Assignment: MLUL Group Project Executive Summary

Batch – AMPBA-2021S

Group - 15

12010011	Prashant	Arya	prashant_arya_ampba2021s@isb.edu
12010078	Abhilash	Gadepalli	abhilash_gadepalli_ampba2021s@isb.edu
12010066	Debjit	Ray	Debjit_Ray_AMPBA2021S@isb.edu

Objective

To build a Prediction system that will help suggest users new restaurants. There are multiple ways to make it an additional revenue stream:

- While recommending restaurants, specific targeted advertisements regarding restaurants/ other products the users might like can be shown.
- The restaurants can be provided with set of user ID's who are most likely to like their restaurants to help them run dedicated marketing campaigns.

Project Approach vs Constraints

Building a predictive engine/ model can vary from being very simple to extremely complex.

- A simple predictive engine might be built based on:
    - The restaurants which, has been positively rated maximum number of times recently.
    - The number of times a restaurant has been rated.

  While the above approaches might be simple and quick to implement, these are generic for all users on the platform.
- A more dedicated/ user specific predictive model might try to cluster users of similar characteristics and then try to predict based on the group's inherent behaviour. This approach is bit more complex and costlier to implement and still does not have the user specific predictions.
- The most complex and dedicated predictive models are the ones which treat each user separately. They take into account the users' past activities, likes/ dislikes, users' attributes, products' attributes, psycho-analysis, etc. etc. with no limit to this list. Obviously, this would be the most complex engine to build and both from time and cost perspective can become a major constraint.

However, predictive engines are now becoming more of a necessary condition for any platform rather than being a sufficient condition. So, each and every platform needs to have the predictive engine in some form or other.

Our Recommendations

Based upon our model building experience, no single model can fit all the sizes and use-cases. Depending upon the time to implement, cost of resources and complexity, our recommendation would be in the following order (decreasing order of recommendation):

Model	Pros	Cons
Cosine Similarity	• Considers users' individual characteristics. • Can come up with appropriate suggestions even for users, who are entirely new to the system.	• Resource Intensive – Calculating the cosine similarity against each user on the platform will require lot of Computation power. • The attributes used to calculate the similarity might lend to add unnecessary biasness. For e.g., a person X being recommended a restaurant liked by person Y just because both are smokers will not be correct. • Need more tests to see the accuracy of the predictions.
NMF	• Breaks down the user preferences into latent features and groups users based on these features. • Mathematically proven and already tested model.	• Resource Intensive – Depending upon the total number of users and items, can be very resource intensive. • The latent features on the basis of which groupings are made can be difficult to interpret.
Collaboration based Filtering	• Easy to implement	• Performs well if we have a significant amount of past data.