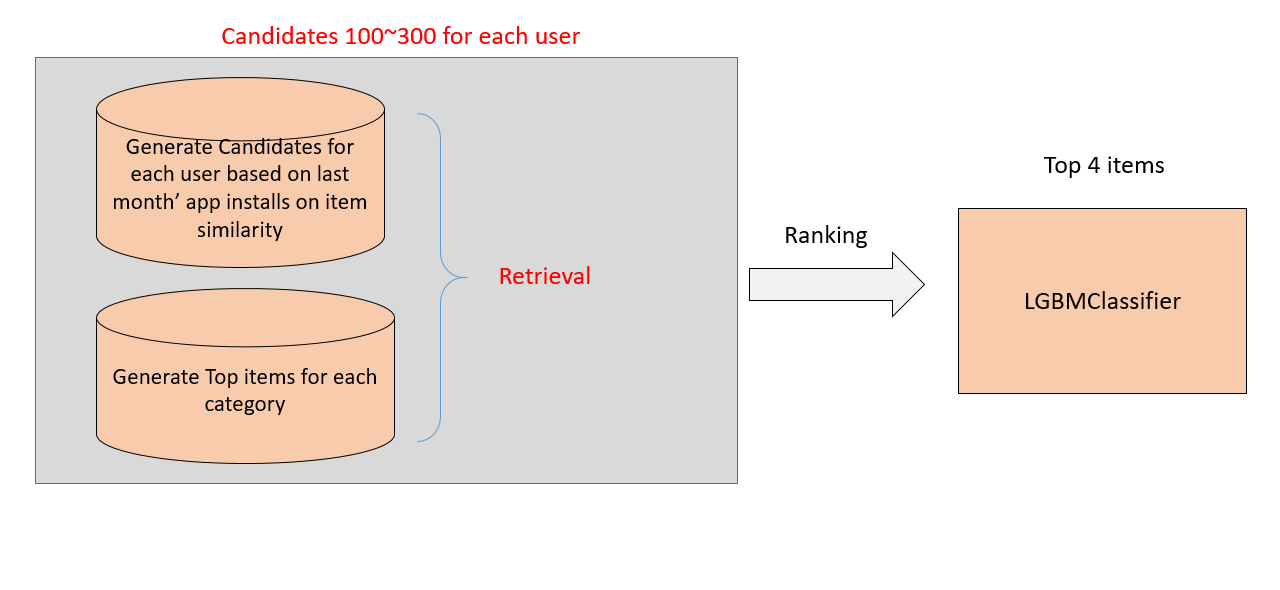
**Approach**:



We would follow the above approach to get the ranking for the users.

Below approach explains more about how the retrieval is done;



**Candidates:**

1) We need to create item similarities for each item and will follow the below approach;

i) Use tfid vectorizer to vectorize the description from app\_metadata

ii) Create an embedding matrix with 512 dimension for tfid vectorizer using SVD.

iii) find the nearest neighbors of each item using a threshold.

2) **Filter** from last month's app install’s of the user as it is found most relevant to the user from experimentation.

3) Check for user's interaction from previous step and map it's matching items.

4) Find out each categories top items from app install only if they are installed.

5) concat step 4 items to user's.

**Model Buliding:**

1) create dataset for train (actual\_set users) and test(validation\_data users).

2) **Score** global average (of time spent) of item from app\_usage.

3) **Score** item average for each user from app\_usage.

4) Mark the target as 1 such that if the item is found in actual set, otherwise mark it as 0.

5) **Rank** for train data for mapk@4 and if the results are good sort the items based on prediction probability.