UNKI

Ecole Polytechnique Fédérale de Lausanne

EPFL

Unki

A restaurant recommender system for individuals, groups and diversity

by

Student Name	Student Sciper
Alexandre Reis de Matos	282552

Supervisor: Anne-Marie Kermarrec
Project Duration: September, 2022 - January, 2023
Faculty: School of Computer and Communication Sciences, EPFL

Cover : Unki logo (Modified)
Style: EPFL Report Style, with modifications by Batuhan Faik Derinbay

Abstract

This work extends a previous project on restaurant recommender systems, focusing on individuals, groups, and diversity. The recommender algorithms are based on a hybrid collaborative filtering approach, where users and items are embedded in the same space. The offline performance of the baseline and main algorithm is reevaluated, providing a new benchmark for comparison.

An exploration of diversity-driven recommendation reveals that while it increases diversity, it significantly compromises performance, approaching that of a random recommender. A context-aware algorithm utilizing images from reviews is then developed, slightly enhancing overall performance. By refining these algorithms, there is potential for personalized and diverse dining experiences to be further enhanced in restaurant recommendations.

Contents

1 Introduction

1.1. Objective

The objective of this project is to develop innovative approaches for restaurant recommendations by incorporating diversity and image analysis techniques. The aim is to provide users with more personalized and diverse recommendations that align with their preferences and enhance their dining experiences.

1.2. Background

In the age of digital platforms and online reviews, restaurant recommendations play a crucial role in assisting users in discovering new dining options. However, traditional recommendation systems often fall short in capturing the diverse preferences and contextual factors that influence individuals' dining choices. This project addresses these limitations by leveraging advanced techniques to improve the accuracy and relevance of restaurant recommendations.

1.3. Expected Outcomes

The expected outcomes of this project are twofold. Firstly, the developed recommendation system is anticipated to provide users with more accurate and contextually relevant restaurant suggestions by leveraging image analysis, tailored to their individual preferences and contextual factors. Secondly, by incorporating diversity-driven algorithms, the system aims to enhance users' exploration of diverse culinary experiences, leading to a broader range of dining options and potentially uncovering hidden gems.

2
Baseline

2.1. Dataset

The dataset used in this study was obtained by scraping data from TripAdvisor during the previous phase of this project. To recap, the final dataset consisted of 31,643 restaurants, 591,520 reviews, and 349,995 users. Among all the users in the dataset, approximately 5,000 individuals have authored a sufficient number of reviews (10 or more) to be considered active reviewers. These active reviewers have collectively contributed 84,000 reviews.

2.2. Previous Work

In previous studies, the dataset had not been divided into separate training and testing sets for evaluating the performance of algorithms and baselines. Thus, the initial focus of this project was to redefine the evaluation methodology for algorithm performance before proceeding with further innovations.

2.3. Metrics and Offline Reevaluation

Initially, the dataset was partitioned using a leave-one-out methodology, where the last review written by each user was included in the test dataset, while all other reviews were included in the training dataset. The rationale behind this approach was to predict the user's most recent visit based on their previous visits.

For the evaluation metrics, after conducting some research, two primary metrics were selected: precision at k and recall at k [1][4] [5]. Precision at k measures the average precision of the top k recommended elements, while recall at k measures the average recall of these elements. This allowed for the assessment of algorithm performance for various recommendation list sizes. Additionally, these metrics were combined to compute the F1 score.

To allow the computation of these metrics, each element in the recommendation list was classified as either relevant or irrelevant. This classification was based on a custom metric developed in the previous project, which involved calculating the weighted Euclidean distance between the recommendation and the test data point. An element was considered relevant if its error was smaller than 20% of the average weighted Euclidean distance. The weights for the weighted Euclidean average were precomputed for each user based on the restaurants they had visited in the training set.

3
Diversity

3.1. Previous trials

Previous attempts to enhance the diversity of recommendations have proven challenging due to either insufficient diversification or excessive diversification, which resulted in limited control for the algorithm. One approach involved computing the diversity level for each recommendation within the top 100 and sorting them based on this metric. However, this method did not provide enough diversity for most users. Another approach utilized a graph-based method, identifying restaurant communities and employing a PageRank algorithm to select a representative for each community. However, this approach failed to effectively transition between different communities based on the desired level of diversity, as the notion of distance between communities was not adequately relevant. Consequently, the results yielded were relatively random.

3.2. Potential Approaches

The second phase of this project involved reviewing published papers on incorporating diversity in recommendations. It was concluded that achieving serendipity in recommendations is not straightforward and heavily depends on the context. Several interesting metrics were found [8] [10] [3] and out of those one metric was seen as an easy way of evaluating diversity: entropy. Consequently, the focus of this project shifted towards leveraging entropy to enhance diversity.

3.3. Algorithm

Upon examining the available data components, the algorithm has been designed to prioritize the cuisine type parameter, as it represents the aspect in which diversity is most desired. The proposed algorithm operates in seven steps:

1. Construct a graph where restaurants are represented as nodes.
2. Compute edge values as the distance between each pair of nodes.
3. Generate the initial set of recommendations for the user using conventional methods, which serves as the starting set of nodes.
4. Utilize a greedy algorithm that considers all neighboring nodes of the set in the graph and adds the one that maximizes entropy.
5. Iterate until the moving average of entropy starts to decrease.
6. Remove the restaurant that maximizes entropy the most by being removed from the set
7. Iterate on previous step until there are only k restaurants left

By following this approach, diversity is incrementally increased with each iteration while still remaining close to something that the user enjoys, and the algorithm halts when neighboring nodes no longer contribute to diversity. Additionally, this allows users to specify their desired level of ”risk” in terms of diversifying their recommendation set.

3.4. Computational Limitations and Solutions

The aforementioned algorithm holds promise in theory; however, it presents several computational challenges that necessitated the exploration of solutions to implement the algorithm effectively.

Initially, precomputing the entire graph was considered, but constructing a complete graph with 30,000 nodes proved impractical. Instead, the approach involved filtering restaurants based on the average coordinates of the user's past visited locations within a given radius. This approximation enabled the selection of nearby restaurants, but the computational cost remained significant when the radius was large or when the user was located in an area with numerous restaurants. To address this, the algorithm employed a radius adjustment strategy: starting with a small radius and gradually increasing it until at least 100 restaurants were included for consideration. If increasing the radius led to more than 1,000 restaurants, the algorithm randomly selected 1,000 restaurants from the available list. Subsequently, all potential edges were considered, but only 10% of them were computed, selected uniformly at random.

Due to the randomization process, the initial set of nodes may form a cluster disconnected from the rest of the graph. To mitigate this issue when it happened, edges were added between each node within the set and any other node in the graph, with the constraint of maintaining a maximum of 1,000 neighbors to the initial cluster. Distance computation was then performed between each neighbor and each node within the set, and the neighbor that maximized entropy was selected.

The values of 100 and 1,000 for the limits were determined through experimental evaluation to ensure computationally feasible yet meaningful results. However, alternative values may better suit specific scenarios. The randomization process for computing nodes addressed the issue of the initial set being disconnected from the rest of the graph in less than 10% of cases, indicating its efficiency.

On average, the algorithm increased entropy by approximately 50%.

4

Context-Aware recommendation

4.1. Background and Related Work

The scrapped TripAdvisor dataset also included images from reviews and restaurants. Based on the idea of considering visual information for predicting favorite restaurants [11] and in line with the current trend of multimodal machine learning systems, the decision was made to explore the use of images to enhance the recommendation process. The objective was to leverage these images to detect the food and drinks served at different restaurants and obtain additional information for improving recommendations.

To accomplish this, a training set was required to train a model capable of generating such labels. An AIcrowd contest [6] addressed this specific task, providing participants with a training dataset of 50,000 labeled images from user reviews and a testing dataset of 1,000 images.

4.2. Computer Vision Model

The winning teams of the AIcrowd contest shared their codes and the models they used. The winning model achieved an average precision of 0.423 and an average recall of 0.636, demonstrating promising results. The winning solution was based on the MMDetection library. However, due to limitations imposed by the code and library dependencies, it was not possible to run the winning code after two weeks of attempts. Consequently, an alternative approach was pursued, involving training a model from scratch.

Using the detectron2 library and following the advice of one of the highly ranked submissions from the contest, a faster R-CNN model was trained for six epochs. While the model's performance did not match that of the winning models with an average precision of 0.12 and an average recall of 0.19, it still generated partly correct food labels as can be seen in figure 4.1. Although the recall was low, meaning the model did not detect all foods possible, the redundancy of inputting up to 9 images per restaurant, still made it an acceptable estimation of the food types present. The lower precision was not a significant concern for this project, which prioritized recall, rather than knowing exactly where the foods were.

(a) Labelled as hamburger **(b)** Labelled as chocolate cake **(c)** Labelled as carrot

Figure 4.1: The bounding boxes show one of the potentially many labels for these images, but they might have included others, in particular the last one

4.3. Training and Labeling

Initially, the plan was to use images from reviews and user profiles to enhance the recommendation system. However, only around 50% of users with at least 10 reviews (those considered for this work) had uploaded images with their reviews. As a result, the focus shifted towards building more robust restaurant profiles by completing the restaurant entries in the dataset with foods detected from user-uploaded pictures. This approach led to 86% of restaurants being labeled with food items, with an average of 17 types of foods per restaurant. These labels encompassed 320 different categories, including ”pasta,” ”french fries,” ”beef,” ”lasagna,” ”red wine,” and more.

4.4. Addition to User's Dataset

Once the restaurant profiles were completed, the user profiles were augmented in a similar manner to the previous completion of cuisine types. The process involved examining the restaurants where each user had visited, considering the food types associated with those restaurants, and generating a ranking of foods for each user.

4.5. Distance Computation

To compute the distance between a user’s vector and a restaurant, the same distance metric as in the previous project, with the addition of the food component, was employed. This food component involved assessing how many of the restaurant’s food types were present in the user’s set of foods, weighted by the previously defined ranking. The resulting value was divided by the total number of foods at the restaurant. In other words, a perfect score would indicate that all of the restaurant’s foods were present in the user’s set.

It is important to note that the difference with the cuisine distance calculation is that, in this case, the quantity of food hits is taken into account, rather than only considering the highest-ranked cuisine type from the user’s cuisine ranking.

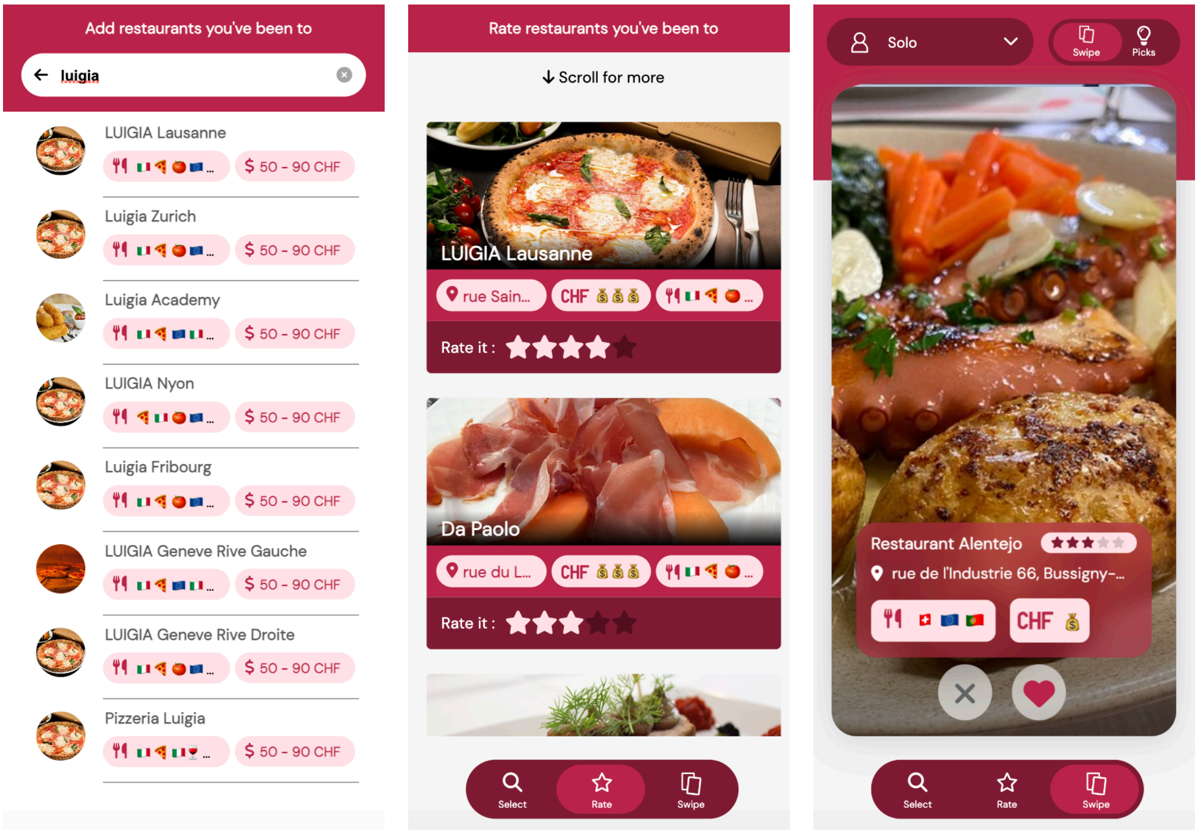
5
Online evaluation

5.1. Initial Idea
The initial idea for this project was to create an online testing platform where user interactions would be captured through card swiping on restaurant recommendations. The goal was to perform A/B testing on different algorithms and gather data on their respective performances. The intention was to validate the offline results through online testing.

5.2. Implementation
The web app was developed using HTML, CSS, and JavaScript, with Flask used as the backend framework to run the collaborative filtering algorithm. The implementation involved creating a user-friendly interface where users could swipe through restaurant recommendations and provide feedback on their preferences. The web app is shown in figure 5.1.

During this stage, the limitations of the dataset, particularly regarding image quality, played a significant role. TripAdvisor heavily relies on images uploaded by users, which posed challenges in terms of data quality. The implementation process can be visually illustrated with the help of images to showcase the website's functionality and user interface.

However, there were several difficulties encountered due to limited experience in this type of development, such as setting up a remote server to run the Python script. Additionally, as the project shifted and the startup it was being developed for pivoted towards a simpler product, the idea of the online platform was ultimately abandoned. Despite investing 10 weeks of work and having a functional prototype working locally, technical challenges and the change in the startup's direction led to the discontinuation of the online platform idea.



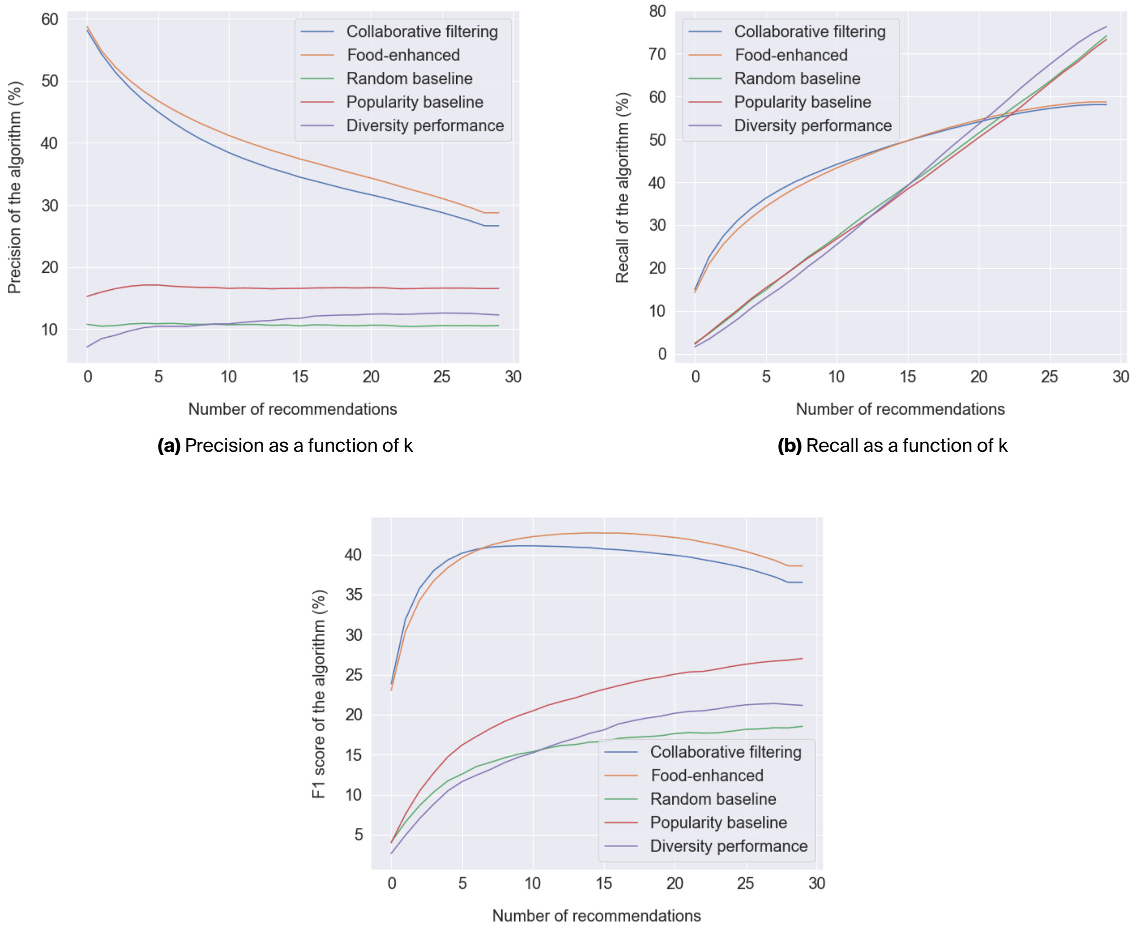
(a) Users pick restaurants they've been to **(b)** Users rate their experience **(c)** Users swipe on recommendations

Figure 5.1: Flow of the developed web app

5.3. Next steps

While the online testing platform was not realized in this project, the need for online testing remains in the future. With the development of the minimum viable product for the startup, there is hope that it may be utilized to perform the online testing required to evaluate the algorithms developed in this project.

6
Results and analysis



(a) Precision as a function of k

(b) Recall as a function of k

(c) F1 score as a function of k

Figure 6.1: Performance of the different algorithms implemented

6.1. Baselines

As expected, the precision of the baselines is rather low. However, the popularity-based recommender, which recommends restaurants from a distribution based on the most reviewed restaurants, still outperforms the random recommender. On the recall side, both baselines increase linearly with a growing list size. This is because no sorting was applied to the recommendations of the baselines, resulting in evenly distributed relevant recommendations and thus linear recall growth with the increasing list size.

6.2. Context-aware recommendation

Despite the weak performance of the image classifying model, the results of the context-aware algorithm are promising. Restaurants could have as many as 9 images associated, meaning that even with low recall from the computer vision model, it was still possible to extract relevant food types and infer the context of the restaurants' food types. The algorithm managed to outperform the classic collaborative filtering method in terms of precision, while achieving similar recall. Further improvements in food identification precision could enhance its performance. In terms of overall performance measured by the F1 score, the image-based algorithm outperformed the classic algorithm on longer recommendation lists, while exhibiting similar performance on shorter lists.

6.3. Diversity

The performance of the diversity algorithm was somewhat surprising, despite the fact that the focus was put on diversifying recommendations rather than optimizing performance. The diversity algorithm's performance falls between the random recommendation and popularity recommendation baselines. The random assignment of edges in the graph undermined the proximity notion initially intended, as the edges were not always connected to the most similar nodes in the initial set. Consequently, adding nodes to increase entropy deviated from the intended proximity concept, resulting in decreased performance. The results indicate that the procedure is somewhat similar to adding random nodes to the set, and once again, no sorting was applied, resulting in evenly distributed relevant picks across the recommendations, hence the similar recall performance to the baselines.

6.4. Key findings

The natural proximity of nodes in the graph for the diversity algorithm alone is insufficient to maintain high performance while diversifying the list of recommendations. Considering relevance alongside diversity is essential in the recommendation process. The overall performance, as measured by the F1-score, for the collaborative filtering and context-aware algorithms remained around 40%, which is suboptimal. However, the context-aware algorithm shows promise, particularly if enhanced with a high-performing computer vision model, which could potentially boost the algorithm's overall performance. This work demonstrates the potential of enriching data with additional contextual information to improve recommendations and holds promise for future advancements in the field.

6.5. Conclusion and next steps

Based on the conclusions drawn from the previous section, the natural progression is to explore multi-modal machine learning techniques, which consider different types of data sources and combine them to achieve superior results. This approach will be the focus of an upcoming master thesis, aiming to leverage user input in conjunction with existing user interactions and images to enhance recommendation outcomes. Deeper context and cultural differences are elements to keep in mind as well [7] [9]. The idea of sentiment analysis [2] could also be explored since the actual textual content of reviews has not been explored yet.

References

[1] Jaime Arguello. *Evaluation metrics*. 2013. URL: https://ils.unc.edu/courses/2013_spring/inls509_001/lectures/10-EvaluationMetrics.pdf.

[2] Javad Sadri Elham Asani Hamed Vahdat-Nejad. "Restaurant recommender system based on sentiment analysis". In: *Machine Learning with Applications* 6 (2021). URL: https://www.sciencedirect.com/science/article/pii/S2666827021000578.

[3] Filip Radlinski Javier Parapar. "Towards Unified Metrics for Accuracy and Diversity for Recommender Systems". In: *RecSys '21* (2021), pp. 75–84. URL: https://dl.acm.org/doi/10.1145/3460231.3474234.

[4] Loren Terveen Jonathan Herlocker Joseph Konstan. "Evaluating Collaborative Filtering Recommender Systems". In: *ACM Transactions on Information Systems* 22 (2004), pp. 5–53. URL: https://grouplens.org/site-content/uploads/evaluating-TOIS-20041.pdf.

[5] M. Ebrahimi M. Kshour. "New recommender system evaluation approaches based on user selections factor". In: *Heliyon* 7 (2021). URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8278338.

[6] Shivam Khandelwal Marcel Salathé Sharada Mohanty. *Food recognition benchmark 2022*. 2022. URL: https://www.aicrowd.com/challenges/food-recognition-benchmark-2022.

[7] Francesco Colace Mario Casillo. "Context-aware recommender systems and cultural heritage: a survey". In: *Journal of Ambient Intelligence and Humanized Computing* 14 (2021), pp. 3109–3127. URL: https://link.springer.com/article/10.1007/s12652-021-03438-9.

[8] Tomaž Požrl Matevž Kunaver. "Diversity in recommender systems – A survey". In: *Knowledge-Based Systems* 123 (2017), pp. 154–162. URL: https://www.sciencedirect.com/science/article/pii/S0950705117300680.

[9] Jason J. Jung Minsung Hong. "Multi-criteria tensor model for tourism recommender systems". In: *Expert Systems with Applications* 170 (2021). URL: https://www.sciencedirect.com/science/article/pii/S0957417420311817.

[10] Neil J. Hurley Saul Vargas Pablo Castells. "Novelty and Diversity in Recommender Systems". In: *Recommender Systems Handbook* 1 (2015), pp. 881–918. URL: https://link.springer.com/chapter/10.1007/978-1-4899-7637-6_26.

[11] Ya-Lun Tsai Wei-Ta Chu. "A hybrid recommendation system considering visual information for predicting favorite restaurants". In: *World Wide Web* 20 (2017), pp. 1313–1331. URL: https://link.springer.com/article/10.1007/s11280-017-0437-1.