Problem Statement

Let's look at a problem statement that asks you to design a Twitter feed system.
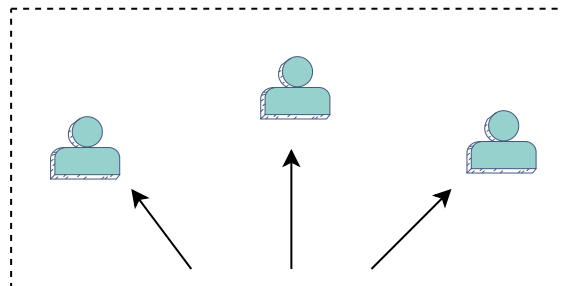
We'll cover the following ^

- Problem statement
- Visualizing the problem
- Scale of the problem

Problem statement#

The interviewer has asked you to **design a Twitter feed system that will show the most relevant tweets for a user based on their social graph.**

How to display the most relevant content for user A's Twitter fe

First, let's develop an understanding of the problem.

Visualizing the problem#

User A is connected to other people/businesses on the Twitter platform. They are interested in knowing the activity of their connections through their feed.
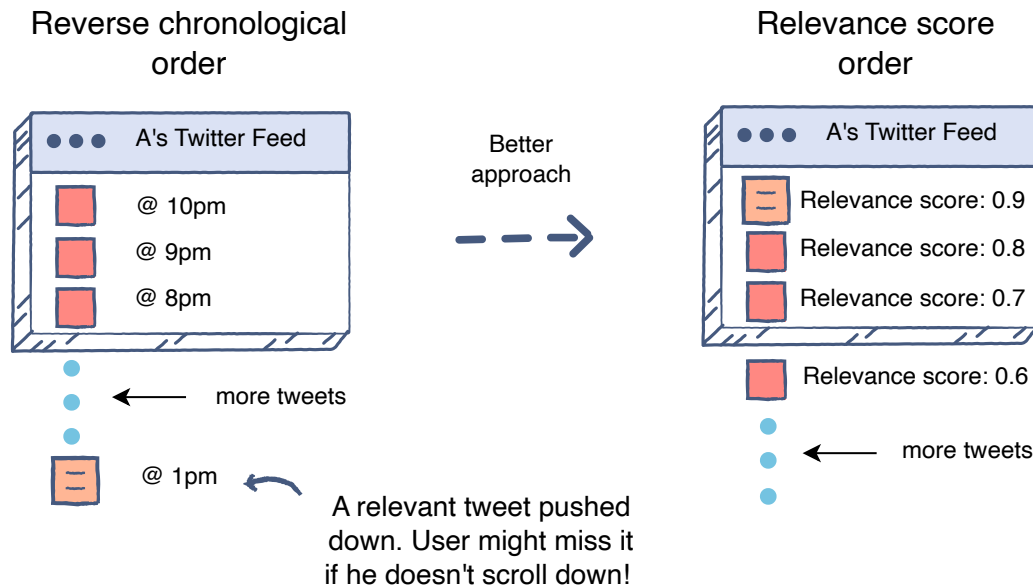
In the past, a rather simplistic approach has been followed for this purpose. All the Tweets generated by their followees since user A's last visit were displayed in **reverse chronological order**.

However, this *reverse-chronological order feed display* often resulted in user A missing out on some Tweets that they would have otherwise found very engaging. Let's see how this happens.

Twitter experiences a large number of daily active users, and as a result, the amount of data generated on Twitter is torrential. Therefore, a potentially engaging Tweet may have gotten pushed further down in the feed because a lot of other Tweets were posted after it.

Hence, to provide a more engaging user experience, it is crucial to rank the most relevant Tweets above the other ones based on user interests and social connections.

Reverse chronological order — Relevance score order

Better approach →

A relevant tweet pushed down. User might miss it if he doesn't scroll down!

Transition from time-based ordering to relevance-based ordering of Twitter feed

The feed can be improved by displaying activity based on its relevance for the logged-in user. Therefore, the feed order is now based on **relevance ranking**.

Scale of the problem#

Now that you know the problem at hand, let's define the scope of the problem:

1. Consider that there are five-hundred million daily active users.
2. On average, every user is connected to one-hundred users.
3. Every user fetches their feed ten times in a day.

Five-hundred million daily active users, each fetching their feed ten times daily, means that your **Tweet ranking system will run five billion times per day**.
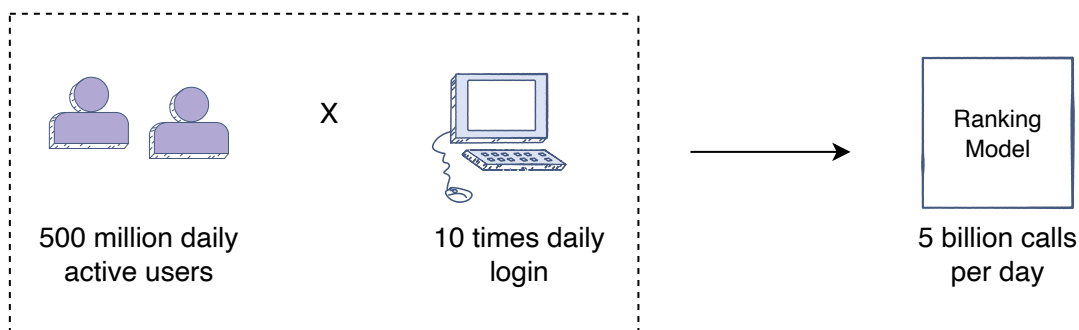
The ranking model may receive as many as five billion calls daily

Finally, let's set up the machine learning problem:

"Given a list of tweets, train an ML model that predicts the probability of engagement of tweets and orders them based on that score"

← **Back**

Filtering Results

Next →

Metrics

✓ Completed

⊘ Report an Issue