ISSN: 2454-9940

INTERNATIONAL JOURNAL OF APPLIED SCIENCE ENGINEERING AND MANAGEMENT

E-Mail :
editor.ijasem@gmail.com
editor@ijasem.org

www.ijasem.org

INTERNATIONAL JOURNAL OF APPLIED SCIENCE ENGINEERING AND MANAGEMENT

ISSN2454-9940

www.ijasem.org

Vol 18, Issue 2, 2024

Understanding the Factors Influencing Click-Through Rates in Online Advertising

Sumrana Siddiqui
Assistant Professor
Department of CSE
DCET
Affiliated to Osmania University
Hyderabad, Telangana
sumranasiddiqui@deccancollege.ac.in

Mohammed Abdul Haseeb
Student
Department of CSE
DCET
Affiliated to Osmania University
Hyderabad, Telangana
abdhaseeb653@gmail.com

Mohammed Shariquddin
Student
Department of CSE
DCET
Affiliated to Osmania University
Hyderabad, Telangana
shariq6694@gmail.com

Ahmed Ashfaq
Student
Department of CSE
DCET
Affiliated to Osmania University
Hyderabad, Telangana
ahmedashfaq619352@gmail.com

Abstract:
We propose to build a model for optimization of Advertisements Click Through Rate using Machine Learning techniques. Click Through Rate commonly known as CTR, is the ratio of users who click on a specific link to the number of total users who view a page, or advertisement, it is generally used to measure the success of online Advertisement and the effectiveness of website. CTR varies largely for different websites. As the CTR increases the effectiveness of the website increases, this gives marketers the valuable information required to measure the efficiency and popularity of the website on a search engine. Advertisements nowadays are extensively dependent on CTR for planning their budget and strategy, even if they are rarely appearing. As the advertisements are scattered over internet calculating their CTR becomes a hectic job. This problem is considered as Multi Armed Bandit problem and can be solved using Reinforcement Learning. Thompson sampling is a heuristic that uses explore and exploit principle to address multi armed bandit problem. It uses training information that evaluates the actions taken rather than instructs by giving correct actions. This is what creates the need for active exploration, for an explicit trialand-error search for good behavior. Based on the results of those actions, rewards (1) or penalties (0) are given for that action to the machine. Further actions are performed in order to maximize the reward that may improve future performance. Upper Confidence Bound Algorithm uses A/B test. It is based on the principle of optimism in the face of uncertainty, which is to choose your actions as if the environment is as nice as is plausibly possible. This we mean that the unknown mean payoffs of each arm as large as plausibly possible based on the data that has been observed.

Keywords: Machine Learning, Click Through Rate, Online Advertising, Upper Confidence Bound, Thompson Sampling, Python, Pandas, NumPy, Scikit-learn, Jupyter Notebook, Data Visualization, Ad Optimization, User Engagement, A/B Testing.

I. Introduction:

The unfolding of modern technologies and preponderance of Digital Marketing, businesses are doing all that they can to match up the pace. Businesses are either changing their plans of action into the digital one, or amplifying existing marketing strategies with digital advertising techniques- And the first question that may arise here is- Why Digital Marketing is important for your businesses followed by what is the role of Digital Marketing and what are the benefits of Digital Marketing. 34% of the businesses already had an integrated digital marketing plan in 2016. 72% marketers believe that traditional marketing is no longer sufficient and Digital Marketing will make their company revenue to be increased by 30% by the end of 2017 More than 80% of businesses will increase their digital marketing spending plan that may go beyond the IT budget.

The increasing of Click Through Rate (CTR) on digital advertisement can be the key to perform a dynamic advertisement module. A remarkable CTR is not only the most important thing in AdWords, but it is also extremely important for other marketing channels. These include organic search, social media, and email marketing. Higher the CTR higher is the quality score of the advertisements. Much higher Ad impression share can lead towards the impacts of CTR. The more your pages beat the expected organic CTR for a given position, the more likely you are to appear in prominent organic positions. Increasing your click-through rate will also increase your conversion rates. If 1 you can increase your CTR by 2x then your conversion rate should increase by 50 percent. Many of the social websites like Google and Facebook do use the concept of social Ads. CTR can also affect Email Marketing.

a. Problem Statement

- Maximizing Click Through Rate (CTR) is essential for enhancing user engagement and driving conversions in online advertising.
- Traditional ad placement methods often fail to dynamically adapt to changing user preferences and market conditions.
- The project aims to implement advanced machine learning algorithms, specifically Upper Confidence Bound (UCB) and Thompson Sampling.
- These algorithms will create a responsive and adaptive ad placement system.
- The system will leverage historical data and real-time user interactions to improve ad relevance.
- The ultimate goal is to increase CTR and boost the efficiency and effectiveness of online advertising campaigns.

b. Problem Definition

In recent years, advertising has become a significant element to the world of marketing business. A good advertisement not only attracts the customers but, also increases the likelihood of purchasing the products. In today's world, the 4 most commonly used advertising methods are Television broadcast, Newspaper, Radio and on-line advertising. It can be observed that, on-line advertising is constantly increasing its growth rate and outperforming all other approaches of advertising methods. The major reason for increase in the rate of on-line advertisement are worldwide reachable and accessible all the time 24 hours, 365 days and it is highly targeted and can be designed to focus a group of customers with similar interest.

II. Literature Survey

- Predicting Clicks: CTR Estimation of Advertisements Using Logistic Regression Classifier.

Author: Rohit Kumar, Sneha Manjunath Naik, Vani D Naik, Smitha Shiralli (2015).

The paper "CTR Estimation of Advertisements using Logistic Regression Classifier" by Rohit Kumar et al. addresses predicting click-through rates (CTR) of online ads using logistic regression. The model, trained on a large dataset, achieves about 90% accuracy by using ad position and impressions as predictors. The study emphasizes the benefits of online advertising, such as global reach, high targeting precision, and cost-efficiency. The authors conclude that logistic regression is effective for CTR prediction and suggest future research to include query-dependent features for enhanced accuracy and personalized advertising. The paper concludes that logistic regression is effective for CTR prediction and suggests future research directions, including incorporating query-dependent features to enhance prediction accuracy and personalizing advertising based on user behavior and query-ad similarity. A CTR Prediction Approach for Advertising Based on Embedding Model and Deep Learning The paper

- A CTR Prediction Approach for Advertising Based on Embedding Model and Deep Learning.

Author: Zilong Jiang, Shu Gao, Yunhui Shi, and Liangchen Chen (2018).

This paper presents a novel method for estimating click-through rates using advanced machine learning techniques. This approach leverages embedding models to convert sparse features into dense vectors, which are then processed by deep learning networks to predict CTR. The study demonstrates that this method outperforms traditional models by capturing complex interactions between features. The authors highlight the superior performance of their model in comparison to other baseline models, emphasizing its potential for real-world applications in online advertising. The research suggests that deep learning techniques, when combined with effective feature embedding, can significantly enhance the accuracy and efficiency of CTR predictions.

- A CTR Prediction Method Based on Feature Engineering and Online Learning.

Authors Chen Jie-Hao, Li Xue-Yi, Zhao Zi-Qian, and Shi Ji-Yun (2017)

This research paper explores a sophisticated technique for improving CTR predictions. Their method incorporates Field-aware Factorization Machines (FFM) to handle sparse feature vectors, enabling the model to account for interactions between different fields of data. This approach allows for more granular and accurate predictions by dynamically learning from incoming data in an online learning framework. The study demonstrates that integrating feature engineering with online learning mechanisms can substantially enhance the performance of CTR estimation models. The authors advocate for the adoption of FFM and similar methodologies in the development of advanced advertising systems to achieve higher predictive accuracy and adaptability.

INTERNATIONAL JOURNAL OF APPLIED SCIENCE ENGINEERING AND MANAGEMENT

ISSN2454-9940

www.ijasem.org

Vol 18, Issue 2, 2024

- Feature Selection Methods Evaluation for CTR Estimation.

Author: Luis Miralles-Pechuain, Hiram Ponce, and Lourdes Martinez-Villasen (2016).

This examines various feature selection techniques to optimize the performance of CTR prediction models. The study focuses on identifying the most influential features that contribute to predicting user clicks on advertisements. By evaluating different feature selection methods, the authors aim to enhance the efficiency and accuracy of CTR estimation. The research underscores the importance of selecting relevant features to reduce computational complexity and improve model interpretability. The authors conclude that effective feature selection is crucial for developing robust and scalable CTR prediction models, which can significantly impact the efficiency of advertising strategies by ensuring that advertisers only pay for actual user clicks, thus optimizing ad spend and targeting precision.

III. System Architecture

The system architecture of our CTR optimization project is designed to leverage advanced machine learning algorithms, specifically Upper Confidence Bound (UCB) and Thompson Sampling, to dynamically enhance the performance of online advertisements. This architecture integrates various components that work together to process large datasets, train models, and make real-time decisions to maximize Click Through Rate (CTR). The implementation involves using Python and its robust libraries such as Pandas, NumPy, and scikit-learn, alongside Jupyter Notebook for interactive data analysis and visualization. The architecture ensures a seamless flow from data ingestion and preprocessing to model training and deployment, enabling continuous learning and adaptation to user interactions. By incorporating UCB and Thompson Sampling, the system effectively balances exploration and exploitation, ensuring optimal ad placement and improved user engagement.

a. Components of System Architecture

- Data Ingestion and Preprocessing: This module is responsible for collecting and cleaning data from various sources. It ensures that the data is structured and ready for analysis, involving steps such as handling missing values, normalizing data, and feature engineering.
- Model Training with UCB and Thompson Sampling: In this phase, the cleaned data is used to train machine learning models based on UCB and Thompson Sampling algorithms. These models are designed to predict the probability of user clicks and optimize ad placements accordingly.

Thompson Sampling Algorithm:

Thompson Sampling is a heuristic for addressing the multi-armed bandit problem, which involves making decisions to maximize rewards by balancing exploration (trying new options) and exploitation (choosing the best-known option). In this context, each "bandit" or slot machine represents an ad variation, and the goal is to determine which ad will yield the highest click-through rate (CTR).

Initially, there is no data on the probability of success for each machine, so we start by exploring each machine one by one. After a certain number of observations, we gather data on the rewards (clicks) and penalties (no clicks) for each machine. Consider the table below, where each column represents a bandit (B1, B2, B3, B4, and B5). The 0s indicate no reward, and the 1s indicate a reward for each pull of the machine's lever.

Figure 1: Representation of Thompson Sampling Algorithm (octopus) trying to solve the Multi Armed Problem.

Upper Confidence Bound (UCB) Algorithm:

The Upper Confidence Bound (UCB) algorithm is a popular method for addressing the multi-armed bandit problem. It operates on the principle of optimism in the face of uncertainty, which means choosing actions as if the environment is as favorable as possible based on the data observed so far. In the context of online advertising, each "arm" or bandit represents a different ad variant, and the goal is to maximize the click-through rate (CTR) by strategically balancing exploration and exploitation.

INTERNATIONAL JOURNAL OF APPLIED SCIENCE ENGINEERING AND MANAGEMENT

ISSN2454-9940

www.ijasem.org

Vol 18, Issue 2, 2024

Initially, we assume all ads have the same average value since there is no prior information about their performance. As the algorithm proceeds, it updates the average reward for each ad based on user interactions. UCB creates an upper confidence bound for each ad, which combines the average observed reward with a measure of uncertainty. This bound determines which ad to display next.

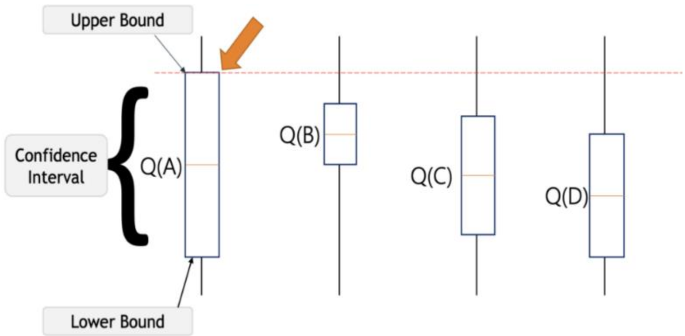

Figure 2: Representation of the Upper Bound Used in UCB Algorithm.

Initially, UCB explores more to systematically reduce uncertainty but its exploration reduces over time. Thus, we can say that UCB obtains greater reward on average than other algorithms such as Epsilon-greedy, Optimistic Initial Values, etc.

- Real-Time Ad Placement Optimization: Once trained, the models are deployed to make real-time decisions about which ads to display to users. The UCB and Thompson Sampling algorithms dynamically adjust these decisions based on ongoing user interactions and feedback.
- Performance Monitoring and Evaluation: This component continuously monitors the performance of the deployed models, tracking key metrics such as CTR, conversion rates, and overall ad effectiveness. It allows for ongoing evaluation and refinement of the models to ensure sustained optimization.
- Visualization and Reporting: Using tools like Jupyter Notebook and Matplotlib, this module provides interactive visualizations and reports that help stakeholders understand the performance of the ad campaigns and the impact of the optimization algorithms.

By incorporating these components, the system architecture ensures a comprehensive and efficient approach to CTR optimization, leveraging the strengths of UCB and Thompson Sampling to achieve superior results in online advertising.

IV. Technologies Used

- PYTHON

Python is a widely used general-purpose, high level programming language. It was initially designed by Guido van Rossum in 1991 and developed by Python Software 11 Foundation. It was mainly developed for emphasis on code readability, and its syntax allows programmers to express concepts in fewer lines of code. Python is an interpreted, high-level, general-purpose programming language. Python features a dynamic type system and automatic memory management. It supports multiple programming paradigms, including object-oriented, imperative, functional and procedural. It also has a comprehensive standard library. It is the world's fastest growing and most popular programming language used by software engineers, analysts, data scientists, and machine learning engineers alike.

It is used by sites like YouTube and Dropbox. It supports functional and structured programming methods as well as OOP. It can be used as a scripting language or can be compiled to byte-code for building large applications. It provides very high-level dynamic data types and supports dynamic type checking. It supports automatic garbage collection. It can be easily integrated with C, C++, COM, ActiveX, CORBA, and Java. Python uses whitespace indentation, rather than curly brackets or keywords, to delimit blocks. An increase in indentation comes after certain statements; a decrease in indentation signifies the end of the current block. Thus, the program's visual structure accurately represents the program's semantic structure.

- NUMPY

NumPy is the fundamental package for scientific computing with Python. It contains among other things:

  - Sophisticated (broadcasting) functions.
  - Tools for integrating C/C++ and FORTRAN code.
  - Useful linear algebra, Fourier transform, and random number capabilities.

Besides its obvious scientific uses, NumPy can also be used as an efficient multidimensional container of

INTERNATIONAL JOURNAL OF APPLIED SCIENCE ENGINEERING AND MANAGEMENT

ISSN2454-9940

www.ijasem.org

Vol 18, Issue 2, 2024

generic data. Arbitrary data-types can be defined. This allows NumPy to seamlessly and speedily integrate with a wide variety of databases. Using NumPy in Python gives functionality comparable to MATLAB since they are both interpreted, and they both allow the user to write fast programs as long as most operations work on arrays or matrices instead of scalars. SciPy is a library that adds 12 more MATLAB-like functionality and Matplotlib is a plotting package that provides MATLAB-like plotting functionality. NumPy is licensed under the BSD license, enabling reuse with few restrictions. Python bindings of the widely used computer vision library OpenCV utilize NumPy arrays to store and operate on data. Since images with multiple channels are simply represented as three-dimensional arrays, indexing, slicing or masking with other arrays are very efficient ways to access specific pixels of an image.

- SCIKIT-LEARN

Scikit-learn is a free software machine learning library for the Python programming language. It features various classification, regression and clustering algorithms including support vector machines, random forests, gradient boosting, kmeans and DBSCAN, and is designed to interoperate with the Python numerical and scientific libraries NumPy and SciPy. Simple and efficient tools for data mining and data analysis. Accessible to everybody, and reusable in various contexts. Built on NumPy, SciPy, and Matplotlib.

- JUPYTER NOTEBOOK

Jupyter Notebook is an open-source web application that allows you to create and share documents that contain live code, equations, visualizations and narrative text. Uses include: data cleaning and transformation, numerical simulation, statistical modeling, data visualization, machine learning, and much more. Jupyter Notebook's versatility and interactivity make it an ideal tool for developing and optimizing Click Through Rate (CTR) models for online advertisements. Its ability to integrate live code with visualizations allows data scientists to iteratively test and refine machine learning algorithms such as Upper Confidence Bound (UCB) and Thompson Sampling. This iterative process is crucial for effectively balancing the exploration of new ad variations with the exploitation of known high-performing ads. By using Jupyter Notebook, the project can leverage real-time visual feedback to monitor model performance, adjust parameters, and implement enhancements swiftly.

Additionally, Jupyter Notebook supports comprehensive data analysis workflows, which are essential for handling large datasets typically encountered in online advertising. With built-in support for popular data science libraries like NumPy, pandas, scikit-learn, and TensorFlow, Jupyter Notebook enables seamless data cleaning, transformation, and visualization. These capabilities are vital for understanding user behavior, extracting meaningful insights, and training robust machine learning models. By facilitating transparent and reproducible research, Jupyter Notebook ensures that all stages of the CTR optimization process—from initial data exploration to final model deployment are documented and easily accessible for review and collaboration.

- MACHINE LEARNING:

Machine learning is an application of artificial intelligence (AI) that provides systems the ability to automatically learn and improve from experience without being explicitly programmed. Machine learning focuses on the development of computer programs that can access data and use it to learn for themselves.

Machine learning plays a pivotal role in optimizing Click Through Rate (CTR) for online advertisements. By leveraging machine learning algorithms such as Upper Confidence Bound (UCB) and Thompson Sampling, advertisers can dynamically adjust their strategies based on real-time data and user interactions. These algorithms help in identifying patterns and trends in user behavior, allowing for more accurate predictions of which ads are likely to be clicked. This continuous learning and adaptation process ensures that the ad placements remain relevant and effective, ultimately leading to higher engagement rates and improved return on investment (ROI) for advertising campaigns. Machine learning's ability to process vast amounts of data and make informed decisions on the fly makes it an invaluable tool for enhancing the performance of online advertising systems.

ISSN2454-9940

www.ijasem.org

Vol 18, Issue 2, 2024

- PANDAS:

Pandas is an open-source, BSD-licensed library that provides high-performance, easy-to-use data structures and data analysis tools for the Python programming language. It is a crucial library for data manipulation and analysis, offering data structures like Series and DataFrame, which are essential for handling and processing structured data. With Pandas, users can efficiently perform operations such as merging, reshaping, selecting, and cleaning data, making it an indispensable tool for data scientists and analysts.

- MATPLOTLIB:

Matplotlib is a comprehensive Python library used to create static, animated, and interactive visualizations in Python. It is widely used for generating 2D graphs and plots, making it an essential tool for data visualization. One of its core modules, PYPLOT, simplifies the process of creating plots by providing a MATLAB-like interface, which includes features to control line styles, font properties, formatting axes, and more.

With Matplotlib, users can create a wide range of static and dynamic visualizations, including line plots, bar charts, histograms, scatter plots, and heatmaps. This flexibility allows data scientists and analysts to visually explore their data, identify trends, and communicate findings effectively. For example, in the context of optimizing Click Through Rate (CTR) for online advertisements, Matplotlib can be used to visualize the performance of different ads, track changes in CTR over time, and compare the effectiveness of various machine learning modules.

V. Module Description

- Data Collection and Preprocessing

This module is responsible for gathering raw data related to advertisements and user interactions, such as clicks, impressions, and timestamps. Using Pandas and NumPy, the data is cleaned, transformed, and structured into a format suitable for analysis. This involves handling missing values, encoding categorical features, and normalizing numerical data to ensure consistency and reliability.

- Exploratory Data Analysis (EDA)

The EDA module leverages Matplotlib to create visual representations of the data. This includes plotting histograms, scatter plots, and heatmaps to identify trends, patterns, and correlations. EDA helps in understanding the underlying distribution of the data and guides feature selection and engineering for the machine learning models.

- Feature Engineering

In this module, new features are created to enhance the predictive power of the models. Techniques such as interaction terms, polynomial features, and aggregation are applied to derive meaningful insights from the raw data. This module ensures that the machine learning algorithms have access to the most relevant and informative features.

- Model Implementation

This module focuses on implementing various machine learning algorithms using Scikit-learn. Logistic regression, Thompson Sampling, and Upper Confidence Bound (UCB) algorithms are deployed to predict the click-through rate (CTR) of ads. Each model is trained, validated, and tested to ensure accuracy and robustness.

- Thompson Sampling and UCB Bandit Algorithms

Dedicated to the multi-armed bandit problem, this module implements Thompson Sampling and UCB algorithms to dynamically select ads. By balancing exploration and exploitation, these algorithms continuously optimize ad selection based on historical performance data. This module ensures that the best-performing ads are displayed more frequently, improving overall CTR.

- Model Evaluation and Optimization

This module evaluates the performance of the implemented models using metrics such as accuracy, precision, recall, and the Area Under the Receiver Operating Characteristic Curve (AUC-ROC). Hyperparameter tuning and cross-validation techniques are applied to optimize model

ISSN2454-9940

www.ijasem.org

Vol 18, Issue 2, 2024

parameters and enhance predictive accuracy. Visualization tools like confusion matrices and ROC curves are used to interpret and compare model performance.

- Deployment and Monitoring

The deployment module integrates the trained models into a production environment. Using Jupyter Notebook, the models are deployed in a way that allows real-time predictions and ad selections. This module also includes mechanisms for monitoring model performance and updating models with new data to maintain accuracy over time.

- User Interface and Reporting

This module focuses on creating a user-friendly interface for stakeholders to interact with the system. Dashboards and reports are generated to provide insights into ad performance, model predictions, and overall system efficacy. The interface is designed to facilitate easy interpretation of results and support decision-making processes.

VI. Implementation

Implementation of Thompson Sampling

Thompson Sampling algorithm has been around for a long time. It was first proposed in 1933 by William R. Thompson. Though the algorithm has been ignored and has had the least attention for many years after its proposal, it has recently gained traction in many areas of Artificial Intelligence. Some of the areas where it has been used are revenue management, marketing, web site optimization, A/B testing, advertisement, recommendation system, gaming and more. Thompson Sampling is an algorithm that follows exploration and exploitation to maximize the cumulative rewards obtained by performing an action. Thompson Sampling is also sometimes referred to as Posterior Sampling or Probability Matching. The action is performed multiple times which is called exploration and based on the results obtained from the actions, either rewards or penalties, further actions are performed with the goal to maximize the reward which is called exploitation. It can also be understood as new choices are explored to maximize rewards while exploiting the already explored choices.

Initially, we do not have any data on how the probability of success is distributed among the machines. So, we start by exploring the machines one by one. So, after some “n” number of observations we have data similar to what’s shown below;

INTERNATIONAL JOURNAL OF APPLIED SCIENCE ENGINEERING AND MANAGEMENT

ISSN2454-9940

www.ijasem.org

Vol 18, Issue 2, 2024

INDEX	B1	B2	B3	B4	B5
0	1	1	1	1	1
1	0	1	1	0	0
2	0	0	1	1	1
3	1	0	0	0	1
4	1	0	1	1	0
5	0	0	1	0	1
6	1	1	1	1	1
7	1	0	1	1	1
8	1	1	1	0	0
9	1	1	0	0	0
10	0	0	0	1	0
11	1	1	0	0	1
12	1	1	1	1	1
13	0	1	0	0	0

Figure 3: Implementation of Thompson Sampling Algorithm for Multi Slot Machine.

Each column represents a bandit or slot machine (B1, B2, B3, B4, and B5). The 0's represent penalties (no reward), and the 1's represent rewards. We have "n" observations, meaning each machine is pulled n times. For Thompson Sampling, we initialize the number of observations and machines. Each machine is selected based on the beta distribution of rewards and penalties. We maintain lists for the selected machines, rewards, and penalties, all initialized to zero.

For each observation, we iterate through the machines and select the one with the highest random beta distribution. We update the "machine selected" list accordingly. If the selected machine returns a reward, we increment the reward list for that machine; if it's a penalty, we increment the penalty list. The total number of rewards is updated, playing a crucial role in determining the CTR. This process continues until all observations are processed, ensuring the most effective machines are identified based on historical performance.

Implementation of Upper Confidence Bound (UCB)

Upper Confidence Bound (UCB) is the most widely used solution method for multi armed bandit problems. This algorithm is based on the principle of optimism in the face of uncertainty. In other words, the more uncertain we are about an arm, the more important it becomes to explore that arm. Since all the ads/bandits are same in the beginning, we cannot distinguish or discriminate any arm or ad. So, we assume they all have same observed values. At the, we don't know what is the best arm or ad. So, the UCB algorithm assumes they all have the same observed average value. Then it creates confidence bound for each arm or ad.

So, it randomly picks any of the arms or ads. Then two things will happen- the user will click the ad or the arm gives reward or does not. If he doesn't click on an ad, the observed average of the ad or arm will go down. And the confidence bound will also go down. If it had clicked, the observed average would go up and the confidence bound also goes up. By exploiting the best one we can decrease the confidence bound.

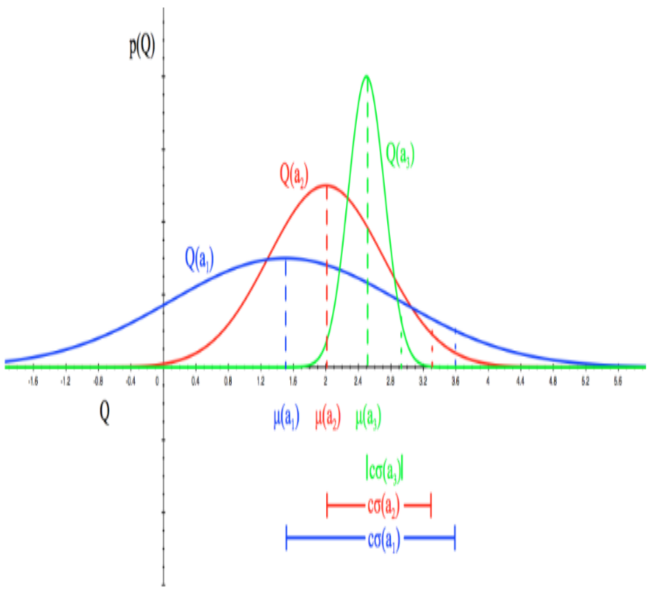

Figure 4: Representation of the UCB Algorithm for Multiple Options.

INTERNATIONAL JOURNAL OF APPLIED SCIENCE ENGINEERING AND MANAGEMENT

ISSN2454-9940

www.ijasem.org

Vol 18, Issue 2, 2024

UCB ACTION SELECTION

Upper-Confidence Bound action selection uses uncertainty in the action-value estimates for balancing exploration and exploitation. Since there is inherent uncertainty in the accuracy of the action-value estimates when we use a sampled set of rewards thus UCB uses uncertainty in the estimates to drive exploration.

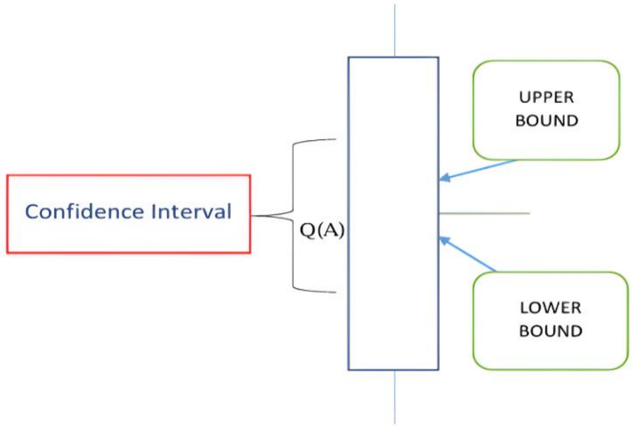

Figure 5: Representation of the Confidence Interval and the Upper and Lower Bounds.

Q(A) in the above picture represents the current action-value estimate for action A. The brackets represent a confidence interval around Q*(A) which says that we are confident that the actual action-value of action A lies somewhere in this region. The lower bracket is called the lower bound, and the upper bracket is the upper bound. The region between the brackets is the confidence interval which represents the uncertainty in the estimates. If the region is very small, then we become very certain that the actual value of action A is near our estimated value. On the other hand, if the region is large, then we become uncertain that the value of action A is near our estimated value.

The Upper Confidence Bound follows the principle of optimism in the face of uncertainty which implies that if we are uncertain about an action, we should optimistically assume that it is the correct action. For example, let's say we have these four actions with associated uncertainties in the picture below, our agent has no idea which is the best action. So according to the UCB algorithm, it will optimistically pick the action that has the highest upper bound i.e. A.

VII. Result

The timely report obtained from the dataset could help us to give a brief description of the output given by the model. This would summarize which day users would click on advertisements the most.

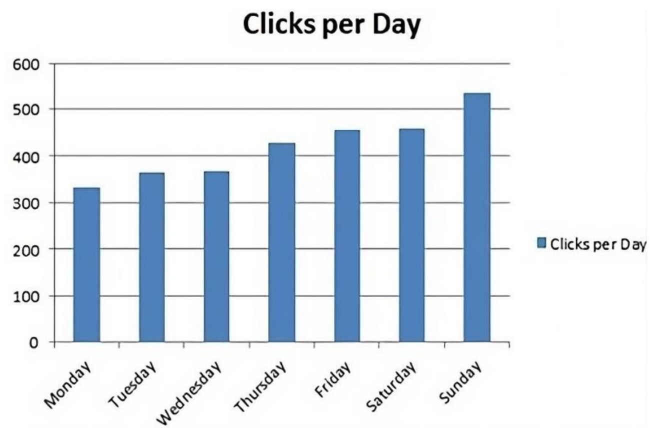

Figure 6: Clicks per Day taken from the Dataset.

The above figure shows that SUNDAY was the day where the most clicks of the advertisements occurred i.e. users accessed a greater number of websites on SUNDAY comparatively and the least clicks occurred on MONDAY. Now coming to the relationship between users and searches the number of unique search users were 38003 and the total number of users were 206079.

Then we have the filter users and non-filter users search percentage which gives us the value of relationship between the user type and their percentages.

The analysis of the Advertisement Click Through Rate (CTR) Optimization project reveals insightful patterns in user interactions with online ads. The data indicates that user Behavior varies significantly across different days of the week, with Sundays showing the highest ad click activity. Moreover, the study uncovers the types of users who are more likely to engage with ads, highlighting the importance of search queries in predicting ad clicks.

User Types and their Percentage

S. No	User Type	Percent
1	Search Users	1.30
2	Filter Users	0.59
3	No Search or Filter	0.50

Figure 7: Percentage of the Types of Users.

The above table complies that the higher percentage comprises of search users CTR which is more compared to the users who uses filters or don't use filters or searches at all. The least CTR could be found for the users who don't use searches or filters at all i.e. 0.50 and the search user's percentage is 1.30. We also could find that only 18% of users enter a search query which isn't too surprising because we expect most people who first land on a site to be just browsing.

Since users who are just exploring and aren't sure what they want to buy, we can help them narrow down the options by providing ads that are relevant based on their search behavior.

Top Item Searches

Item	Count
Bicycle	1297
Sofa	609
Stroller	491
Shoes	466
Bag	450

Figure 8: Top Five (5) Searched Items in the Dataset.

With this we could find that the most item searches where the category was taken into consideration was BIKE category which had 1297 searches and the SOFA category had 609 making it the second highest top item search. There are 35% of users searching for BIKE category which would also draw a conclusion that it can be the duration of summer where users might've needed this particular item.

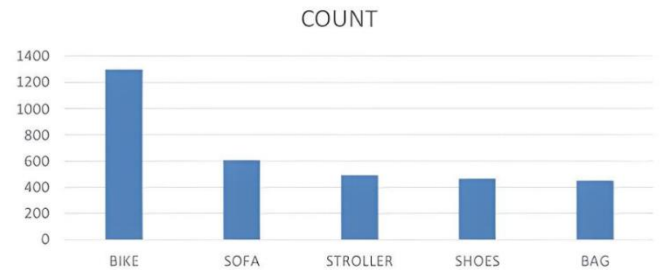

Figure 9: Representation of the Count of Searches of the Top Five (5) Items Searched.

We can now further explore the types in the BIKE category and fetch its impressions and get a generalized view of different Advertisement IDs that fall under this category.

Advertisement ID	Impressions
6115044	200
13485461	139
36414129	93
21649512	77
33940616	74

Figure 10: Top Five (5) Ads and their Respective Counts.

In the above table the Advertisement IDs and their impressions are represented which shows that 6115004 has the highest number of impressions. Whereas Fig 45 shows us the unique search query percentage where the pie chart suggests that more than 34% is given to 6115044.

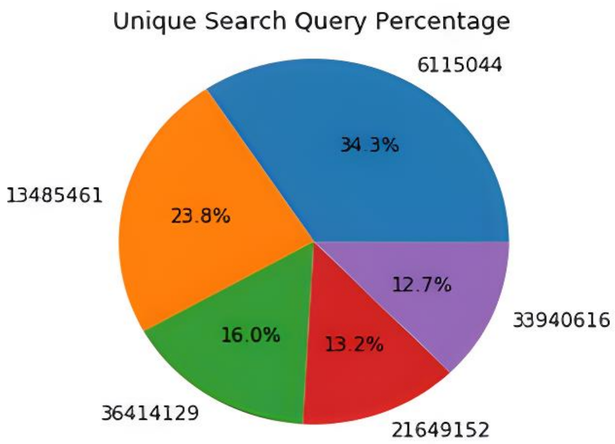

Figure 11: Representation of the Unique Search Query Percentage in a Pie Chart.

INTERNATIONAL JOURNAL OF APPLIED SCIENCE ENGINEERING AND MANAGEMENT

ISSN2454-9940

www.ijasem.org

Vol 18, Issue 2, 2024

Representation between the Impressions and their CTR:

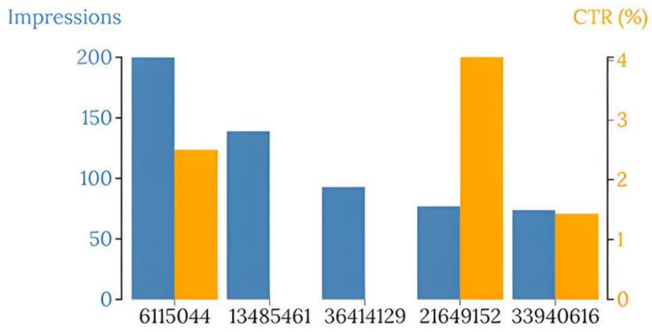

Figure 12: Graphical Representation of the Relation between Impressions and their CTR.

The top five impressions for bikes exhibit considerable variability in CTRs. While the highest impression ad has a 2.2% CTR, the second and third ads show a 0% CTR. Interestingly, the fourth most viewed ad boasts the highest CTR at 4%, highlighting the need for improvement. By associating different bike names with Advertisement IDs, we can treat them as "machines" for reinforcement learning algorithms.

Utilizing Thompson Sampling, which balances exploration and exploitation, ads are randomly selected from different machines. The algorithm searches for 1s in different machines and exploits the one with positive outcomes. For instance, encountering two 1s from machine 2 resulted in the highest CTR, with a total of 5 rewards. This approach optimizes ad selection by prioritizing machines with the highest probability of success.

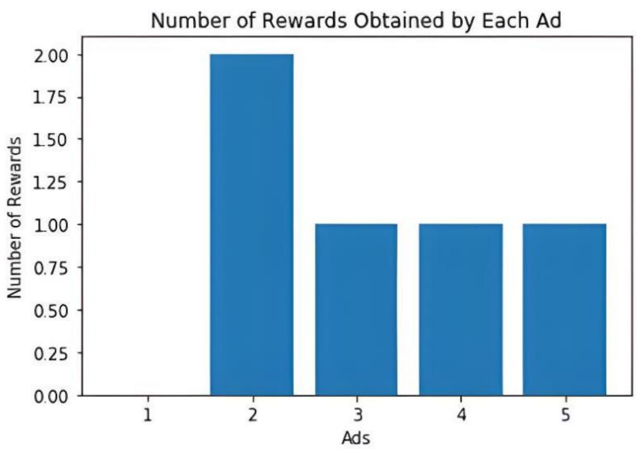

Figure 13: Number of Rewards Obtained by Each Advertisement.

Now coming to the UPPER CONFIDENCE BOUND (UCB) where the max_upper_bound is 0 and the upper_bound is 1e400. Now coming to the number_of_selections we could obtain the following information [106, 113, 112, 106, 119] these are the selections that UCB took from five different machines. The average reward is found to be 0.016806722 and the sum_of_rewards is [0, 1, 1, 0, 2] that makes the total_reward to be 4.

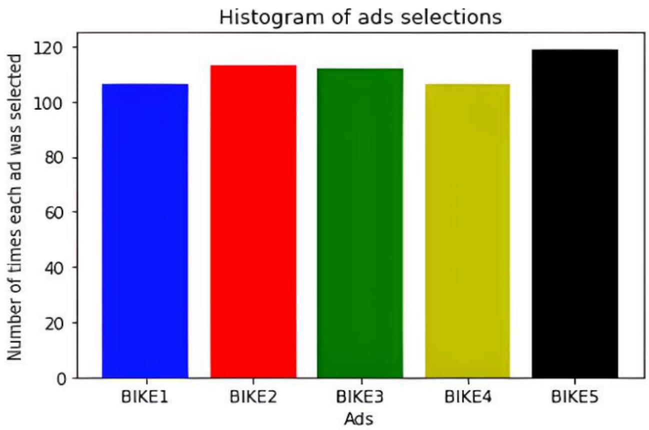

Figure 14: Graphical Representation of the Bike Ad Selection.

VIII. Conclusion

Estimation of click-through rates (C.T.R) have been examined from a mixed bag of points before as the most common method of advertising is on-line advertising shows the comparison of various advertising methods over the past five years. It can be observed that, on-line advertising is constantly increasing its growth rate and outperforming all other approaches of advertising methods. Research work has been carried on to predict the number of clicks per ad, the variables that contribute to the CTR, and predicting which advertisements ought to be shown amid web surfing. The techniques selected for estimation of CTR were Thompson Sampling and UCB. Although structural equation model approach, maximum likelihood estimates with keyword and hierarchical clustering, neural networks, Social Intentional Corpora, Bayesian networks, feature selection, and modelling trees can be further used for different scopes. This model can help the advertising agents or marketing supporters to create different tactics and help them to adapt and make people rely on the upcoming problems which can be tackled.

The objective in future is to discover insights in user behaviour related to ad clicks as well as build a learning algorithm that will predict whether a user will click on a given ad. This will also lead to overall website activity which will be given to the admins who are responsible in handling the back-end operations.

IX. Future Scope

Future enhancements for our project could involve implementing dynamic ad content generation based on user behaviour, personalized ad targeting using predictive analytics models, and experimenting with multivariate testing techniques to identify the most effective ad creatives. Additionally, integrating contextual targeting methods and advanced attribution modelling could further enhance ad relevance and accuracy.

Real-time bidding optimization algorithms and AI-driven ad placement can automate bid strategies and optimize ad placements across multiple channels, ensuring maximum ROI and user engagement. Cross-channel integration of data from various sources can provide a holistic view of the customer journey, enabling more effective ad placements and campaign optimization. These enhancements aim to keep our project at the forefront of digital advertising technology, delivering personalized and efficient ad experiences for users and advertisers.

X. References

[1]. Zilong Jiang, Shu Gao, Yunhui Shi and Liangchen Chen, "A CTR Prediction Approach for Advertising Based on Embedding Model and Deep Learning" - IEEE Conference on Parallel and Distributed Processing, (2018).

[2]. Luis Miralles-Pechuán, Hiram Ponce and Lourdes Martínez-Villaseñor, "Feature Selection Methods Evaluation for CTR Estimation" - Fifteenth Mexican International Conference on Artificial Intelligence, (2016).

[3]. Rohit Kumar, Sneha Manjunath Naik, Smita Shiralli and Moula Husain, "Predicting clicks: CTR estimation of advertisements using Logistic Regression classifier" - IEEE International Advance Computing Conference (IACC), (2015).

[4] Chen Jie-Hao, Li Xue-Yi, Zhao Zi-Qian, Shi Ji-Yun, "A CTR prediction method based on feature engineering and online learning" – Internation Symposium on Communications and Information Technologies (2017).

[5] Sen Zhang, Zheng Liu, Wendong Xiao, "A Hierarchical Extreme Learning Machine Algorithm for Advertisement ClickThrough Rate Prediction" - IEEE Access, (2018).

[6] Anh-Phuong Ta, "Factorization machines with follow-the-regularized-leader for CTR prediction in display advertising" – IEEE International Conference on Big Data (2015).

[7] Estimating Click Through Rate Using Reinforcement Learning Algorithms (2023) - Zhipeng Fang, Kun Yue, et al.

[8] Web-Scale Bayesian Click-Through Rate Prediction for Sponsored Search Advertising in Microsoft's Bing Search Engine (2010) - Lihong Li, Wei Chu, et al.

[9] Improved Online Learning Algorithms for CTR Prediction in Ad Auctions (2023) - Jiafan Yu, Jingyi Zhang, et al.

[10] Ad Optimization Via Machine Learning: A Focus on Upper Confidence Bound and Thompson Sampling Algorithms (2023) - Available on ResearchGate.

[11] Web Ads Optimization using Reinforcement Learning Techniques (2023) - Amandeep Kaur, Himanshu Sharma, et al.

[12] Contextual Bandit Algorithms for Real-Time Bidding (2019) - Shuai Li, Lihong Li, et al.

[13] Thompson Sampling for Real-Time Personalized Advertising (2017) - Lihong Li, Wei Chu, et al.

[14] Contextual Bandit Algorithms for Real-Time Bidding (2019) - Shuai Li, Lihong Li, et al. This paper delves into the application of contextual bandit algorithms, specifically UCB and Thompson Sampling, in the dynamic world of real-time bidding for online advertising. It offers a comprehensive analysis of their performance and effectiveness in this context.

[15] Thompson Sampling for Real-Time Personalized Advertising (2017) - Lihong Li, Wei Chu, et al. This work focuses on the use of Thompson Sampling for real-time personalized advertising, highlighting its ability to strike a balance between exploration and exploitation. It emphasizes the importance of adapting to user behaviour and context for improved ad targeting.

INTERNATIONAL JOURNAL OF APPLIED SCIENCE ENGINEERING AND MANAGEMENT

ISSN2454-9940

www.ijasem.org

Vol 18, Issue 2, 2024

[16] A Contextual Bandit Approach to Personalized News Article Recommendation (2010) - Lihong Li, Wei Chu, et al. This paper demonstrates the potential of the contextual bandit framework, including UCB and Thompson Sampling, for personalized news article recommendation. It showcases how these algorithms can enhance user engagement and content discovery.

[17] Optimistic Bandit Learning for Online Advertising (2019) - Olivier Chapelle and Lihong Li. This book chapter provides a detailed explanation of optimistic bandit learning, the broader framework encompassing UCB and Thompson Sampling. It offers valuable insights into the theoretical foundations and practical applications of these algorithms in online advertising.

[18] Bandit Algorithms for Online Advertising: A Tutorial (2016) - Shuai Li, Lihong Li, et al. This tutorial serves as a clear and concise introduction to bandit algorithms, including UCB and Thompson Sampling, specifically in the context of online advertising. It covers the key concepts, algorithms, and their practical implementation in real-world scenarios.