Optimizing Online Advertising Using Adaptive Learning Techniques

Course Project For 15.S07 SSIM: Real-time Analytics for Digital Platforms

Haibu Shi haibu239@mit.edu
Ellie Yang ellie_y@mit.edu
Yu He yh527@mit.edu
Yuki Yu yuki620@mit.edu

Executive Summary

In the evolving landscape of digital advertising, optimizing ad placements and bidding strategies in real-time remains a significant challenge for businesses. Traditional methods relying on historical data and static models often lead to suboptimal ad targeting and inefficient budget allocation. This project leverages adaptive learning techniques, including machine learning models and reinforcement learning-based Multi-Armed Bandits (MAB), to enhance ad click-through rates (CTR) and maximize return on investment (ROI).

Using a Kaggle ad click prediction dataset, our approach integrates:

- **Predictive modeling**: Machine learning algorithms (Logistic Regression, SVM, Random Forest, XGBoost, and LightGBM) to estimate the probability of user ad clicks.
- **Personalized ad positioning**: Optimizing ad placement using logistic regression and gradient boosting models, ensuring better engagement.
- **Multi-Armed Bandit (MAB) strategies**: Dynamically adjusting ad placements through an epsilon-greedy reinforcement learning framework to balance exploration and exploitation.

Our findings highlight that XGBoost and LightGBM outperform traditional models, capturing complex user interactions and significantly improving ad placement recommendations. Moreover, MAB-driven adaptive ad selection surpassed a static baseline model, leading to an increase in CTR from 11.26% to 13.76% by dynamically allocating impressions to high-performing positions.

This project demonstrates that adaptive machine learning and reinforcement learning techniques offer superior ad targeting strategies, enabling businesses to reduce inefficient spending, improve engagement, and enhance marketing ROI in real-time digital advertising environments.

1 Project Scope

Digital advertisers struggle to optimize ad placements and bidding strategies in real time, leading to inefficient spending and suboptimal targeting. This project leverages machine learning and reinforcement learning-based Multi-Armed Bandits (MAB) to enhance ad performance.

Using a Kaggle ad click dataset, we:

- Machine learning models as baselines
- Optimize ad placement through personalized recommendations
- Multi-Armed Bandits (MAB) to balance exploration (trying new ad placements) and exploitation (using the best-performing ads)

By comparing RL-based models with traditional baselines, we provide data-driven strategies to maximize Click-Through Rate (CTR) and Return on Investment (ROI).

2 Data

2.1 Data Overview

The dataset used in this project comes from Kaggle's Ad Click Prediction Dataset, which contains structured data on user interactions with online advertisements. It includes features related to user demographics, browsing behavior, ad attributes, and engagement metrics, making it a valuable resource for modeling click-through rate (CTR) prediction and ad optimization.

Key features in the dataset:

- **User Information:** Includes attributes such as device type, browsing history, and demographic indicators
- **Ad Characteristics:** Captures details like ad position, type, and content
- **Time of Day:** Logs when an ad interaction occurred, enabling analysis of engagement patterns at different times
- **Click Behavior:** A binary target variable (clicked: 1 or 0) indicating whether the user interacted with the ad

2.2 Data Preprocessing

To prepare the dataset for modeling, we employed a series of data preprocessing steps:

1. **Duplicate Removal:** Eliminated redundant records to avoid bias in model training. This gave us 4,000 unique observations as compared to the original 10,000.
2. **Handling Missing Values:** Missing values were filled using the most frequent value based on the condition of another column, ensuring context-aware imputation.
3. **Feature Engineering:** Created age buckets.

This preprocessing pipeline enabled the dataset to be ready for logistic regression modeling and other analytical tasks.

3 Methodology

3.1 Predicting Ad Click Probability with Machine Learning

In the competitive digital advertising landscape, maximizing ad effectiveness is critical for improving return on investment (ROI). To help the company better understand and target potential customers, we developed a machine learning-based system that predicts the probability of whether a user will click on an advertisement. By leveraging predictive analytics, we can optimize ad placements, personalize content, and allocate marketing budgets more efficiently.

Our models analyze user characteristics such as gender, device type, browsing behavior, ad position, and time of day to identify the most responsive audience segments. By predicting the likelihood of a user clicking on an ad, the company can prioritize high-probability users, reducing wasted impressions and improving conversion rates.

We trained five models—Logistic Regression, SVM, Random Forest, XGBoost, and LightGBM—to predict ad clicks. To ensure accuracy and fairness, we addressed data imbalances using undersampling and optimized model performance through hyperparameter tuning.

We find out that Gradient Boosting models (XGBoost & LightGBM) performed best, indicating that user behavior is non-linearly correlated with ad engagement. Random Forest showed lower probability scores, suggesting potential issues with probability calibration in tree-based models. SVM and Logistic Regression were helpful but less effective in capturing complex interactions between user features.

3.2 Personalization of Ad Positioning

3.2.1 Preprocess - Loss Reweighting

To address class imbalance in our dataset, we implemented loss reweighting, specifically for the age group feature. Initially, we observed a high imbalance among age groups, which could introduce bias in model training and impact the model's ability to generalize across different user segments. By computing class weights inversely proportional to class frequencies, we ensured that underrepresented age groups were not overshadowed by dominant ones. This approach balanced the model's learning process, ultimately improving prediction performance for all user segments and leading to more personalized ad placement recommendations.

3.2.2 Logistic Regression

The initial logistic regression model served as a strong baseline. The model demonstrated relatively good precision for predicting the bottom and top ad positions. However, it struggled with predicting side ads, showing lower precision and recall, likely due to data imbalance and overlapping feature distributions.

To enhance model interpretability and remove redundant features, we applied L1-based feature selection. By training a logistic regression model with an L1 penalty, we identified and retained the most informative features, discarding less relevant ones. Surprisingly, feature selection did not significantly alter model performance, suggesting that the original feature set was already optimized.

We fine-tuned C, the regularization strength, using Grid Search CV, but observed that performance remained largely unchanged. This outcome suggests that logistic regression reached its performance ceiling due to the linear nature of the model. Despite this, the model still performed well in recommending bottom and top ad positions with reasonable precision, making it a useful benchmark for comparison.

3.2.3 XGBoost

Moving to a more powerful, gradient-boosting model, XGBoost showed immediate improvements. Unlike logistic regression, it captured complex feature interactions, improving precision for all ad positions, particularly for top ads, where precision increased significantly.

Using XGBoost's feature importance scores, we refined the feature set further. Features with an importance score above the median threshold were selected, reducing noise and improving model generalization. This led to a more robust model while maintaining high performance.

Through Grid Search CV, we optimized key parameters such as learning rate, max depth, and number of estimators. The final model exhibited better precision across all ad positions, particularly excelling in predicting bottom and top ads with high confidence. The side ad position still posed a challenge, but overall, the XGBoost model provided a strong and reliable recommendation system with enhanced precision and adaptability.

3.2.4 Final Results and Insights

Our personalized ad placement system successfully utilizes both logistic regression and XGBoost to predict optimal ad positions. While logistic regression provided a simple, interpretable baseline, XGBoost significantly enhanced precision and adaptability. The final model excels in recommending bottom and top ads with high precision, making it an effective system for real-time ad personalization. Further improvements could involve deeper feature engineering and ensemble techniques to enhance predictions for side ads.

3.3 Multi-Armed Bandits (MAB)

Online advertising platforms constantly adapt to shifting user behavior, seasonality, and market competition. Traditional ad placement strategies rely on historical CTR data, but static models fail to adapt when engagement trends shift. To address this, we implemented a reinforcement learning (RL)-based Multi-Armed Bandit (MAB) algorithm that we learned in class to dynamically optimize ad position selection.

We designed a dynamic advertisement market environment for our RL model to learn based on the ad click dataset by incorporating:

- **Seasonality Effects:** CTR fluctuates periodically to reflect variations in user engagement throughout the day and week. We simulate the click probability change with seasonality using periodic trigonometric function.
- **Competitor Influence:** Some ad positions gain or lose effectiveness over time due to changes in competing ads.
- **Random Fluctuations:** We introduce small noise variations to simulate real-world uncertainty in user behavior. For simplicity, we simulated the combined effect of competitor influence and random fluctuations using a carefully rescaled random functions.

To simulate these conditions, we modified the true CTR of each ad position dynamically at each time step. This ensured that our MAB model did not overfit to static trends but instead adapted continuously to real-time feedback.

3.3.1 Epsilon-Greedy Strategy for Ad Placement

Our MAB model employs an **epsilon-greedy** strategy with an exploration rate of $\epsilon = 0.05$. This means that:

- **95% of the time**, the model selects the ad position with the highest observed CTR (exploitation).
- **5% of the time**, the model randomly selects an ad position to explore new possibilities.

By maintaining a small exploration rate, the model avoids being stuck in a suboptimal policy, ensuring that it can adapt if CTR patterns shift over time.

3.3.2 Performance Evaluation and Conclusion

To assess the effectiveness of MAB-driven adaptive ad selection, we compared our approach against a static baseline model that naively always places ads in the “Top” position in the test set. Because in the training dataset, "Top" position has the best CTR statically.

Our results show that the MAB model **outperformed the baseline**, achieving an overall CTR of **13.76%** compared to the baseline CTR of **11.26%**. The model successfully identified that the “Side” ad position was the most effective, allocating **9,236 trials** to it, resulting in a CTR of **13.48%**. Additionally, the model allocated fewer trials to the “Bottom” and “Top” positions as it learned that they were less effective.

In conclusion, the MAB approach demonstrates the power of reinforcement learning in real-time ad selection. Unlike static models that rely on past CTRs, the MAB model continuously learns from user interactions, allowing it to adapt to changing engagement trends, reduce inefficient spending, and improve overall ad performance. The results validate that adaptive strategies outperform traditional rule-based models in dynamic advertising environments.

4 Appendix

4.1 Predicting Ad Click Probability with Machine Learning

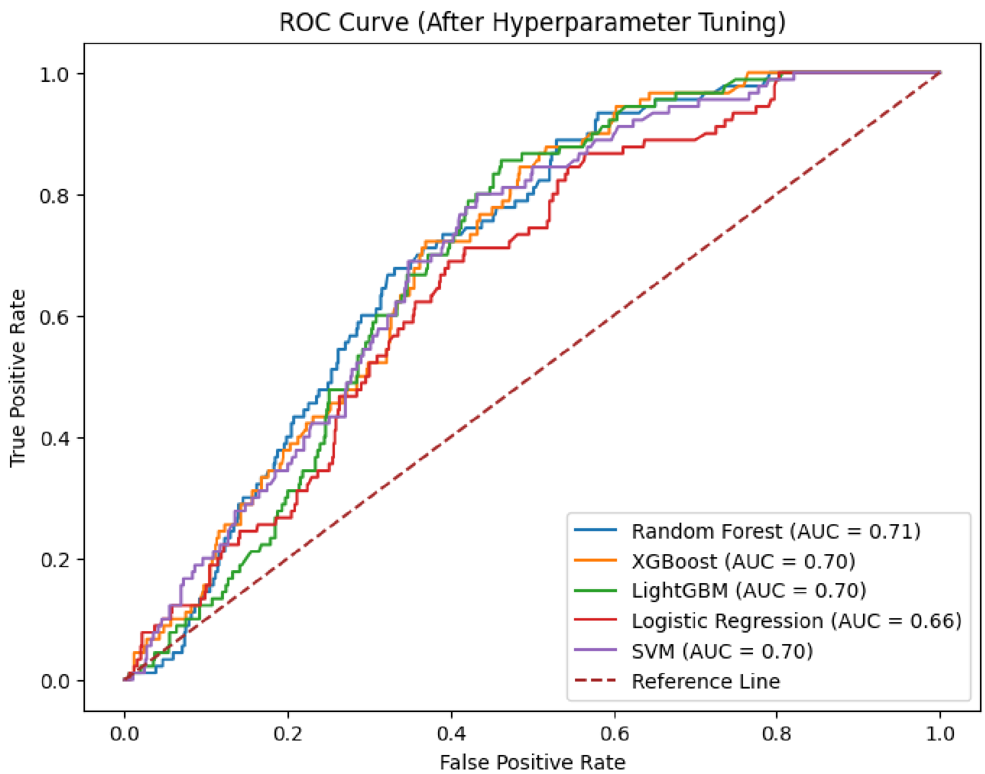

Figure 4.1: Logistics Regression - Baseline Model

4.2 Personalization of Ad Positioning

```
Classification Report:
              precision    recall  f1-score   support

      Bottom       0.56      0.68      0.61       286
        Side       0.36      0.18      0.24       222
         Top       0.52      0.61      0.56       292

    accuracy                           0.51       800
   macro avg       0.48      0.49      0.47       800
weighted avg       0.49      0.51      0.49       800
```

Figure 4.2: Logistics Regression - Baseline Model

	precision	recall	f1-score	support
Bottom	0.558739	0.681818	0.614173	286.000
Side	0.364486	0.175676	0.237082	222.000
Top	0.517442	0.609589	0.559748	292.000
accuracy	0.515000	0.515000	0.515000	0.515
macro avg	0.480222	0.489028	0.470335	800.000
weighted avg	0.489760	0.515000	0.489665	800.000

Figure 4.3: Logistics Regression Model After Feature Selection and Hyperparameter Tuning

	precision	recall	f1-score	support
Bottom	0.560000	0.685315	0.616352	286.00
Side	0.413043	0.085586	0.141791	222.00
Top	0.497525	0.688356	0.577586	292.00
accuracy	0.520000	0.520000	0.520000	0.52
macro avg	0.490189	0.486419	0.445243	800.00
weighted avg	0.496416	0.520000	0.470512	800.00

Figure 4.4: XGBoost - Baseline Model

	precision	recall	f1-score	support
Bottom	0.585443	0.702190	0.614618	286.0000
Side	0.417647	0.121622	0.175896	222.0000
Top	0.536780	0.690765	0.561505	292.0000
accuracy	0.527500	0.527500	0.507500	0.5075
macro avg	0.463102	0.477619	0.450673	800.0000
weighted avg	0.474912	0.507500	0.473486	800.0000

Figure 4.5: XGBoost Model After Feature Selection and Hyperparameter Tuning

4.3 Multi-Armed Bandits (MAB)

```
Baseline Model CTR (always Top): 0.1138
MAB Model Performance:
   ad_position  wins  trials       CTR
0       Bottom    71     543  0.130755
1         Side  1245    9236  0.134799
2          Top    27     221  0.122172
```

Figure 4.6: Comparison of MAB and baseline performance. The MAB model achieved a higher CTR by dynamically adjusting ad placements based on real-time feedback.

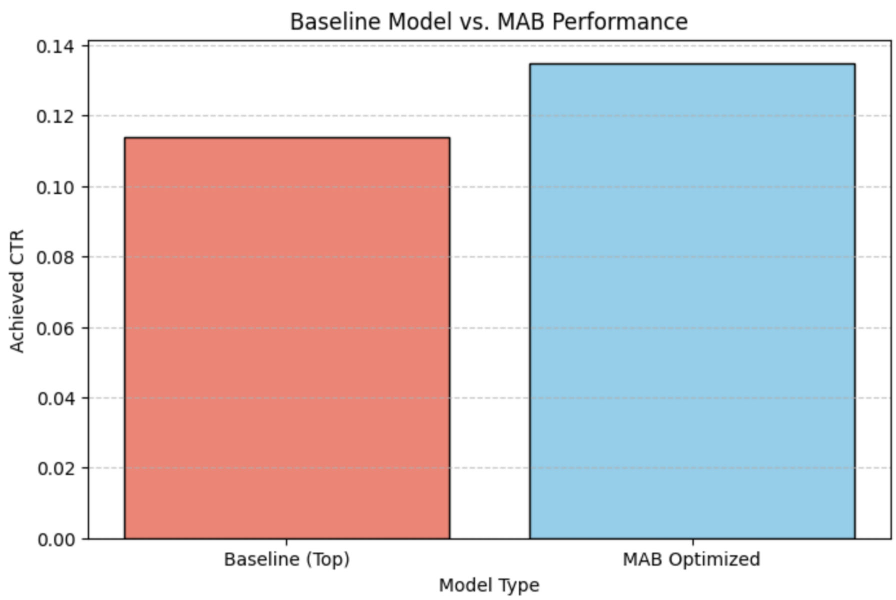

Figure 4.7: MAB model performance comparison. The model dynamically adjusted ad placements, achieving a higher CTR than the baseline approach.