type of content a user is interested in, similar ads appear more often, so that the ads can reach the right user. The overall progression of the study development process is shown in Figure 1.

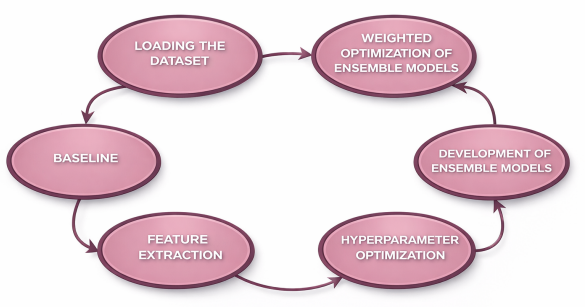

Figure 1 Project Development Flowchart

RELATED STUDIES

Yang and Zhai (2022) stated that the main purpose of CTR prediction in online advertising is to predict the likelihood that a user will click on an ad they see. They also mentioned that it was done by creating a model that numerically calculates the probability of a user clicking on the ad. They stated that the interest in this subject has increased in the last decade and that while statistical methods such as the LR model were used at first, these older models are now being replaced by more advanced approaches. The main advantages of the LR model are that it efficiently captures linear correlations between the feature and the label, offering an interpretable probability. On the other hand, they showed that the model was insufficient to model the complex and nonlinear interactions between them because it assumed that the features were independent.

Lou (2024) has conducted a comparative analysis of LR, RF, and XGBoost for CTR prediction in digital advertising.The study discussed machine learning concepts such as model selection, data processing, and feature engineering.He stated that there are difficulties that reduce the performance of the model, especially in the new data, and to address these difficulties, the performances of the LR, RF and XGBoost models were compared after preprocessing was applied to a complex dataset. Among the three models considered, XGBoost has been the most successful model, achieving the highest performance with an accuracy rate of 94.10% and an AUC score of 0.98 Compared to XGBoost, RF achieved 93.52% accuracy with an AUC score of 0.97, while LR achieved 93.23% accuracy and an AUC score of 0.96. As a result, it was revealed that XGBoost, which was effective in CTR prediction in the three models considered, is the most promising model with its efficient processing of complex data structures and superior prediction ability.

Zang (2019) focuses on predicting the click-through rate of loan ads. He performed data cleaning and transformation operations on user data obtained from the Finup platform. The first modeling to show the effects of the imbalanced structure of the dataset was SVM, Naive Bayes, using decision tree and neural network models and performing predictive analysis. While the accuracy rates of the models were over 97%, the sensitivity rates were found to be 0%. The reason for the sensitivity of 0% was interpreted as excessive imbalance, and it was concluded that undersampling was required to increase it. He stated that the SVM model performed better than other models. Experiments were conducted on 5 different datasets to compare the performance of the models. As a result of the experiments, the lowest 0.8936, the highest 0.9324 accuracy and the lowest 0.7295 and the highest 0.7839 precision were given. The study also used the results of SVM, the most successful model, to create a clear profile of users with high click potential and thus provide a directly applicable targeting strategy for loan ads.

AlAli *et al.* (2021) conducted a study on the prediction of click-through rate effectiveness in mobile ads using XGBoost and developed a machine learning-based CTR model in their study. They used the "Click-Through Rate Prediction Competition Dataset" dataset from the Kaggle platform and mentioned that the dataset they used was unbalanced. They applied the under-sampling technique to solve the imbalance problem. In their study, they used 4 different models, namely K-Nearest Neighbor (KNN), LR, RF and XGBoost, and after measuring the baseline performance of these models, they applied hyperparameter optimization methods on the models to improve the ROC-AUC score. In the optimized results of the models, it was revealed that the XGBoost algorithm performed better than the other three algorithms, RF 0.7544, LR 0.6428 and KNN 0.7172, with a ROC-AUC score of 0.7640. As a result, they added that important results were obtained depending on the superiority of XGBoost over the other three models and the evaluation metrics of the models.

Bratus and Bidyuk (2023) In their study, applied various feature engineering techniques to obtain low-dimensional representations of sparse and high-dimensional input features. They then trained the Naive Bayes, LR, SVM, Random Forest, and XGBoost models to compare their performance on the preprocessed data and select the best one. They stated that the dataset they used was KDD Cup 2012. The dataset was divided into test and training set, and they showed that there was no strong correlation between the variables and that there was an imbalance in the target variable. The evaluation metrics are ROC-AUC and LogLoss.

When the results of the models were taken, XGBoost achieved an AUC-ROC score of 0.6909 and a LogLoss score of 0.1783. Random Forest achieved an AUC-ROC score of 0.6654 and a LogLoss score of 0.1821. As a result, they stated that XGBoost stands out with its high performance by offering an approach supported by comprehensive data preparation processes and comparing different models for CTR prediction.

Table 1 Comparative Literature Analysis

Authors	Study	Methods	Results
Yang, Y., Zhai, P.	Click-through rate prediction in online advertising: A literature review	LR	CTR forecasting, advantages and limitations of the LR model
Lou, J.	Comparative Analysis of Logistic Regression, Random Forest, and XGBoost for CTR Prediction in Digital Advertising	LR, Random Forest, XGBoost	XGBoost: 94.10% accuracy, highest 0.98 AUC; RF: 93.52%, 0.97 AUC; LR: 93.23%, 0.96 AUC
Zhang X.	Click Prediction for P2P Loan Ads Based on Support Vector Machine	Naive Bayes, Decision Tree, Neural Network, SVM	The SVM model achieved 89.36-93.24% accuracy and 72.57-78.39% sensitivity using RBF core
M. AlAli et al.	Click-Through Rate Effectiveness Prediction on Mobile Ads Using Extreme Gradient Boosting	KNN, LR, RF, XGBoost	XGBoost: 0.7640 ROC-AUC (highest); RF: 0.7544; KNN: 0.7172; LR: 0.6428.
O. S. Bratus, P. I. Bidyuk	Towards Click-Through Rate Prediction in Online Advertising	Naive Bayes, LR, Random Forest, SVM	XGBoost: AUC-ROC 0.6909, Logloss 0.1783 (best result)
J. R. Guillen	Click Through Rate Prediction Leveraging Machine Learning Techniques for Mobile Digital Advertisement	LR, RF, XGBoost, CatBoost, FFNN	CatBoost: Lowest logloss 0.5836, highest F1-score 0.7093

Rojas Guillen (2024) emphasizes the importance of estimating CTR to optimize the effectiveness of mobile advertising campaigns. It has demonstrated the potential of machine learning models in predicting CTR for mobile digital ads. The models it uses are LR, RF, XGBoost, CatBoost and Feedforward Neural Network (FFNN). The dataset consists of 10 days of click data corresponding to a high-dimensional categorical mobile ad impressions. Due to its low click rate, it has an unstable data structure and is balanced by the downsampling technique. In terms of model performance, it has been the most successful model with the lowest LogLoss score of 0.5836 and the highest F1-score of 0.7093. Overall, the study reveals the superior performance of CatBoost, especially in advertising data with a high-dimensional and categorically weighted data structure. The information on the studies, authors, methods used and the results obtained are shown in Table 1 in which the literature analyses carried out within the scope of the study are explained comparatively.

MATERIALS AND METHODS

In this study, the dataset named "Click-Through Rate Prediction" shared by user swekerr on the Kaggle platform was used swekerr (2024). In addition, the importance of click-through rate in digital advertising and different methods were examined in order to find more realistic value in the dataset. This dataset consists of 10000 data samples and 10 features. The features in the dataset and their definitions are shown in Table 2.

These features give us data such as users' information, which device they use, time information. The dataset contains both categorical and numeric features. The Ad Topic Line, City, Gender, Country features are categorical, the Daily Time Spent on Site, Age, Area Income, Daily Internet Usage and Timestamp features are numeric, the target feature is the clicked or non-clicked on Ad.

There are 5 data samples from the dataset are shown in Table 3. It is possible to see the subheadings in the features. In this table, all values are known and there are no missing values. Not all features can have similar or identical features, they are not unique. Daily Time Spent on Site, Age, Area Income, Daily Internet Usage, Ad Topic Line, City, Country, Timestamp features can take similar and different values, while Gender and Clicked on Ad can take stereotyped values. Gender takes male and female values, while Clicked on Ad takes 1 (clicked)- 0 (not clicked) values. As a result of the data quality checks, incompatible matches and content inconsistencies were detected between the City and Country attributes. In the analyzes, it was observed that there were more than one different country value for the same city value. For example, multiple locale matches have been identified for 417 cities. These inconsistencies have been omitted from the City feature dataset, as they can impact the reliability of the model.

Table 2 Features of The Original Dataset

Feature	Description	Type
Daily Time Spent On Site	Average time spent on the website (minutes)	Numeric
Age	Age of the user	Numeric
Area Income	The average income level of the user's region	Numeric
Daily Internet Usage	User's daily internet usage time (minutes)	Numeric
Ad Topic Line	Headline or subject text of the ad	Categorical
City	The city where the user lives	Categorical
Gender	User's gender	Categorical
Country	Country where the user lives	Categorical
Timestamp	Date and time data was collected	Datetime
Clicked on Ad	Click indicator (0: no click, 1: click)	Target

Histogram graphs according to the ad click or non-click rate of the features in the dataset are given in Figure 2. In this study, graphs were used to see the click rate. As can be seen in the graphs, there are histogram graphs of numerical features. According to the analysis, the 'Daily Time Spent on Site' (a) graph shows a positive relationship between the time spent on the site and the click-through rate, especially in the 80-90 minute range. The 'Age' (b) graph shows that users aged 40-60 tend to click on ads more.

Table 3 Samples of The Original Dataset

Daily Time Spent on Site	Age	Area Income	Daily Internet Usage	Ad Topic Line	City	Gender	Country	Timestamp	Clicked on Ad
62.26	32.0	69481.85	172.83	Decentralized real-time circuit	Lisafort	Male	Svalbard & Jan Mayen Island	2016-06-09 21:43:05	0
65.77	34.0	59785.94	168.34	Cloned explicit middleware	Kingshire	Male	Namibia	2016-02-27 08:52:50	0
79.6	23.0	62784.85	146.8	Team-oriented executive core	Patriciahaven	Female	Czech Republic	2016-01-14 14:00:09	1
40.47	27.0	14548.06	190.17	User-friendly upward-trending intranet	New Lucasburgh	Female	Poland	2016-02-10 19:20:51	1
50.63	31.0	61067.58	236.87	Optional multi-state hardware	Austinland	Male	Hong Kong	2016-07-04 23:17:47	0

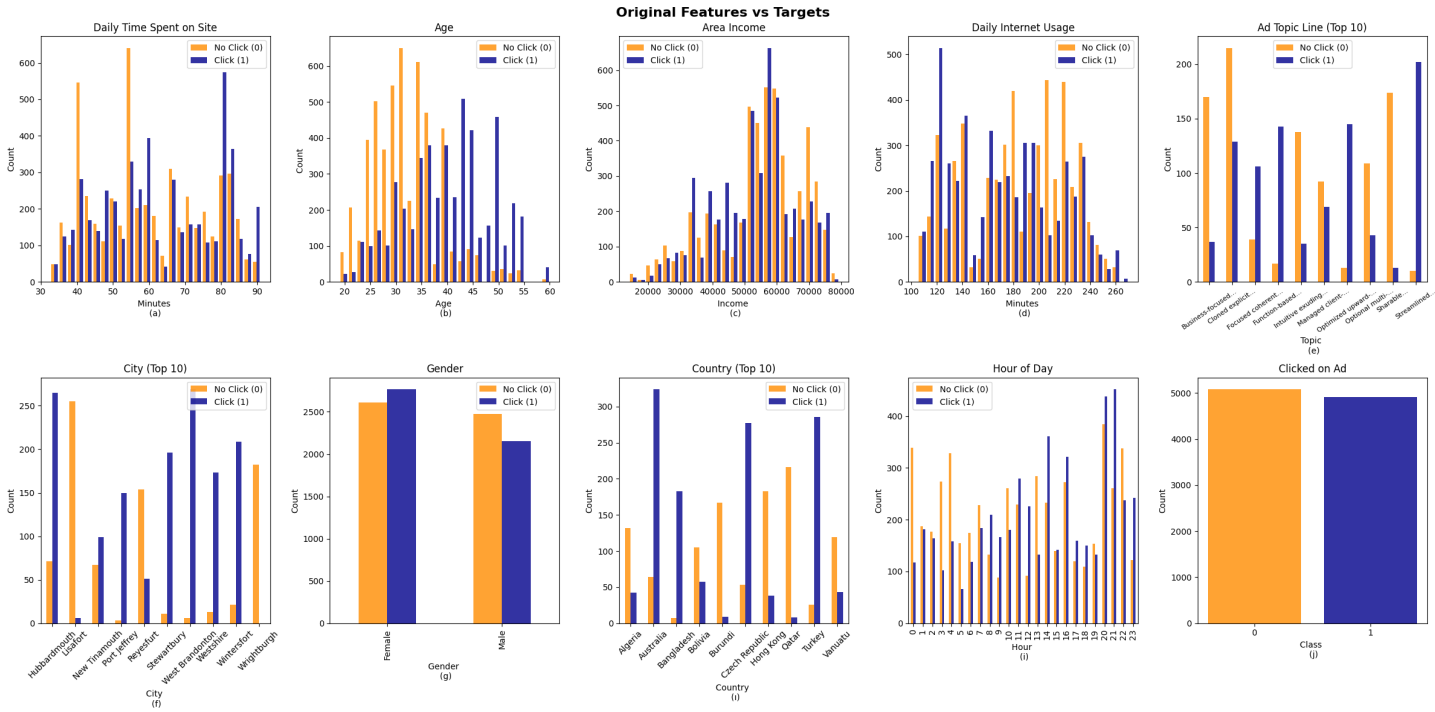

Figure 2 Feature Distribution Histograms for Click and Non-Click Samples

features such as 'Area Income' (c) and 'Daily Internet Usage' (d) do not present a significant separation, and the density in these graphs suggests that the overall user profile in the dataset is clustered in these ranges rather than a specific behavior. The 'Ad Topic Line (Top 10)' (e) chart, which emphasizes the importance of ad content, proves that some headlines directly increase click-through rates. In geographic distribution, the 'Country (Top 10)' (h) chart shows that countries like Autralia, Burundi, and Turkey have higher click-through rates, while the 'Gender' (g) chart shows that women are more likely to click. The 'City (Top 10)' (f) graph was used for analysis prior to data quality control; During model training, the City feature was removed due to incompatibility, while the 'Gender' (g) chart reveals that women are more likely to click. Additionally, the 'Hour of Day' (i) chart indicates that the hours of day also play a significant role, with click-through rates increasing in the morning hours. Finally, the 'Clicked on Ad' (j) graph, which shows the overall distribution of the target variable, shows that the number of clicked and non-clicking users in the dataset was

almost equal.

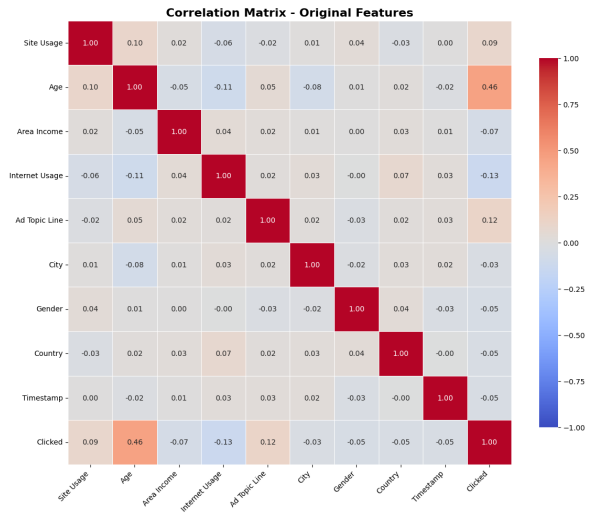

Figure 3 Correlation Matrix Across Features

There is an almost equal distribution in click-through and non-click-through rates in the dataset. The correlation matrix in Figure 3 is used to see the interactions of the features in this study. Weak relationships are mostly seen in the correlation matrix. However, the highest score is seen with 0.46 among the age and click features. For this reason, it can be observed that the ad click-through rate increases as the age value increases.

All of the machine learning models and data processing processes developed in the study were carried out by using the open source libraries of the Python programming language. DataFrame structures provided by the Pandas library were used to bring the data into an easy-to-process format with the loading and structuring stages (The pandas development team 2020; McKinney 2010). Numpy library was preferred for numerical operations and mathematical transformations of features that require high performance (Harris *et al.* 2020). It has enabled us to execute computationally intensive preprocessing steps such as StandardScaler in a more performance-friendly manner. In the data visualization phase, two libraries were used. Matplotlib was used to create the basic and customized images we wanted (Hunter 2007). The seaborn library was preferred in order to present our analyzes and make our findings more understandable by producing statistically rich and aesthetically advanced graphics with less code (Waskom 2021). The machine learning workflow was created by the Scikit-learn library. (Pedregosa *et al.* 2011).

The basic steps of the dataset, such as separating it into training and testing, setting up and training the models, and evaluating their performance with metrics such as F1 and ROC-AUC, were carried out through this library. It has significantly accelerated the experimental process by facilitating the experimentation and comparison of different algorithms. In the Feature Engineering phase, two complementary libraries were used. New features were derived from both temporal and textual data in order to increase the performance of the model. The semantic richness of the text data was also utilized. The Natural Language Toolkit (NLTK) was used as an infrastructure to break down texts into their basic components, such as words (Bird *et al.* 2009). The Gensim library was also preferred in order to convert the semantic meaning of these components into a numerical format that the model could understand (Řehůřek and Sojka 2010). Our dataset is strengthened with new features that reflect temporal and textual meaning thanks to these approaches.

In the study, a hybrid encoding strategy was applied to convert categorical features into numerical form. This strategy includes different encoding methods according to the cardinality of categorical variables, that is, having many features. One-Hot Encoding is applied for low-cardinality features. This encoding method preserves the information by converting each category into a column in binary (0/1) and ensures that each category is evaluated equally. In this step, the gender feature includes two new features (Male/Female), the season feature contains four new features (Spring, Summer, Autumn, Winter), and the time_of_day feature contains four features (Morning, Afternoon, Evening, Night). The features resulting from the feature extraction process are shown in Table 4.

Table 4 Extracted Feature Set

Feature	Description	Type
hour	Time when data was recorded (between 0–23)	Numeric
day_of_week	Day of the week (0 = Monday ... 6 = Sunday)	Categorical
is_weekend	Whether the data belongs to the weekend (0 = weekdays, 1 = weekends)	Binary
quarter	Quarter of the year (1 = January–March, ... 4 = October–December)	Categorical
month	Month to which the data belongs (1–12)	Numeric
week_of_year	What week of the year it is in (1–52/53)	Numeric
season	Seasonal information (Winter, Spring, Summer)	Categorical
time_of_day	Part of the day (Morning, Afternoon, Evening, Night)	Categorical
textual features	Numeric attributes derived from the ad headline (Ad Topic Line) using TF-IDF, Word2Vec, and text statistics methods	Numeric

Frequency Encoding method was used for categorical variables with high cardinality. This approach represents the incidence of each category value in the dataset as a numerical feature and controls the number of features. If One-Hot Encoding is applied in this step, it adds a large number of new columns, increasing the complexity of the model and leading to overfitting. Frequency Coding represents category values in a single numerical column, reducing model complexity and training time. The advantage of this method is that, unlike Ordinal Encoding, it does not create an artificial numerical order between categories; thus, reducing the risk of bias in model results. Following the data quality controls,

the City feature was removed from the dataset, and Frequency Coding was applied for the Country feature. As a result of the hybrid approach model, 2 new features were created for gender, 4 for season, 4 for time_of_day, and country feature 1 numeric feature represented by frequency value. This approach minimizes information loss while keeping feature size in check and does not create artificial sorting between categories. Three different approaches were used to extract meaningful information from many text data, such as advertising content. TF-IDF (Term Frequency-Inverse Document Frequency) vectorization converts text data into numerical data. In this study, 100 new features were obtained using TfidfVectorizer. In order to obtain these features, the 100 most important and most frequently seen words were selected with the parameter max_features=100. The parameter ngram_range = (1,2) is set to include both single word (unigram) and binary word groups (bigram). This approach can capture the meaning of word groups and create more detailed features. The parameters min_df=2 and max_df=0.85 filter out both very rare and very common words,creating more meaningful features.

Word2Vec embedding and TF-IDF-weighted concatenation create 100-dimensional vectors for each word name to capture the semantic relationships of words. 100-dimensional embeddings are produced with the parameter of model vector_size=100. The calculation was made by representing each sentence as a vector and weighting the weighted average of the Word2Vec vectors of the Word2Vec vectors with the TF-IDF scores. This approach can create more detailed features by combining both the semantic meaning and importance of words in the dataset TF-IDF models. With this method, 100 more features were obtained.

Text statistics features In order to capture the structural features of the text, four statistical features were calculated: word count, character count, unique word count, average word length. These features improve model performance by capturing the structural features of the data in the ad headline. With the combination of these 3 methods, a total of 204 new features were extracted from the ad headline data, including TF-IDF vectorization 100 features, Word2Vec embedding 100 features, and text statistics 4 features. These features were combined with numerical and categorical features to create a 225-dimensional feature set. This hybrid approach significantly enhances the model's performance by capturing different aspects of text content, such as the ad headline. As a result of all these methods, our F1 and ROC-AUC values change significantly when we run LR on the dataset again.

USED MODELS AND HYPERPARAMETER OPTIMIZATIONS

In order to improve the performance of machine learning models, various optimization techniques can make significant contributions to the process of finding the most appropriate hyperparameter combinations. In our study, Random Search, Grid Search, Optuna, Genetic Algorithm (GA) and Artificial Bee Colony (ABC) Algorithm were used for hyperparameter optimization. Random Search is a search method in which a predetermined number of combinations of hyperparameters are randomly sampled from a user-defined search space. Grid Search is a comprehensive search method in which all combinations on a subset of the hyperparameter space (a grid) manually specified by the user are tried (Bergstra and Bengio 2012).

Optuna is automating and making the hyperparameter search process, which is one of the laborious tasks in machine learning projects, efficient (Akiba *et al.* 2019). GA is a search and optimization technique inspired by the mechanisms of natural evolution and genetics. Genetic algorithms are used to find robust and efficient solutions for problems with complex and large search spaces (Goldberg 1989). The ABC Algorithm is a search and optimization algorithm based on swarm intelligence. The main purpose of the algorithm is to find the optimal solution in complex optimization problems (Karaboga and Basturk 2007).

LR analysis is an analysis that allows us to construct a regression model without requiring assumptions such as normality, continuity, covariance and multivariate normality (Şenel and Alatlı 2014). However, the capacity to model nonlinear relationships is limited due to the linear decision boundary. In order to overcome this limitation Random Forest is a supervised machine learning method consisting of an ensemble technique and a large number of decision trees used for categorization. Similarly SVM are supervised maximum-margin models in machine learning with correlated learning algorithms that analyze data for classification and regression analysis (Cortes and Vapnik 1995). In addition, XGBoost is a different approach to developing a decision tree than the classical gradient boosting decision tree methodology (Shams *et al.* 2024).

Within the scope of the Nonlinear Dynamics-Inspired approach the hyperparameter optimization methods we used in our study were applied in the LR model. Our goal is to increase our F1 and ROC-AUC score by taking optimized hyperparameters. As a result of the trials, Genetic Algorithm achieved the highest performance among all the methods examined and achieved an F1 score of 0.7826; ABC Algorithm 0.7820, Random Search 0.7818, Grid Search 0.7806 and Optuna 0.7686 F1 scores. These results show that the evolutionary search strategy of Genetic Algorithm optimization yields the most effective results in this problem and dataset, thanks to the more effective scanning of nonlinear performance surfaces. As a result, the Logistic Regression model was retrained using the optimal parameters found by the Genetic Algorithm (C: 17.13, penalty: 'l1', solver: 'liblinear', max_iter: 5658, class_weight: 0: 1, 1: 2) and an F1 score of 0.7826 was obtained. Genetic Algorithm has been shown to be an effective and reliable method to find the best combination in defined parameter space involving nonlinear structures.

In line with the Nonlinear Dynamics-Inspired approach, after the hyperparameter optimizations were implemented, the model training phase was started. In order to make a strong classification, a detailed model development and optimization process was followed. First, the classification algorithms XGBoost, Random Forest and SVM, which are discussed in the study, were used. To enhance the performance of these models on the training data, extensive optimization was conducted using Random Search to explore nonlinear parameter interactions more effectively. The optimization process was carried out together with the Cross Validation method in order to ensure the generalization ability of the model. In addition to this optimization, threshold optimization was also used to optimize the binary classification threshold of the results of the classification models. The main purpose of this optimization is to enhance the F1 score, a critical score especially in imbalanced datasets where nonlinear decision boundaries need to be determined effectively. When the optimization results were examined, it was observed that these optimizations significantly increased the F1 score of the models. Among the trained models, it was observed that the XGBoost model showed the highest performance by being more effective on LR and was a more effective classification model.

ENSEMBLE MODELS AND PERFORMANCE ENHANCEMENT

In order to take individual model performances to the next level, ensemble methods powered by model diversity were applied. The strategy aims to produce more robust and highly accurate results by compensating for the weaknesses of different models that exhibit nonlinear decision structures with each other. In the process, the combinations of each base model with the LR model were examined and which model would be more effective was examined. The LR model was used as a key component in the task of combining and weighting predictions. In this context, a total of seven different LR-based ensemble combinations were created using the basic models SVM, RF and XGBoost. These ensemble structures, detailed in Figure 4, were examined with two approaches; Uniform Weighting, equal contribution was given to all models in the combination. In this approach, it is aimed to generalize by averaging the model diversity. In Optimized Weighting (Best Weighting), the Random Search technique was used to automatically optimize each model's contribution to the prediction on nonlinear (nonlinear) performance surfaces to increase the F1 score. This method allowed to learn which model should have more say through data.

As a result of the investigations, it was seen that the optimized weighting approach was more effective and produced higher F1 scores compared to equal weighting. This approach has proven that it can be enhanced not only by incorporating powerful models but also by evaluating and optimizing the models' contributions to performance within a nonlinear performance structure. In particular, the ensemble structure of the models combined with the optimized weights approach achieved the highest F1 score in the study and was seen as the most effective method among the methods used. In this ensemble model, it was also observed that the threshold optimization applied had a positive effect on performance. The model development process has shown that instead of looking for a single model, powerful models are comprehensively optimized and combined with ensemble methods to provide the highest classification performance. The highest F1 score was achieved by the weighted ensemble method, which included LR, XGBoost and SVM models. The weighting of each model in the final estimate was optimized to maximize the F1 score by considering complex and non-linear contribution relationships, and this strategy outperformed all individual models. Thus, it was seen that different models compensated for their weaknesses with each other and created a successful solution.

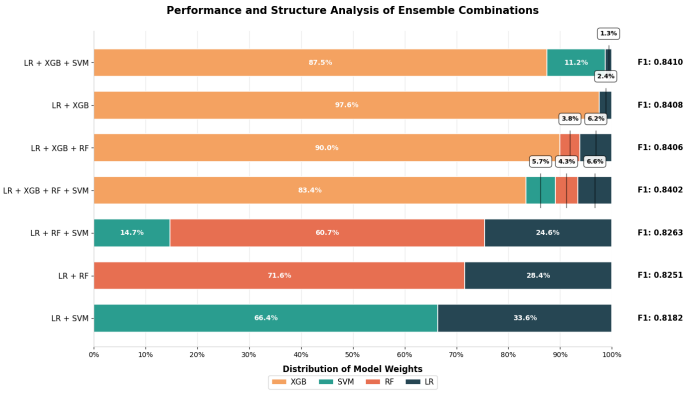

Figure 4 Performance and Structure Analysis of Ensemble Combinations

RESULT AND DISCUSSION

In this section, the results of the multi-stage optimization process carried out to improve the performance of the CTR forecasting model are presented and the results obtained are analyzed. Based on a baseline model created with the LR model, it is aimed to gradually increase performance through feature extraction, various hyperparameter optimization techniques, and finally ensemble modeling approaches that can capture nonlinear interactions, respectively. In this process, the performance of the ensemble structures was evaluated, where the LR model was refined with different optimization algorithms, followed by the inclusion of powerful models such as LR, XGBoost, RF, and SVM were incorporated, which exhibit complex and non-linear decision structures. F1 score and ROC-AUC values were used to measure model success at each stage of the study. First, the effects of hyperparameter optimization algorithms on the LR model are presented comparatively in the graph in Figure 5, followed by a comparative performance analysis of individually optimized models and ensemble structures consisting of various combinations of these models, detailing the approach that yields the highest success.

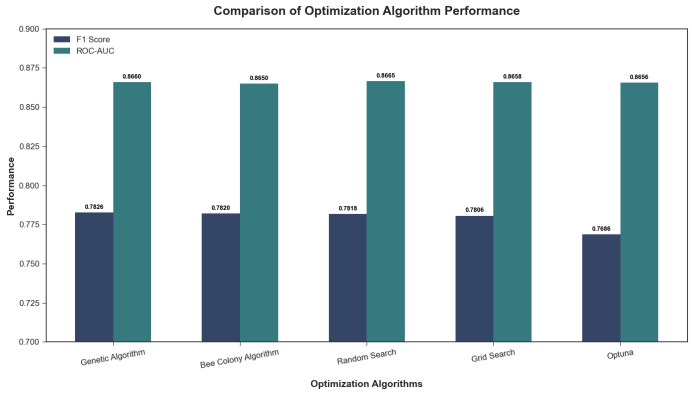

Figure 5 Performance of Hyperparameter Optimizations

After maximizing the performance of the baseline LR model through hyperparameter optimization, more powerful algorithms such as XGBoost, RF, and SVM were introduced to further enhance its predictive power, offering nonlinear decision structures. Various combinations of XGBoost, RF and SVM models were created with our base model LR and the comparative results of all ensemble combinations obtained after this optimization process are shown in Figure 5. ranked according to the F1 score. The performance of the different ensemble combinations tested is analyzed in Figure 4. When we analyze the graph, it is clear that the XGBoost model, which can model complex and nonlinear relationships more effectively, has the highest weight of all combinations involved. XGBoost is followed by SVM and Random Forest models, which also make significant contributions. LR, on the other hand, systematically has an extremely low weight in all combinations, which suggests that LR is not a direct predictor in these advanced structures, but rather a fundamental part that unifies the predictions of models. As a result, this analysis shows that the highest performance is achieved through an optimized combination of models, driven by XGBoost and powered by SVM.

In line with the expected results, a progressive improvement method such as feature extraction, hyperparameter optimization, and eventually ensemble modeling was found to be effective in the CTR prediction problem.It was observed that the holistic use of different optimization and model combination methods, which

is the hypothesis of the study, under complex and nonlinear model interactions will increase performance. The fact that the highest performance model obtained includes powerful algorithms such as XGBoost and SVM, which are frequently found successful in the literature, shows the robustness of the theoretical infrastructure of the study. The results obtained in this study are specific to the dataset used. The model's performance may vary across other advertising datasets with different distributions or features. Additionally, the complexity of the model and its inclusion of multiple algorithms increase the computational cost in training and prediction processes.

CONCLUSION

This study presents a machine learning approach that combines feature extraction, hyperparameter optimization, and ensemble modeling techniques to solve the CTR prediction problem. In the first stage, a base model was created on the dataset using Logistic Regression, followed by deriving new features from temporal and textual data through TF-IDF and Word2Vec methods. Hyperparameter adjustments were performed on the LR model using different optimization algorithms, and in the final stage, weight-optimized ensemble structures including LR, XGBoost, Random Forest, and SVM models were developed. As a result of combining the models by weighting them according to the F1 scores, an F1 score of 0.8694 was obtained and it was observed that the ensemble modeling approach offered higher and more stable performance instead of single models. However, since the development and testing processes of the study were conducted solely on a single dataset obtained from the Kaggle platform, the generalizability of the model is limited by the statistical distribution of the dataset used, the feature structure, and the frequency distribution of CTR rates.

In order to reduce this limitation, it is important to perform cross-dataset validation on CTR datasets obtained from different platforms or different time intervals in future studies to evaluate the durability of the model against different data distributions and user behavior patterns. Additionally, to mitigate the negative effects of distribution differences between datasets, implementing feature normalization, domain adaptation, and resampling techniques can enhance the model's generalization capability by preventing it from overfitting a specific data source. Instead of just a single training-test separation in the model evaluation process, using repeated cross-validation strategies with different random bins will make performance metrics more reliable and generalizable. Although all features were used directly in the current study, high-dimensional feature spaces increase computational cost, increase model complexity, and strengthen the risk of overlearning. Therefore, the application of feature reduction techniques in future studies offers significant potential for improvement.

Eliminating unnecessary or highly correlated features with feature selection-based methods makes the model simpler and more interpretable; Feature selection-based methods eliminate unnecessary or highly correlated features, making the model simpler and more interpretable; while feature extraction-based methods allow for a more meaningful representation of the data. This representation can be high-dimensional or low-dimensional, depending on the method used. However, since it is critical to keep information loss under control in this process, the effects of dimensionalization methods on model accuracy, computational cost, and generalization performance should be analyzed comparatively. Accordingly, the inclusion of different dimensionality reduction algorithms and alternative gradient boosting methods such as CatBoost and LightGBM in the ensemble structure and testing them on larger and different CTR datasets will more comprehensively demonstrate the effectiveness of the proposed approach in practical applications. LightGBM offers structural optimizations to improve computational efficiency in large-scale and high-dimensional datasets (Ke *et al.* 2017), while CatBoost reduces the need for preprocessing due to its architecture that can handle categorical variables directly, allowing for the development of more stable models (Pemila *et al.* 2024). The integration of these models into the community structure can lead to meaningful gains in both scalability and generalization performance.

Ethical standard

The authors have no relevant financial or non-financial interests to disclose.

Availability of data and material

The data that support the findings of this study are available from the corresponding author upon reasonable request.

Conflicts of interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

LITERATURE CITED

AgencyAnalytics, 2025 Click-through rate (ctr) definition. https://agencyanalytics.com/kpi-definitions/click-through-rate-ctr.

Akiba, T., S. Sano, T. Yanase, T. Ohta, and M. Koyama, 2019 Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD '19)*, pp. 2623–2631, ACM.

AlAli, M., M. AlQahtani, A. AlJuried, T. AlOnizan, D. Alboqaytah, *et al.*, 2021 Click-through rate effectiveness prediction on mobile ads using extreme gradient boosting. Computers, Materials & Continua **66**: 1681–1696.

Bergstra, J. and Y. Bengio, 2012 Random search for hyper-parameter optimization. Journal of Machine Learning Research **13**: 281–305.

Bird, S., E. Klein, and E. Loper, 2009 *Natural Language Processing with Python*. O'Reilly Media, Inc., Sebastopol, CA.

Bratus, O. S. and P. I. Bidyuk, 2023 Towards click-through rate prediction in online advertising. Problems of Applied Mathematics and Mathematical Modeling **23**: 3–17.

Cortes, C. and V. Vapnik, 1995 Support-vector networks. Machine Learning **20**: 273–297.

Gangopadhyay, B., Z. Wang, A. S. Chiappa, and S. Takamatsu, 2025 Adaptive budget optimization for multichannel advertising using combinatorial bandits. arXiv preprint arXiv:2502.02920 .

Goldberg, D. E., 1989 *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley, Reading, MA.

Harris, C. R., K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, *et al.*, 2020 Array programming with NumPy. Nature **585**: 357–362.

Hunter, J. D., 2007 Matplotlib: A 2d graphics environment. Computing in Science & Engineering **9**: 90–95.

Karaboga, D. and B. Basturk, 2007 A powerful and efficient algorithm for numerical function optimization: Artificial bee colony (abc) algorithm. Journal of Global Optimization **39**: 459–471.

Ke, G., Q. Meng, T. Finley, T. Wang, W. Chen, *et al.*, 2017 Lightgbm: A highly efficient gradient boosting decision tree. In *Advances in Neural Information Processing Systems*, volume 30.

Lou, J., 2024 Comparative analysis of logistic regression, random forest, and xgboost for ctr prediction in digital advertising. In *Proceedings of MIED 2024*, Atlantis Press.

McKinney, W., 2010 Data structures for statistical computing in python. In *Proceedings of the 9th Python in Science Conference*, pp. 56–61.

Pedregosa, F., G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, *et al.*, 2011 Scikit-learn: Machine learning in python. Journal of Machine Learning Research **12**: 2825–2830.

Pemila, M., R. K. Pongiannan, R. Narayanamoorthi, K. M. AboRas, and A. Youssef, 2024 Application of an ensemble catboost model over complex dataset for vehicle classification. PLOS ONE **19**: e0304619.

Řehůřek, R. and P. Sojka, 2010 Software framework for topic modelling with large corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pp. 45–50.

Rojas Guillen, J. M., 2024 *Click Through Rate Prediction Leveraging Machine Learning Techniques for Mobile Digital Advertisement*. Master's thesis, Lund University.

Şenel, S. and B. Alatlı, 2014 Lojistik regresyon analizinin kullanıldıgı makaleler üzerine bir inceleme. Journal of Measurement and Evaluation in Education and Psychology **5**: 35–52.

Shams, M. Y., A. M. Elshewey, E.-S. M. El-kenawy, A. Ibrahim, F. M. Talaat, *et al.*, 2024 Water quality prediction using machine learning models based on grid search method. Multimedia Tools and Applications **83**: 35307–35334.

swekerr, 2024 Click-through rate prediction. Kaggle Datasets .

The pandas development team, 2020 pandas-dev/pandas: Pandas.

Waskom, M. L., 2021 Seaborn: Statistical data visualization. Journal of Open Source Software **6**: 3021.

Yang, Y. and P. Zhai, 2022 Click-through rate prediction in online advertising: A literature review. Information Processing & Management **59**: 102853.

Zang, X., 2019 Click prediction for p2p loan ads based on support vector machine. Journal of Physics: Conference Series **1168**: 032042.

How to cite this article: Çağ, C., Akbulut, N., and Çankırlı, Y. Ad-Click Prediction Enhanced by Nonlinear Dynamics-Inspired Feature Extraction and Ensemble Optimization. *Chaos and Fractals*, 3(1), 38-46, 2026.