Data Science in Online Advertising – a study on Click-through Rate

Yihui Zhou

013799500

San Jose State University

Yihui Zhou – MS in Data Analytics, San Jose State University

1 Washington Sq, San Jose, CA 95192

Contents

[Introduction 3](#_Toc6857626)

[Purpose 5](#_Toc6857627)

[Literature Review 6](#_Toc6857628)

[Research Design and Methods 10](#_Toc6857629)

[Results 11](#_Toc6857630)

[Discussions and Insights 12](#_Toc6857631)

[Recommendations 13](#_Toc6857632)

[References 13](#_Toc6857633)

# Introduction

**Brief Introduction**

Advertising is bringing fundamental changes to the Internet. It is easy to insert advertising to websites thus many companies have adopted strategies to use ad-based monetization on their website and subsequently use ad revenue to subsidize low-priced products or free products. Many companies make great products free sponsored by advertisement, and few of them make a big name out of it, for example, Google.

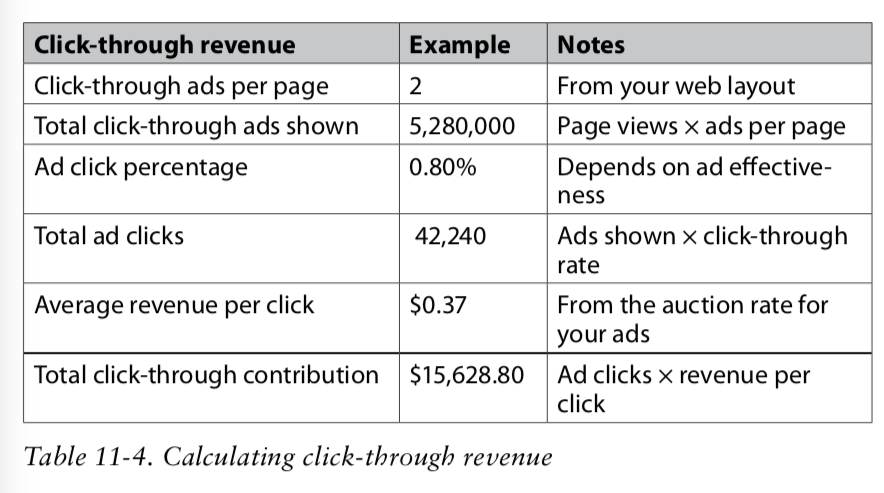
There are clear traits of successful content generating companies: they have competitive contents; they have ability to attract repeated visitors who are increasingly likely to spend longer time on their websites. Many content generating companies measure advertising revenue through click-through rates or display rates. Other than that, content generating companies also need to watch out for “inventory” such as time visitors spent on the website, number of repeated visitors, and number of unique visitors etc.

There are four major revenue models in the industry. First type is sponsorship. The content company enters into a contract with sponsor to display banner on website for certain period of time. Second type is display advertising, which refers to that content company will get paid for certain amount for certain amount of views of the display ads. The third type is click-based revenue. The content company is paid by the numbers of clicks on the ads multiplied by the times the ad is seen. In this method, the content company is paid by ad clicks instead of ad views. The fourth type is affiliate model where the content company could get paid when a transaction takes place through the ad display.

The paper focuses on the third type of revenue model, based on click-through rate. As click-through rate is so closely related to revenue growth in the Internet industry, it is sometimes referred to as the key to growth. This paper focuses on click through rate prediction, spanning literature review over major prediction models including logistic regression, factorization machine, gradient boosting decision tree to deep learning models.

**Basics of click-through rate**

This paper focuses on the third type of model revenue model: click-based revenue. Lean Analytics has a very example to walk through click-through rate from a revenue generation perspective.



(Pic1: Calculating click-through revenue, Lean Analytics)

In this example, the click through rate is 0.80% which determined the total number of ad views to be 42,240; then given the revenue per click to be $0.37, revenue generated from click-through is $15,628. Click-through revenue is thus dependent on the percentage of visitors who view the ad (measured by click-through rate) and the amount paid for each view. (Croll & Yoskovitz, 2013)

Global search marketing agency Covario once conducted a report showing that the average click-through rate for search companies is around 2%. Even well designed and brilliant ads could hardly get a click-through rate higher than 5%.

In many cases, click-through rate is measured by the number of clicks of an ad divided by the number of impressions or total views of the ad. This form of click through rate is also the most discussed and comparison format This paper uses the same method to discuss and measure click-through rate on the aggregate level.

On individual view level, which means for each time a user views an ad, there are two possible outcomes: the viewer is going to click on the ad, or the viewer is not going to click on the ad. So, on individual level, click through rate prediction is a binary classification problem. This paper later examines literature review on various machine learning models to summarize previous approaches to predicting click-through rate.

**Terminology**

Some terminology frequently used in the paper are listed here. Explanation are quoted from Lean Analytics. (Croll & Yoskovitz, 2013)

Audience and churn: how many people visit the site and how loyal they are

Ad inventory: the number of impress: the number of impressions that can be monetized

Ad rates: sometimes measured in cost per engagement – essentially how much a site can make from those impressions based on the content it covers and the people who visit

Click-through rates: how many of the impressions actually turn into money

Content/advertising balance: the balance of ad inventory rates and content that maximizes overall performance.

**Tools and Skills**

This paper develops code written in Python, SQL and Spark. The running environment for code is on Spark powered by Databricks community edition. Packages involved in the code are from Spark machine learning library. For data exploratory purposes on sampled data, the paper uses jupyter notebook and python packages to find data patterns.

Skills involved in conducting research include data collection, exploratory data analysis, ETL, feature engineering, statistical modeling, machine learning modeling, and model evaluations. Furthermore, the paper aims to extract insights that have business impacts on the companies that need skills in business acumen, data visualization, presentation and persuasion.

# Purpose

**Problem details**

Click-through rate is a very important measure to advertising revenue, which is a major revenue income for many content generating companies. How to predict whether an ad is going to be clicked or not is very important for both content generating companies and sponsor companies.

**Rationale**

There are two perspectives to the problem. One angle is from content generating companies. An optimal approach for content generating companies, for example a search engine, is to choose an ad based on expected value, which means the price of a click times the likelihood that the ad will be clicked. A $1.0 ad with a 5% probability of being clicked has an expected value of $0.05 whereas a $2.0 ad with a 1% probability of being clicked has an expected value of only $0.02. In this case, the content generating would choose to display the first ad. Thus estimating the likelihood that a given ad will be clicked is very critical to content generating companies.

Another angle is from sponsor companies, the companies that pay to have their ads. Click-through rates show up as a top metric for performance measurement when those companies are designing their ads through A/B Testing.

**Problems addressed**

The paper aims to understand features impacting whether users are going to click on certain ad or not, which could provide insights to online advertising.

**Research methods and sources**

Sources in this paper come from three areas: first, online research paper by scholars in the field; second, github.com repositories to check out sample code realizations; third, Kaggle competition for company Avazu on click-through rate provides the dataset and general introduction.

# Literature Review

**Model evolvement introduction**

Click-through rate modeling is a core modeling practice in Internet industry. A quick review on the model evolvement could provide insights on how drastic and fast this industry is progressing. Before 2010, models are based upon Logistic Regression; after that models evolve into great diversity from Factorization Machine, Gradient Boosting Decision Tree, all the way till deep learning after 2015.

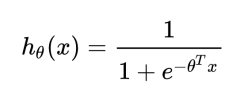
The paper divides models into two parts: traditional click-through rate models (Logistic Regression, Factorization Machine, Gradient Boosting Decision Tree etc) and deep learning click-through rate models. Traditional models still play an important part in current day modeling due to high clarity, high explainability, less heavy training requirement, and possibility to move to online study. Moreover, many deep learning models are based on traditional click-through rate models; for example, the influential FNN, DeepFM, NFM deep learning models are derived from FM model.

The first part of the literature review will focus on traditional CTR models. The second part will touch upon deep learning models. In traditional models, the paper proposes that there are four major directions of models all based upon logistic regression.

**Logistic Regression**

Logistic regression is a common form of general linear models. Its assumption is that given x, y conforms to a Bernoulli distribution. Thus logistic regression model resonates with the nature of the click-through problem which is binary classification.

The formula for logistic regression is:



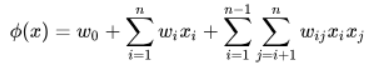
In the formula, x refers to the features, θ refers to parameters the model needs to learn. There is a linear combination of x and θ put in sigmoid function to output probability of y, or h(x) – the probability of whether the user is going to click the ad or not.

The model is intuitive in human understanding to have combinations of features and coefficients, and then fit output between (0,1) to show probability. Thus the model is easy to understand and explain to colleagues across multiple functions and easy to implement, and run in company environments whether it be distributing systems or local. In the industry, a common practice is to use logistic regression first and as a benchmark and move to other models only if they have potential to beat logistic regression.

**Factorization Machine**

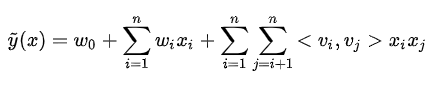
Logistic regression has its limitations. A noticeable one is Simpson Paradox where a trend appears in several different groups of data but disappear or reverses on the combined level. Many features might have non-linear relationships and might interfere with each other so by simply combing them all independently could lead to problems in Logistic Regression. (Rendle, 2010)

One step forward is Degree-2 Polynomial Margin (Poly2):

****

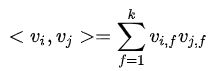
where w refers to weight of feature pairs. However this model adds to the sparsity of features, and subsequently demands more space and computation powered needed, and increases chances of overfitting with increased number of features with the same amount of data so is not preferred.

Factorization Machine model comes in to solve the issue. The formula is:



Parameters to estimate are:





where (vi,vj) is the inner product of two k-dimensional vectors.

Rendle in paper has introduced the model as a new model that combines the advantages of Support Vector Machines (SVM) with factorization models. Factorization Machine models all interactions between variables using factorized parameters. By limiting the number of k, the model can reduce the number of variables from n^2 in Poly2 to k\*n variables, reducing computational complexity tremendously. The model can also learn hidden relationships between two remotely connected variables. This is very similar to the Matrix Factorization, or Alternative Least Squares model, widely applied in recommendation to learn k latent factors to bring down huge calculation required by sparse matrixes. (Rendle, 2010)

Factorization Machine became very popular around 2012-2014 and was widely adopted in industry.

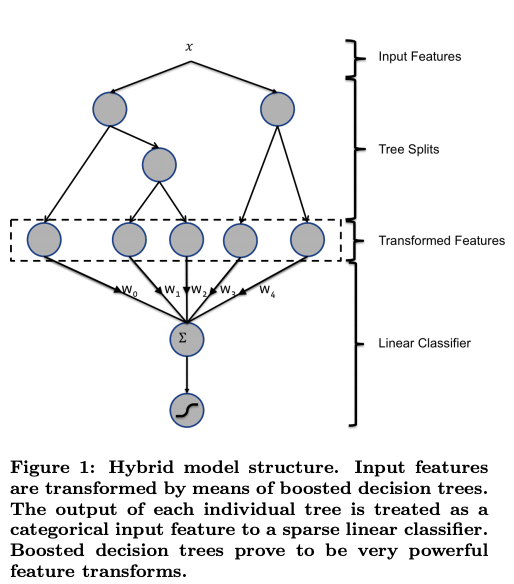
A step forward from FM is FFM, Field-aware Factorization Machine, introduced by Yu chin Juan, Criteo. The formula is summarized as:



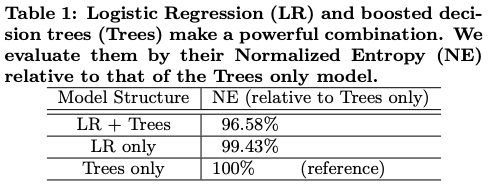
where the model identifies different fields (f) from features and would have different hidden vectors relating to different fields. The model contains more information by introducing the concept of fields but also brings the number of features to n\*k\*f. (Juan, 2016)

**Gradient Boosting Decision Tree + Logistic Regression**

In 2014, Facebook team introduced hybrid model to combine Gradient Boosting Decision Tree and Logistic Regression. GBDT is used to learn features and output new feature vectors, which are then inputted to logistic regression model to analyze and output predictions. A clear explanation could be seen from the chart below. (He, et al, 2014)



According to Facebook, the model has reduced 3% Normalized Entropy compared to LR only or Tree only models. (He, et al, 2014)



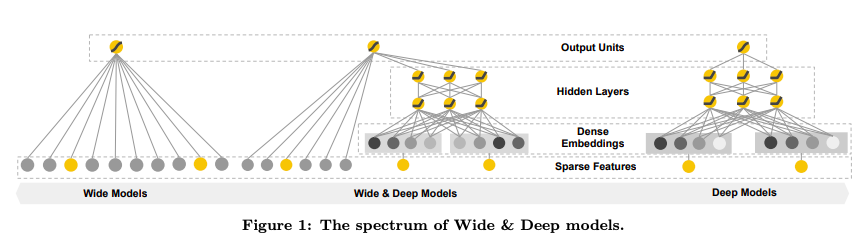
**Deep Learning**

The second part of the literature review focuses on several main stream deep learning methods.

Deep learning models start from Deep Neural Network (DNN). From its structure could be known as multiple layer perceptron MLP. Many later deep learning models focus on various embedding methods to transform high dimensional sparse feature to DNN to low dimensional dense feature.

Factorization-machine supported Neural Networks (FNN)use field (f) to categorize features to bring the dimensionality of input features. Product-based Neural Network (PNN) adds a product layer between embedding layer and hidden layers to show not only individual features but also the product of features.

Above two are “deep” models that add to embedding layer for model input however they lack “wide” in terms of generating features. Google, in 2016, proposed a Wide & Deep model and applied in Google Play App recommendations which performed well. (Cheng, et al, 2016)



The model brought by Google has quickly been adopted in industry for its advantage in combining both wide models (logistic regression), which is good at memorization-relating label to high frequent features given small amount and deep models (DNN), which is good at generalization-being applied to wider datasets.

Many companies improved based on Wide & Deep model and generated subsequent models like DeepFM model, Neural Factorization Machine, Deep& Cross Network etc.

# Research Design and Methods

**Data Collection**

Data source is from Kaggle.com, sponsored by Avazu company.

There are two datasets: training set and test set. Training set is 10 days of click-through data, ordered chronologically. Non-clicks and clicks are subsampled according to different strategies. Test set is 1 day of ads for testing model predictions.

The training dataset contains 40million rows of user impressions in total.

Data field includes the following categories: id (ad identifier), click(0/1 for non-click/click), hour, C1 (anonymized), banner\_pos, site\_id, site\_domain, site\_category, app\_id,app\_domain, app\_category,device\_id, device\_ip,device\_model, device\_type, device\_conn\_type, and C14-C21(anonymized). (Kaggle,2014)

**Data Analysis**

The paper divides data analysis into two parts. The first part is to sample 100k data and run on local machine for EDA purposes. The second part is to move dataset to Spark, powered by Databricks, to examine a larger amount of data and apply machine learning models.

In the first part, the paper examined available features in the dataset. There are 5 types features: the target feature: click or not; site features: site\_id, site\_domain, site\_category; app features: app\_id, app\_domain, app\_category; device feature: device\_id, device\_ip, device\_model,device\_type, device\_conn\_type; anonymized features C1, C14-C21.

The target features shows that around 17% impressions are clicked and 83% are not. Based on prior knowledge on click-through rate of less than 2%, the training dataset has already resampled to address imbalanced dataset issue.

Then the paper applied feature engineering to date time features by extracting hour and assigning day of week information. A closer examination on distributions of click-through rate (clicks/total impressions), number of clicks, and number of impressions show that midnight and late afternoon tend to show higher click through rates, whereas midnight shows fewer amount of clicks and impressions. Weekends have higher click through rates but Tuesday and Wednesday show larger amount of actual clicks.

Banner positions, which means where the ads are displayed have an impact on distribution as well as C1 feature, which is anonymized in this case, and device type.

The second part is moved on to Spark platform by Databricks for a dataset of 4 million records. Feature engineering includes designing a pipeline to incorporate all stages including Quantile Discretizer, String Indexer, and Vector Assembler to prepare features. Four models are applied in the second part including Logistic Regression, Decision Tree, Random Forest and Gradient Boosting Tree Classifier. The best performing model is from GBTC which achieved Area under Curve (AUC) of 72%. A breakdown on feature weight shows that the anonymized C17 has the largest weight of 37%.

**Interpretations**

The Gradient Boosting Tree method has returned the highest AUC of all four models, which is within expectation because the boosting method is a sequential method aiming at lowering bias.

The Random Forest model as an ensemble model, provided the second-best AUC in all models.

**Obstacles**

There are three obstacles for the paper. First is on dealing with features. Many of the features are categorical features and some of them are anonymized, which makes EDA less intuitive and harder to explain. Second is on the large amount of data. The paper decided to move the second part of analysis to Spark for its distributed system for parallel processing, which involves Spark RDD, Spark Dataframe and Spark SQL for ETL. Even on Databricks because the paper uses a community version, a limit on the unique values in categorical feature was placed. The third challenge is that some of the classic machine learning models do not have packages on Spark MLlib.

# Results

**Anticipated results**

The research anticipated results in comparing four Spark machine learning models to predict click through rate as a binary classification problem. Boosting model showed the best result followed by Bagging model, another ensemble model.

The best AUC of around 72% is within expectation given limitations in the dataset and Spark platform.

**Impacts and improvement**

Impacts from the research include two aspects: first is that research confirmed that given platform, user, and ad data it is possible to predict click through rate for different users; second the research pointed out possible future directions to personalization by digging deeper in the most important features.

**Visualization tools**

Research was conducted on two platforms thus two sets of visualization tools are used here: one is Python packages, including Matplotlib and Seaborn which are realized in Jupyter Notebook; the other is imbedded plotting options on Spark by Databricks.

# Discussions and Insights

**Significance**

Click through rate plays a key role in advertising business in three aspects. First aspect is for the sponsor companies that design the ads. Companies as such need to design multiple versions of ads to increase its appeal to targeted audience. Second aspect is for content generating companies that need to display or position the ads and charge cost per click accordingly. For content companies, they need to design data strategies and maximize the revenue generated from displaying ads at the right place to the right audience. Third aspect is for targeted audience. In order to increase click through rate, many companies in the marketing field are designing strategies for personalization and segmentation.

**Insights**

Insights from the project could be drawn in the three aspects similar to its significance.

First, on the user level, given the explosion of digital information, targeting right customers need to be moved earlier to initial search stage where much of customer’s information is anonymous. This echoes with the anonymity in the research dataset. Later when customers indeed sign up and provide personal information to company, machine learning strategies could be used to help on personalization and segmentation on users, to create more targeted and efficient advertising strategy. Personalization strategy is essentially a data strategy which requires companies to incorporate more information beyond traditional marketing segmentation strategy based on name, location, demographics but purchasing behavior, lifestyle, and social media histories as well. This echoes with research findings that features related to user behaviors have larger feature importance, for example app\_domain, app\_category and hour features.

Second, on the sponsor company level, there is work to be done for personalization. Given the growing number of features that could be used in machine learning, sponsor companies are under pressure to undergo large amount of experiments. Many companies in the industry in order to face up this challenge have adopted new strategies to include multi arm bandit testing together with traditional A/B testing methods.

Third, on the platform level, content generating companies need to have data strategy to maximize advertising revenue by conducting efficient personalization, according to Arm Treasure Data, which is not to personalize every touchpoint for every person at all times, but to personalize the experience on the channels each customer prefers at the time when they’re ready.

**Tools and skill sets**

Tools in data collection, data manipulation, feature engineering and modeling are very important in building click-through rate project. Common tools include Python and SQL. Tools in large scale data processing is needed too to handle large amount of data, including Hadoop and Spark. Machine learning tools including Sklearn, Spark MLlib, Tensorflow/Keras could be applied as well.

# Recommendations

**New trends**

There are two dominant trends in the field. One is that an increasing amount of companies are adopting deep learning models to predict click through rate. Second is that, while click through rate prediction is one of the most critical inputs used, there are more actions and behaviors to be monitored and analyzed between users and websites for measuring advertiser and user satisfaction.

**Proposals for future study**

In future study, below three areas could be further researched on. First, deep learning models could be included in research to compare performance with traditional machine learning models. Second, more clarify on dataset with full information on the features including device, app, site, even user information could provide more insights to the subject. Third, larger capacity distributed system could be used to resolve limitations on the number of unique categorical features. For the next two quarters, the research could be carried on following above three directions to provide deeper analysis and more comprehensive overview of the click through rate prediction problem.

# References

Croll A., Yoskovitz B. (2013). *Lean Analytics Use Data to Build a Better Startup Faster.* O’Reilly. 114-115, 117-118.

Rendle,S. (2010). *Factorization Machines.* Retrieved from

<https://www.csie.ntu.edu.tw/~b97053/paper/Rendle2010FM.pdf>

Juan, Y.C. et al. (2016). *Field-aware Factorization Machines for CTR Prediction.* Retrieved from

<https://www.csie.ntu.edu.tw/~cjlin/papers/ffm.pdf>

He, X.R. et al. (2014). *Practical Lessons from Predicting Clicks on Ads at Facebook.* Retrieved from

<https://quinonero.net/Publications/predicting-clicks-facebook.pdf>

Cheng, H.T. et al. (2016). *Wide & Deep Learning for Recommender Systems.* Retrieved from

<https://arxiv.org/pdf/1606.07792.pdf>

Li, S. (2018). *Mobile Ads Click-Through Rate (CTR) Prediction, Online Advertising, Google PPC, AdWords Campaign, Mobile Ads.* Retrieved from

<https://towardsdatascience.com/mobile-ads-click-through-rate-ctr-prediction-44fdac40c6ff>

Kaggle (2015).