Restaurant Recommendation System

Shaival Shah
Northeastern University
shah.shaivals@northeastern.edu
Boston, USA

Vaidehi Parikh
Northeastern University
parikh.v@northeastern.edu
Boston, USA

I. Introduction

With the multitudinous amount of data available in the market nowadays, we are well into a data-driven world, wherein customers are spoilt for choice. Filtering, prioritizing and effectively transmitting critical information is necessary to alleviate the problem of information overload, which has become a roadblock for many users. Recommender systems solve this problem by sifting through massive volumes of data to provide consumers with tailored content and services. In an e-commerce setting, recommender systems improve income because they are a cost-effective technique of selling more items.

Yelp is a platform that consists of a massive database of reviews, ratings and other information provided by the users about various businesses. Users still find it difficult to make a well-informed choice when it comes to picking a business. Simply, looking at the ratings is not enough and reading through all the reviews is time consuming for an average user. Due to this, the users would highly benefit from a recommendation system. In the second quarter of 2021, Yelp earned $2.57 billion. Advertising contributes almost 90% of the total revenue and recently companies are shifting toward digital advertising than traditional advertising. In order to increase revenue, companies now focus more on expanding their user base. As a result, Yelp could help users by providing more personalized recommendations, thereby increasing customer engagement. Recommendation systems are basically information filtering systems as they provide elements that are more relevant to the search item or that are related to the search history of the user.

Here, we aim to build a recommendation system that focuses on the restaurant business specifically. This will allow us to make sophisticated food recommendation for all users by applying various predictive algorithms on users' ratings and reviews along with other attributes. In order to build a good model, we implement four different algorithms; Location-based, Collaborative Filtering, Content-based and finally a Hybrid approach. In order to process our data, we perform sentiment analysis using libraries like spacy and NLTK on users' reviews. We also apply LDA(Latent Dirichlet Allocation) to find dominant topics and keywords. Finally, we train our cleaned reviews on the three proposed models and construct the final recommendations. To evaluate our system, we use Precision@K.

II. Related Work

There are two common approaches for providing recommendations - Collaborative Filtering(CF) techniques and Content based Filtering(CBF). The task in CF is to calculate the utility of an item to a user based on the preferences of other users. There are mainly two types of CF algorithms-Memory-based and model based. Memory-based algorithms obtains similar relationships between users or items, by using the ratings given by users to items to provide predictions [17]. Contrastingly, model based learns some latent features which are then used for predictions [16]. The Latent Factor Model factorizes the user-item matrix into two low-rank user and item matrices. This drastically decreases the memory requirements while alleviating data sparsity issues [17]. Farooque et al. proposed a CF-based simple restaurant recommender system, and k-means clustering algorithm was applied to increase the accuracy [18]. CF performs well when there is an abundance of rating information and is limited when it comes to handling the rating sparsity problem. The content-based approach makes use of attributes or keywords of items(previously rated by the user) and creates a user profile based on the features of items rated by that user. Asani et al. proposes a content-based recommender system that analyzes the content of user-written comments and recommends restaurants based on the similarity of their menu to the preferences extracted from the user's comments [20].

However, these methods are not adequate when it comes to personalized recommendations. A personalized review consists of ratings, a textual description and some figures. However, it is not easy for any user to make a decision in real-time with the amount of reviews and the plain structure of text. Sentiment analysis and opinion mining techniques are applied, where reviews and ratings provided by the user are used to understand the user's preferences [2].Liu et al. makes use of restaurant features from users' reviews and a polarity is calculated based on sentiment analysis and machine learning [19].

To solve the cold-start problem, Christakopoulou et al. developed a preference elicitation framework to identify which questions to ask a new user to quickly learn their preferences in restaurant recommendations [19].To mitigate this issue, users' past visit of restaurants is being considered and based on the demographics of the new user fed into the system, recommendations are provided [5].All recommendation systems suffer from the ramp-up problem i.e new items into the database are not easily recommended to the users until they get proper ratings and the same can be said for new users, whose information is not useful until their habits are well known [22]. To alleviate this issue, which is majorly seen in content-based methods, a hybrid recommendation is suggested. Most commonly, collaborative filtering is combined with some other technique. In the weighted hybrid recommender, a recommendation is made by combining results of all available techniques present. It initially gives collaborative and content-based recommenders equal weight, but gradually adjusts the weighting as predictions about user ratings are confirmed or disconfirmed [21].

III. Background Information

A. *Sentiment Analysis*

Sentiment Analysis [6] [8] is one of the most important tasks in NLP. It is the technique of computationally recognizing and categorizing opinions stated in a piece of text, particularly to assess whether the writer's attitude toward a specific topic, product, etc. is positive, negative, or neutral. One of its applications is to analyze customer's feedback/opinion. A sentiment analysis tool can discover positive piece of text demonstrating strengths, as well as negative piece of text demonstrating negative evaluations and difficulties that the users encounter and discuss online. There are multiple ways to perform sentiment Analysis. There are two main approaches for sentiment analysis, one is rule-based method and the other is machine learning based method.

1) **TextBlob**
TextBlob [7] is a Python library for processing textual data. The two measures that are used to analyze the sentiment are:
Polarity Measures how positive or negative the opinion is
Subjectivity Measures how subjective the opinion is

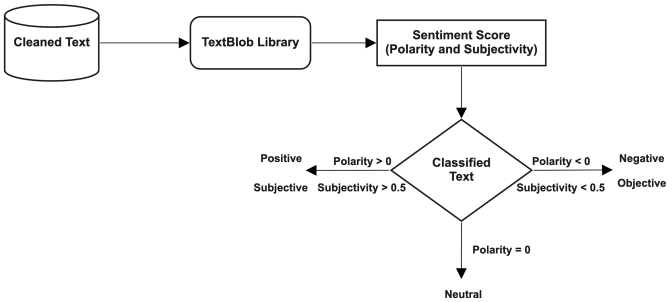


Fig. 1: Sentiment Analysis using Textblob

Polarity ranges from -1 to 1 (1 is more positive, 0 is neutral, -1 is more negative) and Subjectivity ranges from 0 to 1 (0 being very objective and 1 being very subjective).

2) **VADER**
Valence Aware Dictionary and Sentiment Reasoner (VADER) [8] sentiment not only tells if the statement is positive or negative but also the intensity of emotion.

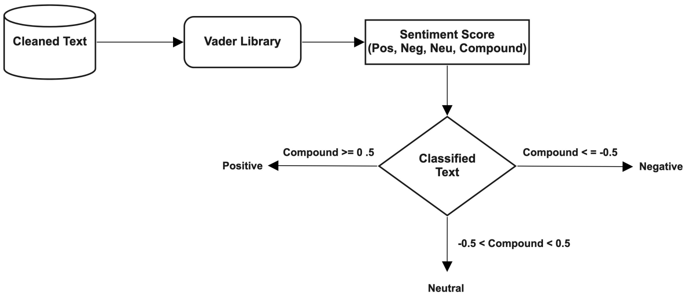


Fig. 2: Sentiment Analysis using VADER

The sum of pos, neg, neu intensities give 1. Compound ranges from -1 to 1 and is the metric used to draw the overall sentiment.

$$\text{Positive} : \text{if compound} \geq 0.5$$
$$\text{Neutral} : \text{if } -0.5 < \text{compound} < 0.5$$
$$\text{Negative} : \text{if compound} \leq 0.5$$

B. *Topic Modeling*

Topic modeling is a form of statistical modeling used to identify abstract ”topics” that appear in a collection of documents/texts. It classifies the documents as different topics based on its semantic similarity or if the text is from the same domain. One of the most popular methods for topic modeling is LDA.

- **Latent Dirichlet Allocation (LDA)**
LDA [1] is a prominent topic modeling technique for extracting themes from a corpus of text. The phrase ”latent” refers to something that exists but has not yet manifested itself. In other terms, latent refers to something that is hidden or concealed. LDA requires n_components i.e. number of topics that we want to have. It is a hyperparameter.

C. *Dimensionality Reduction*

We deal with high-dimensional data in the majority of NLP applications. One of the primary challenges with high-dimensional data is that it is computationally expensive to analyze. Dimensionality reduction is a technique for transforming data from a higher dimensional space to a lower dimensional space in such a way that the low-dimensional representation retains some meaningful properties of the original data, ideally close to its intrinsic dimension.

There are two primary approaches to dimensionality reduction: classic matrix factorization, which is good for retaining global data structure in lower dimension space, and manifold learning, which is good for preserving local data structure in lower dimension space.

1) **Truncated SVD**
When it comes to matrix factorization techniques, Truncated Singular Value Decomposition (SVD) [9] is a popular method to produce features from sparse matrices. It factors a matrix M into the three matrices U, $\sum$, and V.

In truncated SVD, we restrict the number of columns which makes it much faster than SVD and works better when the data is sparse.

2) **t-SNE**

t-SNE [10] is a dimensionality reduction approach that uses Neighbor graphs to reduce dimensions. It addresses the shortcomings of the SNE technique by reducing the cost function and employing the t-distribution in a lower dimension to solve the crowding problem.

3) **UMAP**

Uniform Manifold Approximation and Projection (UMAP) [11] is a type of nonlinear dimensionality reduction technique. If differs from t-SNE in a way that it has an assumption that there exists a dimension where the data is uniformly distributed and the underlying manifold of interest is locally connected. Firstly, it constructs a graph in a higher dimension and then in the lower dimension. If the generated graphs are almost similar, it converges, otherwise it will start from scratch.

D. *Clustering*

Clustering [12] is an unsupervised machine learning approach which groups the data in such a way that the data points in the same cluster have certain similarities. A powerful clustering algorithm can decipher structure and patterns in a data set that are not apparent to the human eye.

- **K-means Clustering**

  K-means clustering is one of the most popular approach to partition the data into k clusters in such a way that the data points in the same cluster are more similar and have more features in common.

E. *Cosine Similarity*

Cosine Similarity measures the similarity between two vectors of an inner product space. Given the vectors $\overrightarrow{V_1}$ and $\overrightarrow{V_2}$, their cosine similarity is calculated by:

$$\mathrm{cos}(\mathrm{V}_1, \mathrm{V}_2) = \frac{\mathrm{V}_1\mathrm{V}_2}{\|\mathrm{V}_1\|\|\mathrm{V}_2\|} = \frac{\sum_{\mathrm{i}}^{\mathrm{n}} \mathrm{V}_{1\mathrm{i}}\mathrm{V}_{2\mathrm{i}}}{\sqrt{\sum_{\mathrm{i}}^{\mathrm{n}} (\mathrm{V}_{1\mathrm{i}})^2}\sqrt{\sum_{\mathrm{i}}^{\mathrm{n}} (\mathrm{V}_{2\mathrm{i}})^2}}$$

Cosine similarity of 1 represents that two vectors are the same whereas a cosine similarity of 0 represents the vectors are different.

F. *Evaluation*

There are many ways to evaluate a recommendation system such as RMSE, MAE, Precision@K, recall@K etc.

Precision@k: It is the proportion of recommended items in the top-k set that are relevant [13].

$$\text{Precision@k} = \frac{\#\text{ of recommended items @k that are relevant}}{\#\text{ of recommended items @k}}$$

Higher the precision@k, more relevant recommendations are retrieved.

IV. Proposed Approach

Our primary objective is to build a personalized **Restaurant Recommendation System** that can recommend places to users based on their interests. This system recommends top k restaurants to a user according to the methodology mentioned below:

A. *Text Pre-Processing*

Firstly, we performed text preprocessing on the user reviews. The task involved removing punctuation, tokenizing, removing special characters, stemming and lemmatizing the text.

B. *Location-Based Recommendation*

Location-Based Recommendation incorporates the location information of a user and provides most popular places which are closer to the location.

The idea behind implementing a location based recommendation system is to tackle the cold-start problem i.e. when there is a new user, we will recommend top k highly rated and nearest places based on the user's current location.

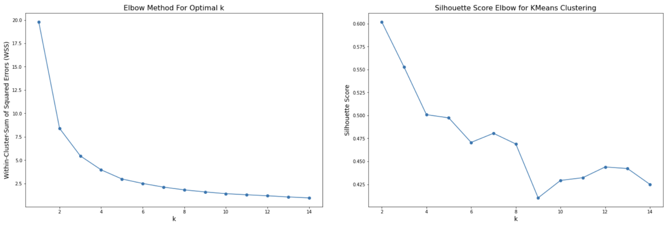


Fig. 3: Optimal k using elbow method and silhouette scores

We implemented k-means clustering to separate the data points based on their latitude and longitude. To find an optimal k, we implemented elbow method and calculated their respective Silhouette scores. As seen in the figure above, there is no significant difference in the silhouette scores. Hence to experiment, we took k as 4, 5 and 6.

The figure below depicts the five clusters received after performing k-means.

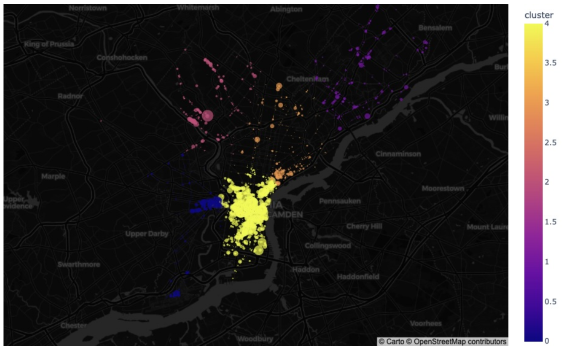


Fig. 4: Clusters retrieved after performing k-means

C. *Collaborative Filtering*

Collaborative filtering [14] is the earliest and most popular method for recommendation.It is a technique that recommends items that a user may be interested in based on the reactions of other users.

There are two methods to perform in collaborative filtering:

1) **User-Based Collaborative Filtering**
   User-Based Collaborative Filtering is a technique for predicting products that a user would like based on ratings provided to that item by other users who have similar tastes as the target user.
2) **Item-Based Collaborative Filtering**
   Item-based collaborative filtering is a form of recommendation system that calculates the similarity of items based on the ratings people have provided to them.

In this project, we implemented item-based collaborative filtering.

- Firstly, we performed sentiment analysis on user reviews using TextBlob's polarity score and VADER's compound scores. We calculated super score which is a combination of user-ratings and the sentiment scores of their reviews.

$$\begin{aligned} &\mathbf{Super\_Score} = \\ &\qquad \mathbf{User\_Rating} + (\mathbf{TextBlob} \times \mathbf{VADER}) \end{aligned}$$

- We created a user-item matrix based on super score and normalized the ratings by subtracting mean ratings.
- We performed Truncated SVD on user-item matrix to create latent features.
- Lastly, we calculated item-item similarity based on cosine similarity i.e.(Pearson's correlation coefficient).

D. *Content Based Filtering*

Content-based Filtering [15] is a technique that recommends items to a user based on item features. It calculates similarity of items based on keywords, categories and other item features.

- We performed Latent Dirichlet Allocation to retrieve five most dominant topics and we categorized each restaurant review with a topic number. We also retrieved ten keywords for each dominant topic.

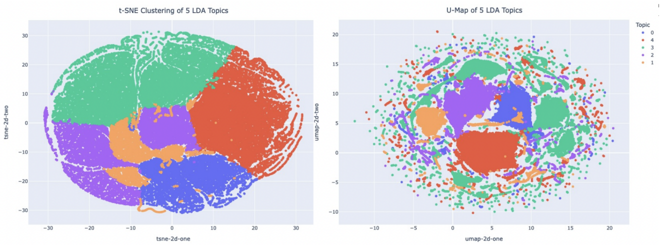
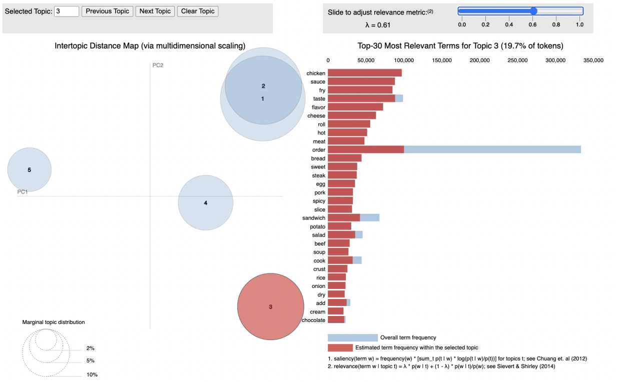


Fig. 5: 2D visualization of t-SNE and UMAP on LDA topics

- We created a bag-of-words by merging the keywords retrieved from LDA with individual restaurant categories and implemented count-vectorizer on the same to convert it into a vector of token counts.
- Lastly, we created an item-item matrix based on cosine similarity.



Selected Topic: 3

Slide to adjust relevance metric:(2) λ = 0.61

1. saliency(term w) = frequency(w) * [sum_t p(t | w) * log(p(t | w)/p(t))] for topics t; see Chuang et. al (2012)
2. relevance(term w | topic t) = λ * p(w | t) + (1 - λ) * p(w | t)/p(w); see Sievert & Shirley (2014)

Fig. 6: Visualization of LDA topics and their keywords

E. *Hybrid Approach*

Hybrid recommendation systems are a combination of single recommendation systems used as sub-components. This hybrid technique was developed to address an issue with traditional recommendation systems.

For hybrid recommendation, we calculated the weighted average of the recommendations received from collaborative and content-based filtering. 60% weightage is given to content-based filtering and 40% to the recommendations received from collaborative filtering.

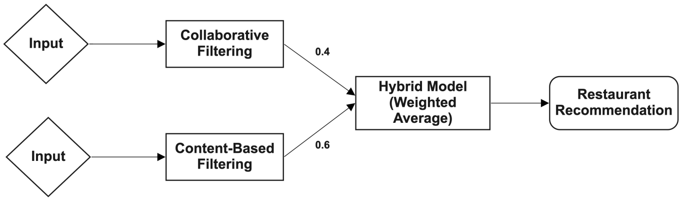


Fig. 7: Hybrid recommendation model architecture

V. Experiments

A. *Dataset*

For this project, we have used the **Yelp Dataset** which has been made publicly available for personal, educational and academic purposes. This datasets consists of a lot of local businesses.The dataset contains 44,0236 restaurants, 8,021,122 reviews, and 1,047,892 users in 10 metropolitan areas located in USA and Canada.

This dataset has three JSON files: Business, User Review, User, and Check-in

- Business: 209K local businesses with stars, attributes, categories such as parking availability, happy hour, drive through, restaurants table service
- Review: 8M user comments, useful votes
- User: User details such as user id, name, review count, useful votes etc.

B. Environment

All the implementation is done on Google Colab Pro with HIGH-RAM (27GB CPU).

C. Exploratory Data Analysis

One of the crucial tasks in any machine learning problem is to deeply analyze and understand the data. The data set contains nearly 200k unique businesses out of which the ”Restaurant” business has the most number of reviews and hence we chose that. The restaurant business covers more than 20 cities, therefore we decided to implement our recommendation system on **”Philadelphia”**.

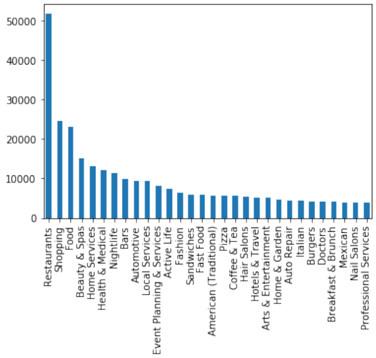


Fig. 8: Distribution of Businesses in Yelp dataset

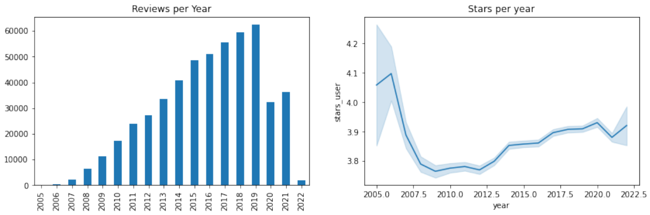


Fig. 9: Reviews and stars per year

The figure above represents the number of reviews received by Yelp each year. It increases every year but started decreasing from 2020. One of the main reasons can be the pandemic of COVID-19.

D. Evaluation

For the purpose of evaluation, we randomly sampled 100 Restaurant names and performed recommendation using different methods. We, then calculated Precision@k on the recommendations received from each method.

We evaluate our metric based on two criteria:

1) **Ratings**
   For Collaborative and Hybrid Approach, if the recommended restaurants have a 4 or more Yelp Rating, then it is a valid recommendation.
2) **Similarity between Restaurant Categories**
   Each restaurant is described by a set of keywords known as categories. We calculated the similarity of keywords between the given restaurant's and the top k recommended restaurants' categories. This is used for content based filtering.

E. Results

The tables below represent top 5 recommendations received for **’Convenient Food Mart’** Restaurant based on different approaches.

Categories of Convenient Food Mart are - ”delis”, ”nightlife”, ”food”, ”beer bar”, ”convenience stores”, ”bars”, ”restaurants”

TABLE I: Recommendations: Collaborative Filtering

Recommended Restaurants	**Categories**	**Ratings**
Milan's Restaurant	american (traditional), breakfast and brunch, restaurants	4.0
Chucks Homemade Waterice	pizza, pretzels, food, ice cream and frozen yogurt, restaurants, american (new)	4.5
Jimmy's Water Ice	food, ice cream and frozen yogurt, pretzels, shaved ice, steaks, restaurants	4.5
The Bagel Hut	restaurants, bagels, food, breakfast and brunch, sandwiches	4.5
Asian Fusion	restaurants, asian fusion	5.0

Collaborative Filtering is able to recommend highly rated restaurants based on other users' ratings.

TABLE II: Recommendations: Content-based Filtering

Recommended Restaurants	**Categories**	**Ratings**
Garden Court Eatery and Beer	nightlife, restaurants, convenience stores, delis, food, beer bar, sandwiches, bars, beer, wine and spirits	3.5
Latimer Delicatessen	beer, wine and spirits, restaurants, nightlife, bars, convenience stores, delis, food	3.0
QQ Deli Market	delis, convenience stores, food, sushi bars, restaurants	3.5
Haggerty Deli	convenience stores, food, restaurants, delis	5
Wawa Food Markets	restaurants, delis, food, convenience stores	3.5

In content-based filtering, though the ratings of the restaurants might not be good, but the recommended restaurants have approximately 70% similar keywords with the given restaurant.

TABLE III: Recommendations: Hybrid Approach

Recommended Restaurants	**Categories**	**Ratings**
Haggerty Deli	convenience stores, food, restaurants, delis	5.0
Asian Fusion	restaurants, asian fusion	5.0
Garden Court Eatery and Beer	nightlife, restaurants, convenience stores, delis, food, beer bar, sandwiches, bars, beer, wine and spirits	3.5
Chucks Homemade Waterice	pizza, pretzels, food, ice cream and frozen yogurt, restaurants, american (new)	4.5
Jimmy's Water Ice	food, ice cream and frozen yogurt, pretzels, shaved ice, steaks, restaurants	4.5

The hybrid approach provides recommended restaurants having similar categories as well as high ratings.

TABLE IV: Recommendations: Location-Based

Recommended Restaurants	**Categories**	**Ratings**
Tortilleria San Roman	Convenience Stores, Italian, Specialty Food, Mexican, Restaurants, Food	5.0
Christie's Deli	Restaurants, Breakfast and Brunch, Delis, Sandwiches	5.0
Bad Brother	Beer Bar, Nightlife, Restaurants, American (Traditional), Burgers, Bars, American (New)	5.0
Mom Mom's Kitchen and Polish Food Cart	Food, Polish, Food Trucks, Street Vendors, Restaurants	5.0
Alma Del Mar	Coffee and Tea, Seafood, Restaurants, Fish and Chips, Food, Mexican, Salad, Breakfast and Brunch, Sandwiches, Comfort Food	5.0

Location-based recommendation always recommends highly rated restaurants which are close to the user's current location.

A summary of the results received after running all three models on 100 randomly sampled restaurants can be seen below:

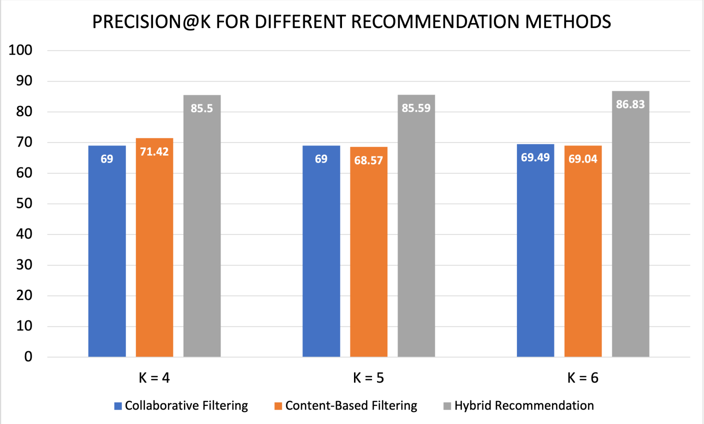


Fig. 10: Results: Precision@k for different methods

VI. Conclusion and Future Work

Adding Demographics into recommendation systems opens up a wide foray of possibilities, and is great for dealing with the cold start problem. Content based filtering performs great in extracting item features and recommending unexplored places. Though the ratings of the recommendations might not be high, The hybrid method performs consistently better than the other two, as it takes into account both content based and collaborative approaches' recommendations.

Our system is limited to one city. Moreover, the user's friends' data has not been taken into account. Hence, integrating it into collaborative filtering might enhance the performance of the system. In future,

1) We would like to incorporate **Graph Theory** in location-based systems to optimize traveling routes.
2) We would also like to explore various **Deep Learning and Neural Networks Architecture** models for obtaining better results.
3) **Pre-trained BERT** weights for topic modeling has lead to better results. So, we can explore more about it.
4) We can create a **user interface** by hosting our model on a website for real-time testing.
5) We can implement a user-centric system and can also predict ratings of our recommendations.

References

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

[2] Sun, L., Guo, J., and Zhu, Y. (2018). Applying uncertainty theory into the restaurant recommender system based on sentiment analysis of online Chinese reviews. World Wide Web, 22, 83-100.

[3] Chen, L., Chen, G., and Wang, F. (2015). Recommender systems based on user reviews: the state of the art. User Modeling and User-Adapted Interaction, 25, 99-154.

[4] Ganu, G., Elhadad, N., Marian, A. (2009). Beyond the Stars: Improving Rating Predictions using Review Text Content. WebDB.

[5] Gupta, A., and Singh, K. (2013). Location based personalized restaurant recommendation system for mobile environments. 2013 International Conference on Advances in Computing, Communications and Informatics (ICACCI), 507-511.

[6] https://getthematic.com/sentiment-analysis/

[7] https://towardsdatascience.com/my-absolute-go-to-for-sentiment-analysis-textblob-3ac3a11d524: :text=Subjectivity

[8] https://www.analyticsvidhya.com/blog/2021/06/rule-based-sentiment-analysis-in-python/

[9] Horn, D., Axel, I. (2003). Novel clustering algorithm for microarray expression data in a truncated SVD space. Bioinformatics, 19(9), 1110-1115.

[10] Schubert, Erich and Gertz, Michael. (2017). Intrinsic t-Stochastic Neighbor Embedding for Visualization and Outlier Detection. 188-203. 10.1007/978-3-319-68474-1_13.

[11] Neil D. Lawrence. 2012. A unifying probabilistic perspective for spectral dimensionality reduction: insights and new models. ¡i¿J. Mach. Learn. Res.¡/i¿ 13, 1 (January 2012), 1609–1638.

[12] Milicchio, F., Gehrke, W. A. (2007). Clustering. Distributed Services with OpenAFS: for Enterprise and Education, 335-350.

[13] https://towardsdatascience.com/evaluation-metrics-for-recommender-systems-df56c6611093

[14] Goldberg, K., Roeder, T., Gupta, D., Perkins, C. (2001). Eigentaste: A constant time collaborative filtering algorithm. information retrieval, 4(2), 133-151.

[15] Melville, P., Mooney, R. J., Nagarajan, R. (2002). Content-boosted collaborative filtering for improved recommendations. Aaai/iaai, 23, 187-192.

[16] Breese, J.S., Heckerman, D., Kadie, C.M. (1998). Empirical Analysis of Predictive Algorithms for Collaborative Filtering. UAI.

[17] Chen, R., Hua, Q., Chang, Y., Wang, B., Zhang, L., Kong, X. (2018). A Survey of Collaborative Filtering-Based Recommender Systems: From Traditional Methods to Hybrid Methods Based on Social Networks. IEEE Access, 6, 64301-64320.

[18] U. Farooque, B. Khan, A. Junaid, and A Gupta, ” Collaborative filtering based simple restaurant recommender” , Intern ational Conference on Computing for Sustainable Global Developm ent. IEEE, 2014, pp . 495-499.

[19] K. Christakopoulou, F. Radlinski, and K. Hofmann, ”Towards conversational recommender systems”, The ACM SIGKDD International Conference. ACM, 2016, pp . 815-824.

[20] Asani, E., Vahdat-Nejad, H., Sadri, J. (2021). Restaurant recommender system based on sentiment analysis.

[21] Burke, R.: Hybrid recommender systems: Survey and experiments. User Model. User-Adapt. Inter. 12(4), 331–370 (2002)

[22] https://en.citizendium.org/wiki/Recommendation_system