ML System Design

Feed Ranking

. Problem Statement.

Design a personalized LinkedIn feed to maximize long-term user engagement. One way to measure engagement is user frequency, i.e., measure the number of engagements per user, but it's very difficult in practice. Another way is to measure the click probability or click through rate (CTR).

On the LinkedIn feed, there are five major activity types:

Category	Example
1. Connection	Member connector follows member/company member joint group
2. Informational	Member or company shares articles/pictures/messages.
3. Profile	Member updates profile, i.e., picture, job-change, etc.
4. Opinion	Member likes or comments on articles, pictures, job-changes, etc.
5. Site-Specific	Member endorses member, etc.

2. Metric design and requirements

Metrics :

Offline metrics

- The Click Through Rate (CTR) for one specific feed is the number of clicks that feed receives, divided by the number of times the feed is shown.

$$CTR = \frac{\text{number of clicks}}{\text{number of times shown}}$$

- Maximizing CTR can be formalized as training a supervised binary classification model. For offline metrics, we normalize cross-entropy and AUC. Normalizing cross entropy (NCE) is the predictive log loss divided by the cross-entropy of the background click through rate. It helps the model be less sensitive to background CTR.

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

Online metrics

- For non-stationary data, ~~online~~ offline metrics are not usually a good indicator of performance. Online metrics need to reflect the level of engagement from users once the model has been deployed, i.e., conversion rate (ratio of clicks with number of feeds).

Requirements :

Training.

- We need to handle large volumes of data during training. Ideally, the models are trained in distributed settings. In social network settings, it's common to have online data distribution shift from offline training data distribution. One way to address this issue is to retrain the models (incrementally) multiple times per day.

- Personalization : Support is needed for a high level of personalization since different users have different tastes and styles for consuming their feed.

- Data freshness : Avoid showing repetitive feed on the user's home feed.

Inference

- Scalability : The volume of user's activities are large and the LinkedIn system needs to handle 300 million users.

- Latency : When a user goes to LinkedIn, there are multiple pipelines and services that will pull data from multiple sources before feeding activities into the ranking model. All of these steps need to be done within 200 ms. As a result, the Feed Ranking needs to return within 50 ms.

- Data freshness : Feed Ranking needs to be fully aware of whether or not a user has already seen any particular activity. Otherwise, seeing repetitive activity will compromise the user experience. Therefore, data pipelines need to run really fast.

Summary:

Type	Desired Goals
Metrics	Reasonable normalized cross-entrop
Training	High throughput with the abilit to retrain many times per day Supports high level of personalizat
Inference	Latency from 100 ms to 200 m Provides a high level of data freshness and avoids showing th same feeds multiple times.

3. Model

Feature Engineering

Features	Feature Engineering
User profile: job title, industry, demographic, etc.	Lower cardinality: one hot encoding Higher cardinality: embedding
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