Unsupervised Machine Learning and Data Mining (DS 5230)
Spring 2022- Project Proposal

Yelp Business Data Analysis and Recommender System

Group Members:
Omkar Pradhan, Yashvi Bhandari

I. Introduction

Customer reviews have a huge impact on business growth. According to Forbes, 94 percent of consumers avoid firms with bad reviews. Negative reviews can have numerous detrimental effects on the business such as loss of revenue, damage to reputation, low search engine ranking, etc. Thus, it is imperative that businesses continually improve their service based on the reviews. However, given the large number of reviews, it can be difficult to extract actionable insights from them. We aim to solve this problem by providing an unsupervised strategy to assist organizations in better understanding the issues they should address. Businesses like Yelp can potentially increase their revenue by recommending specific services to the users based on their past. Users can't know every business. Thus, it is time-consuming and many times frustrating to search for a specific business of interest, amongst hundreds of available options. Therefore, we developed a recommender system that can aid consumers to get the desired service based on their previous activity. Overall, we made an integrated platform that benefits both users and businesses in the following way:

I)Individual business owners can improve their service based on past customer behavior (reviews, ratings, like/dislike, etc.)

II)Individual customers can benefit by getting personalized recommendations based on their past activities.

The main contribution of this work are as follows:

I) We developed a Topic model to summarize restaurant reviews into broad topics, which can be used by restaurant owners to improve their service

II) We developed a recommender system to recommend relevant restaurants to users based on their previous activity. We used 3 approaches: a) Content based filtering b) Item-Item collaborative filtering and c) Neural Collaborative filtering.

III) We demonstrate the effectiveness of deep learning-based collaborative filtering (Neural Collaborative Filtering in specific) compared to traditional matrix factorization-based collaborative filtering.

II. Related Work and Research Gap

Collaborative filtering is widely used for recommender systems due to its proven success in personalized recommendations and is used by various companies like Netflix, Amazon, Google, etc. Recently, there has been a surge in using matrix factorization-based approaches for recommendations inspired by its success in the famous Netflix Prize challenge.

However, collaborative filtering based on matrix factorization uses a simple inner product of the user-item matrix to estimate user-item interactions in low dimensional latent space, which may not be the case in practical settings due to its linearity assumption. We may increase the value of

latent factors to address this issue, but that may adversely affect model generalization (for example, overfitting, etc.), especially in sparse settings. Therefore, in this work, we address this limitation by using Neural Collaborative Filtering (NCF), which replaces the user-item inner product in matrix factorization by a neural network.

III. Proposed Approach

We will now give a brief introduction to various approaches used in this Project.

Topic Modeling

Topic modeling is a method for ***unsupervised*** classification of documents, which finds some natural groups of topics by recognizing the word and phrase patterns within them. We're simply using this technique to group different reviews together into a set defining the most occurring problems and get insights on what is going wrong or what can be improved further. The goal is to use the topic modeling to help restaurant owners understand their consumers' sentiments about their establishments and suggest potential areas for improvement (e.g., ambiance, parking, service, wait time etc.). We have used the following two popular topic modeling approaches in this work.

LDA: The purpose of LDA is to learn the representation of a fixed number of topics and given this number of topics learn the topic distribution that each document in a collection of documents has.

LSA: The core idea behind LSA is to take a matrix of what we have — documents and terms — and decompose it into a separate document-topic matrix and a topic-term matrix.

Content-based recommender System

In this section, we formally define the working of a content-based recommender system that we build for this project. The main idea of the content-based recommender system is to recommend restaurants to the user which are like the restaurants which were highly rated by the users previously. Once we have an item-feature matrix, we next use KNN to recommend similar restaurants to users. KNN calculates the distance between each pair of restaurants, ranks them, and returns the K nearest neighbor restaurants as a recommendation to the user.

Item-Item Collaborative filtering

Collaborative filtering is based on identifying members in a community who share similar tastes. When two users have almost the same rated items, their tastes are comparable. Such users form a community, or "neighborhood." There are two main approaches to collaborative filtering: Item-based and user-based. We build item-based collaborative filtering for this work.

The entry A_{ij} of the matrix corresponds to the rating given by user i to item j. Next, we use Singular Value Decomposition (SVD) to factorize this matrix into user-feature and item-feature matrices. SVD is a technique for breaking down a matrix into three smaller matrices U, V, and S

$$A = USV^T$$

U is an orthogonal left singular matrix that depicts the relationship between users and latent factors while V is a right singular matrix that reflects the similarity between items and latent factors. S is a square diagonal matrix that describes the strength of each latent component.

Neural Collaborative filtering (NCF)

NCF replaces the user-item inner product in matrix factorization with a neural network [6]. Thus, it can be thought of as a nonlinear generalization of traditional matrix

factorization approaches [7]. A schematic diagram of a deep learning-based recommender system is presented below.

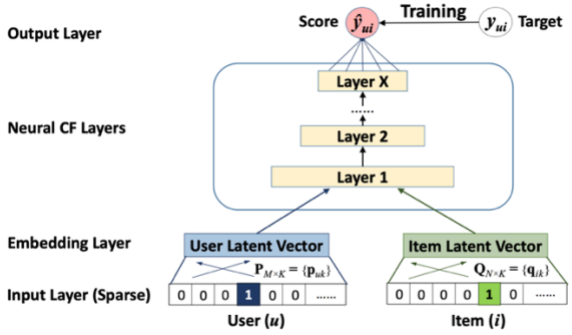

Fig. 1 – Neural Collaborative Filtering Framework [6]

Let p_u and q_i denotes latent vector for user u and item I respectively. Then Matrix factorization estimates $\hat{y}_{ui}$ as follows

$$\hat{y}_{ui} = f(u, i|p_u, q_i) = \boldsymbol{p_u^T q_i}$$

NCF modifies the estimate $\hat{y}_{ui}$ in the following way:

$$\hat{y}_{ui} = f\left(P^T v_u^U, Q^T v_i^I \middle| \boldsymbol{P}, \boldsymbol{Q}, \Theta_f\right)$$

P: latent factor matrix for users
Q: the latent factor matrix for items
Θ_f: denotes model parameters of interaction function
f: multi-layer neural network

As f is MLP network, it can be formulated as

$$f\left(\boldsymbol{P^T v_u^U}, \boldsymbol{Q^T v_i^I} \middle| \boldsymbol{P}, \boldsymbol{Q}, \Theta_f\right)$$
$$= \emptyset_{out}(\ldots\ldots(\emptyset_2(\emptyset_1(\boldsymbol{P^T v_u^U}, \boldsymbol{Q^T v_i^I}))\ldots\ldots))$$

$\emptyset_{out}$ and $\emptyset_x$ denotes mapping function for output and x th neural CF layer respectively.

IV. Experiments

We have chosen an all-purpose learning dataset provided by Yelp Inc. This dataset is a subset of Yelp's businesses, reviews, and user data which has a rich variety of ratings, comments, and metadata of businesses. The description of the JSON files that we have used in our project are as follows.

1.Business.json: Contains business data including the business id, location, stars, attributes, and categories. The attributes consist of hours, parking, availability, ambiance, Wi-Fi availability, and much more.

2.Review.json: Contains review text data including the user id that wrote the review and the business id the review is written for. It contains around 7M reviews for 180K+ businesses for 11 metropolitan areas.

3.User.json: Contains user data including the user's mapping and all the metadata associated with the user

We started with the exploration of our datasets and identified 'Restaurants' business with the greatest number of reviews. Further, we analyzed the number of restaurants in various US cities and found that restaurants based in Boston had the third highest number of reviews. We found it interesting to work on Boston restaurant data, since it is relevant to us. Hence, for further analysis of methods we used in this project, we limit the scope to analyzing only Boston based restaurants.

We first started with classifying the reviews as positive, if the review ratings were 4 or 5 and classified them as negative if the review ratings were 1, 2 or 3. In order to identify potential areas for improvement, we chose to focus our investigation solely on negative reviews. To choose the optimal number of topics for summarizing reviews of a restaurant, we did a grid search with cross-validation and used the topic coherence score as a metric.

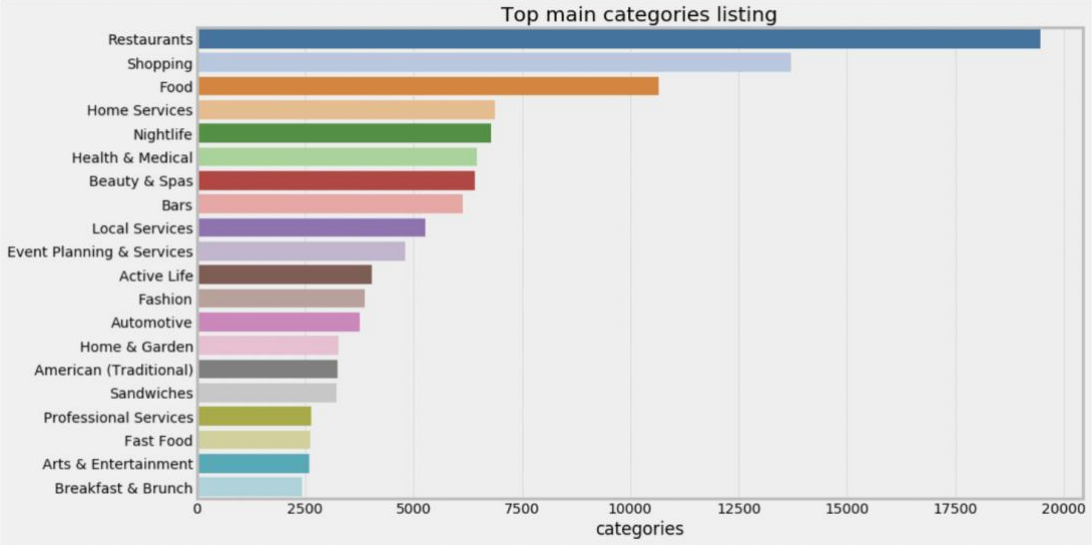

Fig. 2 – Number of reviews corresponding to business category

One of the best ways to evaluate topic modeling is to randomly sample the topics and see if they "make sense". We experimented with varying numbers of topics and discovered that higher values resulted in less subjectively distinct topics. Both LSA and LDA gave fairly good results, however, for some restaurants LDA worked better and for some LSA. We tried to investigate this behavior and found out that LDA and LSA results are solely dependent on inputs and input alone and hence for a particular input one works better than the other.

In the dataset containing all the negative reviews for the restaurants we calculated topic coherency score to find out the optimum number of topics. And keeping that as our hyperparameter we performed topic modeling. One more thing to consider was the number of words. We observed that number of words greater than 7 was not contributing to any novel discovery and hence we restricted the word limit to 7 for each topic.

Among the various restaurants the highest number of negative reviews were associated with the restaurant “Pok Pok” and hence we have presented our results for this restaurant. The highest topic coherency score for this restaurant was observed when the number of topics were equal to 7 and a major drop could be seen for number of topic greater than 7 and hence the choice for our modeling for this restaurant.

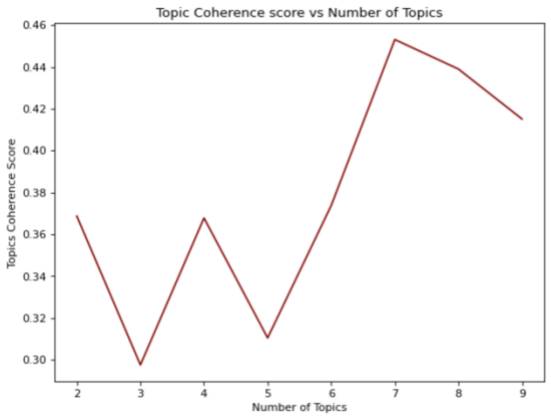

Fig 3. Topic coherence score vs Number of topics for ‘Pok Pok’ restaurant

The table given below summarizes the topics found in topic Modeling for the

restaurant “Pok Pok”. Few of the topics describes the long wait time associated with the restaurant, other hints towards rude staff, we can also see a lot of mention of food in the negative reviews which hints towards the bad quality of food, etc.

Topic Number	Top Words
1	Waitstaff, story, hearing, pokpok, similar, rude, street
2	Pok, seated, hour, america, wit, ordered, restaurant
3	Pok, food, portland, place, better, service, great
4	Food, pok, wing, place, thai, good, like
5	Food, wait, wing, good, place, chicken, dish
6	Food, pok, reservation, table, family, thai, wait
7	Cash, america, place, food, star, good, pok

Table 1: Prominent words corresponding to each topic of LDA for ‘Pok Pok’ restaurant’s reviews.

Content-Based Filtering

The main bottleneck of a content-based recommender system is to generate item features from the dataset. In our dataset, we, fortunately, had different features mentioned in the dataset. Our dataset consists of many attributes of each restaurant such as price range, type of cuisine for example - Chinese, Indian, Japanese, etc., Children-Friendly, or not, Bar, Vegan, Vegetarian, etc. We leverage these features to generate an item profile vector corresponding to every restaurant. Corresponding to every restaurant we had 428 features. Once we had the item-feature matrix, we used KNN to find restaurants that are similar to a given restaurant. To select the optimal value of K, we did a grid search and used a cross-validation score corresponding to accuracy. We found that the optimal value of K=22 for restaurants based in Boston

SVD Based Collaborative Filtering,

First, we generated an item-user matrix from the dataset. Later we used SVD to decompose this matrix into a user-latent factor and item latent-factor matrix. The latent factors here may correspond to the properties of the restaurants such as their category, price range, type of cuisine, etc. From the item-feature matrix, we then use Pearson correlation to find a correlation between every pair of restaurants. Given the past rating of restaurants of a user, the recommender system recommends restaurants by ranking the restaurants by the correlation value. The SVD-based collaborative filtering gave an RMSE of 0.8742.

Neural Collaborative Filtering

To implement NCF, we first encoded business and users' ids. Then we used embeddings to represent each user and each restaurant in the data. To get these embeddings we calculated the dot product between the user vector and restaurant vector. To increase the model performance, we add the "bias" to each embedding. We scale the result using the data's minimum and maximum ratings after running the output of the dot product through a sigmoid layer. We calculated RMSE to evaluate our model. We extracted the item-latent vector embedding layer from the model, which was then used to compute the cosine similarity between every pair of restaurants. And finally, we used this

matrix to recommend restaurants similar to the desired restaurant. Our NCF model performed quite well compared to SVD based Collaborative filtering and gave RMSE of 0.4837.

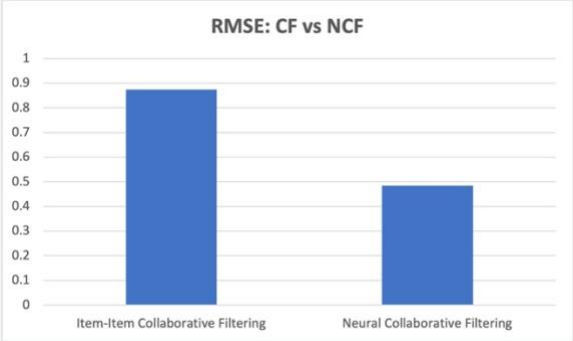

Fig 4: RMSE of CF vs NCF

As expected, we got less RMSE for the NCF model compared to the CF based model. This is because, NCF can learn nonlinear features dependencies in the data, whereas CF assumes linear dependency.

V. Conclusions

In this project, we broadly addressed two major business issues i) given huge amounts of reviews, how to derive actionable insights from them, in order to improve the service, and ii) recommending restaurants to users based on their previous likes and dislikes. To achieve the first objective, we used a Topic modeling approach and classified the reviews of a particular restaurant into broad topics. In order to recommend restaurants to users, we used 3 approaches i) Content-based recommender system ii) Item-item based collaborative filtering and iii) Neural Collaborative Filtering (NCF).

On examining the derived topics for restaurants based in Boston, we were able to decipher their semantic meaning and the topics made sense on visual inspection. In the case of collaborative filtering, we found that NCF performed much better than traditional SVD-based collaborative filtering in terms of reducing the test RMSE. We could achieve, an RMSE of 0.485 for NCF-based collaborative filtering compared to 0.8742 for CF which is quite impressive.

VI. Limitations and Future Scope

In the future, we wish to explore the NCF framework by combining the non-linear MLP-based embedding with that of linear Matrix factorization, so that they can mutually reinforce one another to model the complicated user-item interaction. Another interesting direction would be to design an embedding layer to exploit different data modalities like text, images, video, etc in one framework. This area called Joint Representation learning is an emerging field of recommender systems.

VII. References

[1] Kherwa, Pooja, and Poonam Bansal. "Topic modeling: a comprehensive review." EAI Endorsed transactions on scalable information systems 7.24 (2020).

[2] Blei, David M., Andrew Y. Ng, and Michael I. Jordan. "Latent Dirichlet allocation." Journal of machine Learning research 3 Jan (2003): 993-1022.

[3] Dumais, Susan T. "Latent semantic analysis." Annual Review of Information Science and Technology (ARIST) 38 (2004) 189-230.

[4] Aciar, Silvana, et al. "Recommender system based on consumer product reviews." 2006 IEEE/WIC/ACM International Conference on Web Intelligence (WI 2006 Main Conference Proceedings) (WI'06). IEEE, 2006.

[5] Newman, David, et al. "Automatic evaluation of topic coherence." Human language technologies: The 2010 annual conference of the North American chapter of the association for computational linguistics. 2010.

[6] He, Xiangnan, et al. "Neural collaborative filtering." *Proceedings of the 26th international conference on world wide web*. 2017.

[7] Davidson, James, et al. "The YouTube video recommendation system." *Proceedings of the fourth ACM conference on Recommender systems*. 2010