Back To Module Home

Machine Learning System Design

0% completed

Search Module

Machine Learning Primer

Video Recommendation

Feed Ranking

Problem Statement and Metrics

Feed Ranking Model

Feed Ranking System Design

Ad Click Prediction

Rental Search Ranking

Mark Module as Completed

Feed Ranking Model

Learn about the Feed Ranking system architecture and the model requirements.

We'll cover the following

3. Model#

Feature engineering#

Features	Feature engineering	Description

User profile: job title, industry, demographic, etc.	For low cardinality: Use one hot encoding. Higher cardinality: use Embedding.	
Connection strength between users		Represented by the similarity between users. We can also use Embedding for users and measure the distance vector.
Age of activity	Considered as a continuous feature or a binning value depending on the sensitivity of the Click target.	
Activity features	Type of activity, hashtag, media, etc. Use Activity Embedding and measure the similarity between activity and user.	
Cross features	Combine multiple features.	See the example in the Machine Learning System Design Primer. Read about cross features

Training data#

Before building any ML models we need to collect training data. The goal is to collect data across different types of posts, while simultaneously improving user experience. Below are some of the ways we can collect training data:

- Rank by chronicle order: This approach ranks each post in chronological order. Use this approach to collect click/not-click data. The trade-off here is serving bias because of the user's attention on the first few posts. Also, there is a data sparsity problem because different activities, such as job changes, rarely happen compared to other activities on LinkedIn.
- Random serving: This approach ranks post by random order. This may lead to a bad user experience. It also does not help with sparsity, as there is a lack of training data about rare activities.
- Use a Feed Ranking algorithm: This would rank the top feeds. Within the top feeds, you would permute randomly. Then, use the clicks for data collection. This approach provides some randomness and is helpful for models to learn and explore more activities.

Based on this analysis, we will use an algorithm to generate training data so that we can later train a machine learning model.

We can start to use data for training by selecting a period of data: last month, last 6 months, etc. In practice, we want to find a balance between training time and model accuracy. We also downsample the negative data to handle the imbalanced data.

Model#

Selection#

- We can use a probabilistic sparse linear classifier (logistic regression). This is a popular method because of the computation efficiency that allows it to work well with sparse features.
- With the large volume of data, we need to use distributed training: Logistic Regression in Spark or Alternating Direction Method of Multipliers.

- We can also use deep learning in distributed settings. We can start with the fully connected layers with the Sigmoid activation function applied to the final layer. Because the CTR is usually very small (less than 1%), we would need to resample the training data set to make the data less imbalanced. It's important to leave the validation set and test set intact to have accurate estimations about model performance.

Evaluation#

- One approach is to split the data into training data and validation data. Another approach is to replay the evaluation to avoid biased offline evaluation. We use data until time t for training the model. We use test data from time $t+1$ and reorder their ranking based on our model during inference. If there is an accurate click prediction at the correct position, then we record a match. The total match will be considered as total clicks.
- During evaluation we will also evaluate how big our training data set should be, and how frequently we should retrain the model, among many other hyperparameters.

Problem Statement and Metrics

Next

Feed Ranking System Design

Mark as Completed

Report an Issue