

[Log In](#)

[Join](#)

[Back To Module Home](#)

Machine Learning System Design

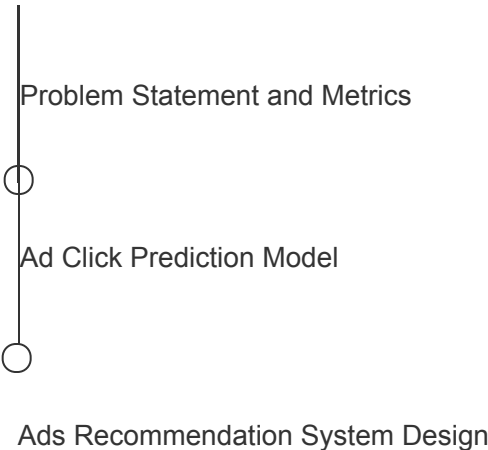
0% completed

Machine Learning Primer

Video Recommendation

Feed Ranking

Ad Click Prediction



Rental Search Ranking

Estimate Food Delivery Time

Machine Learning Knowledge

Machine Learning Model Diagnosis

Conclusion

Mark Module as Completed

Problem Statement and Metrics

Learn about the problem statement and metrics for building an Ad click prediction machine learning system.

We'll cover the following

- Ad click prediction
 - 1. Problem statement
 - 2. Metrics design and requirements
 - Metrics
 - Offline metrics
 - Online metrics
 - Requirements
 - Training
 - Inference
 - Summary

Ad click prediction#

1. Problem statement#

Build a machine learning model to predict if an ad will be clicked.

For the sake of simplicity, we will not focus on the cascade of classifiers that is commonly used in AdTech.

Ads recommendation system

- Let's understand the ad serving background before moving forward. The ad request goes through a waterfall model where publishers try to sell its inventory through direct sales with high CPM (Cost Per Million). If it is unable to do so, the publishers pass the impression to other networks until it is sold.

□

2. Metrics design and requirements#

Metrics#

During the training phase, we can focus on machine learning metrics instead of revenue metrics or CTR metrics. Below are the two metrics:

Offline metrics#

- Normalized Cross-Entropy (NCE): NCE is the predictive **logloss** divided by the

cross-entropy of the background CTR. This way NCE is insensitive to background CTR. This is the NCE formula:

$$NCE = \frac{-\frac{1}{N} \sum_{i=1}^N (\frac{1+y_i}{2} \log(p_i) + \frac{1-y_i}{2} \log(1-p_i))}{-(p \log(p) + (1-p) \log(1-p))}$$

Online metrics#

- **Revenue Lift:** Percentage of revenue changes over a period of time. Upon deployment, a new model is deployed on a small percentage of traffic. The key decision is to balance between percentage traffic and the duration of the A/B testing phase.

Requirements#

Training#

- **Imbalance data:** The Click Through Rate (CTR) is very small in practice (1%-2%), which makes supervised training difficult. We need a way to train the model that can handle highly imbalanced data.
- **Retraining frequency:** The ability to retrain models many times within one day to capture the data distribution shift in the production environment.
- **Train/validation data split:** To simulate a production system, the training data and validation data is partitioned by time.

Inference#

- **Serving:** Low latency (50ms - 100ms) for ad prediction.
- **Latency:** Ad requests go through a waterfall model, therefore, recommendation latency for ML model needs to be fast.
- **Overspent:** If the ad serving model repeatedly serves the same ads, it might end up over-spending the campaign budget and publishers lose money.

Summary#

Type	Desired goals
Metrics	Reasonable normalized cross-entropy and click through rate
Training	Ability to handle imbalance data High throughput with the ability to retrain many times per day
Inference	Latency from 50 to 100ms Ability to control or avoid overspent campaign budget while serving ads

Quiz on loss function
<div>Q</div> <div>For the loss function, why can't we use the logloss function?</div>

<div>Reset Quiz</div> <div>Submit Answer</div>

Back

Feed Ranking System Design

Next

Ad Click Prediction Model

Mark as Completed

Report an Issue