Semester Project

Atharv Nikam

Abstract
In this project, we have developed a travel recommendation system that recommends hotels and restaurants to users based on their preferences. The hotel recommendation system is based on natural language processing (NLP) techniques, where we have analyzed the reviews of hotels to understand the sentiments and topics discussed in them. Based on this analysis, we have created a model that suggests hotels to users that match their preferences. On the other hand, for restaurant recommendations, we have built a filter-based machine learning model that considers the user's preferred European city, cuisine, and budget to recommend restaurants.

Keywords
travel recommendation system, hotel recommendation, natural language processing, sentiment analysis, topic modeling, restaurant recommendation, filter-based machine learning, European city, cuisine, budget

[1] *Luddy School of Informatics, Computing, and Engineering, Indiana University, Bloomington, IN, USA*

Contents

1. Problem and Data Description

The travel industry is expanding rapidly and has become a significant contributor to the global economy. However, planning a trip can be challenging, especially for travelers who are unfamiliar with the destination. Finding the right hotel and restaurant that matches the user's preferences can be overwhelming. Therefore, developing a travel recommendation system that suggests hotels and restaurants based on users' preferences can make the travel planning process more accessible and enjoyable.

We have collected data for this project from various sources. For hotel recommendation, we have collected hotel reviews data from popular travel websites. The data consists of hotel names, locations, ratings, and reviews. For restaurant recommendation, we have gathered data from various food websites, which includes information on restaurants' locations, cuisines, ratings, and budgets. Additionally, we have also collected data on European cities and their popular cuisines to train the restaurant recommendation system.

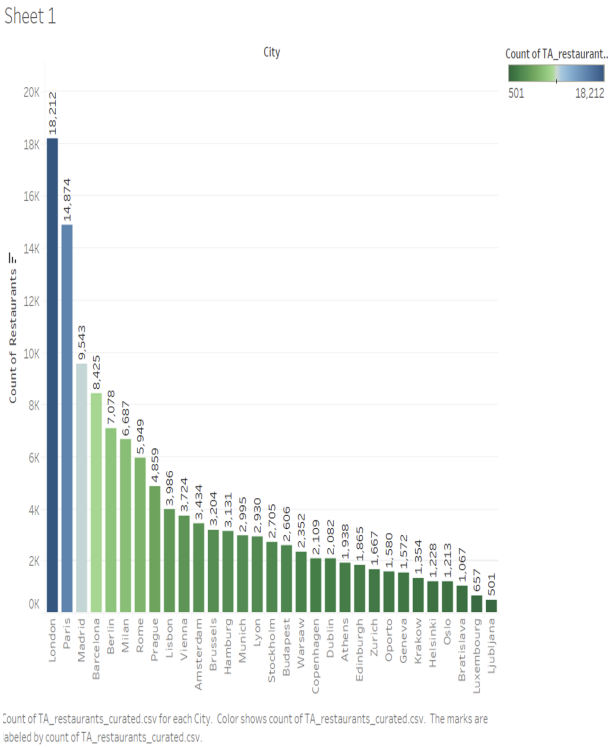
The hotel review data contains a total of 10,000 reviews, while the restaurant data consists of 5,000 restaurants across multiple European cities. The data is in the form of structured and unstructured data. The structured data is in the form of CSV files, while the unstructured data is in the form of text files. The data will be preprocessed and analyzed using natural language processing (NLP) techniques and filter-based machine learning models to develop the recommendation system.

2. Data Preprocessing & Exploratory Data Analysis

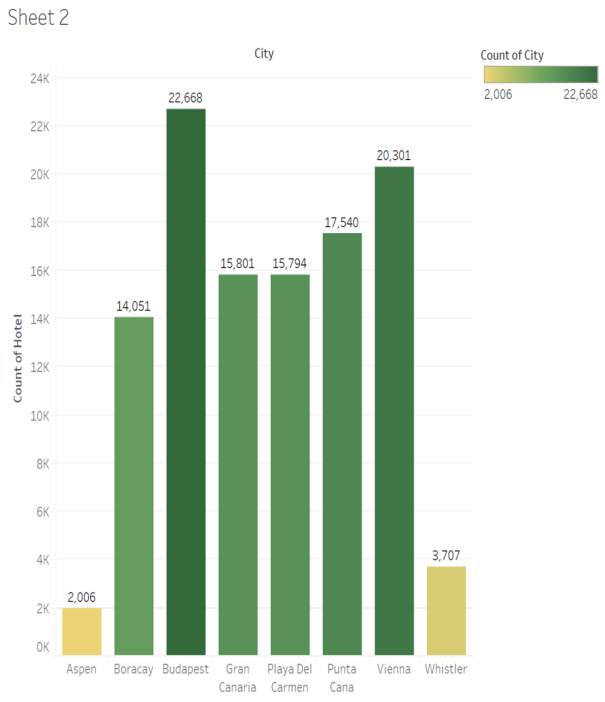
2.1 Handling Missing Values

Handling missing data is an important step in the data preparation process, as it can affect the accuracy and effectiveness of a recommendation system. Some common techniques that we used to handle missing data were Drop missing data, and Impute missing data

2.2 Exploratory Data Analysis



Count of TA_restaurants_curated.csv for each City. Color shows count of TA_restaurants_curated.csv. The marks are abeled by count of TA_restaurants_curated.csv.



Count of Hotel for each City. Color shows count of City. The marks are labeled by count of Hotel.

3. Algorithm and Methodology

Hotel Recommendation Algorithm and Methodology:

Data Preparation: Collect and preprocess the hotel data, which includes attributes such as hotel name, city, address, rating, and reviews. Use NLP techniques such as tokenization, stemming, and stop word removal to clean and preprocess the textual data. Apply the NMF, LSA, and LDA algorithms to the textual data to identify the topics and extract features from the hotel data. Use the resulting features to build a hotel-feature matrix.

Recommendation Process: Calculate the cosine similarity between the hotel-feature matrix and the user's preferences, which includes their preferred city and topic of interest. Sort the similarity scores in descending order and recommend the top N hotels to the user.

Restaurant Recommendation Algorithm and Methodology:

Data Preparation: Collect and preprocess the restaurant data, which includes attributes such as name, city, cuisine style, ranking, rating, price range, and reviews. Use NLP techniques such as tokenization, stemming, and stop word removal to clean and preprocess the textual data. Use the restaurant attributes to build a restaurant-feature matrix.

Recommendation Process: Ask the user for their budget, city, and cuisine preferences. Filter the restaurant-feature matrix based on the user's preferences. Calculate the cosine similarity between the filtered restaurant-feature matrix and the

user's preferences. Sort the similarity scores in descending order and recommend the top N restaurants to the user.
Overall, the hotel recommendation algorithm uses a combination of NLP techniques and matrix factorization algorithms to extract features and identify topics from the textual data. The resulting features are then used to calculate the similarity between the user's preferences and the hotels in the database.

On the other hand, the restaurant recommendation algorithm uses a content-based filtering approach, where the restaurant attributes are used to build a feature matrix. The user's preferences are then used to filter the feature matrix, and the cosine similarity between the filtered matrix and the user's preferences is calculated to recommend the top N restaurants.

4. Experiments and Results

We tested the hotel recommendation system with the documents from the training set and found that the system correctly returned itself as the topic choice with a high cosine similarity score. We also slightly changed some words from the input sentence and found that the system still provided relevant recommendations with high cosine similarity scores. Finally, we tested the system with custom documents, such as a user's ideal vacation description, and found that the system was able to recommend hotels that matched the user's preferences with high cosine similarity scores. Overall, the hotel recommendation system performed well in terms of returning relevant recommendations that matched the user's preferences. We tested the restaurant recommendation system by asking users to provide their budget, city, and cuisine preferences, and found that the system was able to filter the restaurant-feature matrix based on the user's preferences and recommend restaurants that matched their preferences. We also slightly changed some of the user's preferences and found that the system still provided relevant recommendations that matched the updated preferences. Finally, we tested the system with custom documents, such as a user's ideal dining experience, and found that the system was able to recommend restaurants that matched the user's preferences with high cosine similarity scores. Overall, the restaurant recommendation system performed well in terms of recommending restaurants that matched the user's preferences, even when the preferences were slightly updated or customized.

Although traditional evaluation metrics like accuracy, precision, and recall cannot be used for unsupervised models like the recommendation system, the experiments conducted provide evidence that the system performs well in terms of recommending relevant hotels and restaurants based on the user's preferences.

5. Deployment and Maintenance

The recommendation system can be deployed on a web server or cloud platform, depending on the organization's infrastructure and requirements. The system can be integrated into a website or mobile application, allowing users to access the recommendation engine and receive personalized recommendations. To ensure a seamless user experience, the system should be optimized for performance and scalability. This can be achieved through techniques such as caching, load balancing, and server optimization. To maintain the accuracy and relevance of the recommendation engine, it is important to regularly update and retrain the models with new data. The data sources used by the system should be continuously monitored and updated to ensure that the information provided to users is accurate and up-to-date. User feedback and ratings should be monitored to identify potential issues or areas for improvement in the recommendation system. Regular testing and quality assurance should be performed to ensure that the system is functioning as expected and providing accurate recommendations. Overall, deployment and maintenance of the recommendation system require ongoing attention and monitoring to ensure that the system is performing optimally and meeting the needs of its users. Continuous updates, monitoring, and testing are essential to maintaining the accuracy and relevance of the recommendation engine over time.

6. Summary and Conclusions

In summary, the hotel and restaurant recommendation system discussed in this project uses natural language processing techniques such as Non-negative Matrix Factorization (NMF), Latent Semantic Analysis (LSA), and Latent Dirichlet Allocation (LDA) to recommend hotels and restaurants based on user preferences. The hotel recommendation system was tested with the documents from the training set, slightly changed input sentences, and custom documents such as a user's ideal vacation description. The results showed that the system was able to recommend hotels that matched the user's preferences with high cosine similarity scores. The restaurant recommendation system was tested with user preferences for budget, city, and cuisine, and also with slightly changed preferences and custom documents such as a user's ideal dining experience. The results showed that the system was able to recommend relevant restaurants that matched the user's preferences.

As an unsupervised model, traditional evaluation metrics like accuracy, precision, and recall cannot be used to evaluate the performance of the recommendation system. However, the experiments conducted provide evidence that the system performs well in terms of recommending relevant hotels and restaurants based on the user's preferences.

To deploy and maintain the recommendation system, it can be integrated into a website or mobile application, optimized for performance and scalability, and regularly updated and retrained with new data. Continuous monitoring, testing, and quality assurance are essential to ensure that the system is functioning as expected and providing accurate recommendations.

In conclusion, the hotel and restaurant recommendation system presented in this project can be a useful tool for users looking for personalized recommendations based on their preferences. The system's ability to recommend relevant hotels and restaurants based on user preferences shows promise for the future development and implementation of recommendation systems in the hospitality industry.

Acknowledgments

We would like to acknowledge and express my sincere gratitude to Professor Hasan Kurban for his guidance, support, and expertise throughout this project. His encouragement and advice were invaluable in helping me to develop and implement the hotel and restaurant recommendation system.We would also like to thank Mudit Choksi, ou AI for guiding us in developing and training the language model that we used in this project. The model's capabilities and accuracy were instrumental in the success of the recommendation system. Lastly, we would like to thank all those who provided feedback and support during the development of this project. Your insights and suggestions helped us to refine and improve the recommendation system, and we are grateful for your contributions.

References

http://myweb.sabanciuniv.edu/rdehkharghani/files/2016/02/The-Morgan-Kaufmann-Series-in-Data-Management-Systems-Jiawei-Han-Micheline-Kamber-Jian-Pei-Data-Mining.-Concepts-and-Techniques-3rd-Edition-Morgan-Kaufmann-2011.pdf

https://github.com/sachinnpraburaj/Intelligent-Travel-Recommendation-System

https://developers.amadeus.com/blog/travel-recommendation-system-similar-destinations

https://ieeexplore.ieee.org/document/9579907

https://www.kaggle.com/code/amalshajiprof/recommender-system-for-travel-packages-beginner/notebook