

Recommendation System of Hotels based in Rome

Sona Hakobyan

BS in Data Science American University of Armenia Yerevan, Armenia

sona_hakobyan2.edu.aua.am

Advisor: Vahagn Saghatelyan

American University of Armenia Yerevan, Armenia

Abstract—My project is aimed to create a recommendation system for hotels in Rome. This will create an opportunity to just chat with the AI assistant and give specific details about what kind of hotel one needs to attend mentioning features like location, street, accommodations, rooms, number of people, purpose, feedback and so on and get the best match for the mentioned needs. The data I'm using is an officially licensed dataset from Airbnb which is appropriate to use for the recommendation system. I used embeddings for each row in the dataset, which gives the opportunity to get the answers for the specific question regarding all the information about the hotels in Rome in a few seconds. Using various techniques that we'll discuss later I have created an interactive chatbot specialized in hotel recommendations which is easy to use just by transferring your thoughts as an input.

I. INTRODUCTION

In today's world, the method of digital solutions, implementations and renovations has become the most important aspect for all spheres of business. They are trying to incorporate artificial intelligence and machine learning best practices into their business model to attract more customers. Firstly it will help to boost the investigation of their business field and be accessible from all around the world, on the other hand it will be easy to use for people without having any trouble. Secondly, the advertising of the products is much more effective and influential via technology use. One of the greatest latest innovations in the tech field are generators such as Gemini and OpenAI, which are already doing the same research instead of a user. This all aims to use systems for automated and faster environments providing users with better experience and easier interaction with knowledge in any form. Most people strive to not waste their time researching their needs on google or websites but just generate a prompt and get instructions, ideas in any aspect of their life starting from entertainment like hotels, cafes, and restaurants ending with learning something new. There are widely known websites and applications which give opportunity to users to give their thoughts, likes, comments, recommendations for next comers to read and view. Meanwhile the user expertise does not always show the main favorable part for others. So we can assume that all the necessary information is important for the specific search. During the last few years it has become one of the ways to implement chat bots which will help users

to find the things they want through just transferring their thoughts in a message. My project is an implementation of a recommendation system for hotels based in Rome. One can easily access and chat about specific needs and get the answer of best matches for the request.

II. METHODOLOGY

The idea of the recommendation system for hotels was not the first step in the 4-month journey during which I needed to initiate and work on this project. The first idea that came to my mind with the help of my supervisor was "next best offer" in the travel and hospitality industry, which assumed having historical customer data from various hotels and based on analyzing the preferences of customers design campaigns and target them suggesting the best offer and optimizing the communication channels again based on the history, taking the one that's most effective and influential on a specific customer. I did my research and could only find hospitality related customer data for 2 american hotels, which didn't seem to be enough base to construct my model on. I met with managers of 2 hotels' managers which are Apricot hotel Yerevan and Radisson Blu hotels. It turned out that both of those companies do not have a proper data collection system and there is no possibility to collect the type of data that would help my model work. The other stage was trying to redirect my project idea a little bit towards just a hotel recommendation, not based on historic customer data, but the prompt or filters that the user will choose. Then I started to search for data that would support the scope of my recommendation system. The most suitable options were Booking.com and Airbnb.com. I tried different approaches of scraping. First of all scraping through html tags, but they were too nested and the versions of scripts I tried failed to find the text inside those tags. Then I tried to do scraping with BeautifulSoup library, which just took everything visible on a single link and returned a messy text. I decided to take these texts from every link I would scrape and then through regular expressions clear them out in a format that after could be used for the model. After spending a lot of time on this I found out that even if I succeeded finding patterns on a single hotel it differed from hotel to hotel which means that for one hotel I managed to scrape hotel name, address, reviews, house rules and so on but for the other hotel the regex had problems finding

every section. Finally, I found a website that has scraped data from Airbnb in various aspects such as Listings, reviews and neighborhood for all the following countries: Australia, Canada, France, Germany, Greece, Italy, The Netherlands, Portugal, Spain, Sweden, the United Kingdom and the United States. Observing those datasets I found out that they are too big and there is no machine that I can use currently that has enough computational resources to process that much information so I decided to narrow down my project on one city and chose Rome for that. That was a big dataset too, in this dataset I have around 30000 hotels and more than 1.6 million reviews for those hotels.

III. IMPLEMENTATION

There are lots of other systems around the internet where the user can enter and search the best place he/she imagines to stay, however that process has its limitations. A person will have to manually select among the filters to find what he/she needs and even if there are a bunch of features both on Booking and Airbnb where they could look for and put a tick but this process might be tiring and non-optimal. It will be much easier to just write as a small text whatever you want to do not want to have in the flat/ hotel that you plan to stay in and the system will analyze your text and give the best results suiting the prompt. What's more, those booking websites just allow you to choose for raw facts like number of rooms, existence of some objects and so on, while my model is capable of analyzing the textual data like hotel description, neighborhood description, host description, reviews and amenities for a better performance. Another competitor of my model could be chat gpt itself, which I have used in it too. One could ask, why would people prefer to use the hotel recommendation assistant rather than chat gpt and the most obvious explanation is that gpt is not up-to-date it contains information and is trained on data before a date while in case of improving and working on my recommendation system I can write an every-day updating scraping code which will be more up-to-date than chat gpt model. However, this is not something that we could expect to overperform chatgpt that was trained on billions of parameters. This model solves a specific task and is good at its specific task. Now that it is clear why and how this model can be used let's dive deeper through the techniques that have been used to implement this project. The data needed to be processed, so first of all I dropped columns that I couldn't make use of, then joined the listings and reviews tables. To have a single row describing a single hotel I decided to group all the comments about every unique id with dots. I ended up with a dataset with 8000 rows and a huge amount of textual data in every row. To be able to make the embeddings optimally I decided to truncate data, make the model work on a small scale and later on develop it. So I took the top 10 hotels and 10 comments for each hotel, concatenated those comments and finally concatenated all the other data in one column with the column name, value pairs. The final data had 10 rows with 2 columns one which was listing url and the other one was combined text. For my recommendation system I decided

to use the logic of Retrieval-Augmented generation which does output generation based on information retrieved from an external knowledge source, in this case our vector db (Deldjoo et al., 2024). So what it really does is take information in a vector db and use a generating model to provide output. Now let's refer to the first part of storing information. To turn the combined text of each hotel to embedding I used OpenAI embeddings and then used ChromaDB to store them in one place. Then I used langchain which ensures the easy retrieval and usage of data. What about the second part, I used the gpt-4-0125-preview model to provide the generation part based on the data and the prompt. To make this model accessible I used the streamlit framework to have a simple frontend where a user can search for their preferred hotel giving the prompt.

IV. LITERATURE REVIEW

The Recommendation systems became actual during the last decade, when people started to use big data and machine learning as systems of implementing and relating it to the field which can be actually non-related to IT. First of the AI generators which become very popular in the world is chat gpt. The use of AI generator platforms brings people to easily get information related to any kind of question which is in the scope of google, Bing and other search platforms. One can easily use it just by entering the question in the chat and interpreting all the necessary information which should be added to the search query. Only after a few seconds all the information accessible to the chat gpt will generate an answer which will be a great match to the specific question. However, because of unlimited data in the world which is not real to refresh even day by day, the scope of the gpt 3.5 becomes non actual after a time and answers can be accessible only for the specific date and time. As a solution to this issue OpenAI created the 4th version which has a unique way to access google or Bing to do research which is not actually known by the documentation. So we can assume that the AI generators are trained only on existing data. On the other hand the gpt is very confident in the answers which are generated and one has to know the right answer to fully understand if the information is right or wrong. This is an issue for the society to understand the answers correctly. According to researchers the information about the recommendation system has become popular during the last decade (B.V., 2018). Company owners bought interesting chatbots which will create much more easy access for their users. Taking into account this information we can assume that every user strives to find the easiest way to get anything that is interesting to them, the best example of this is chatbot. Streamlining the selection process for cafes, restaurants, and hotels, our chatbot simplifies queries by allowing users to input their preferences directly. Instead of navigating through complex filters, users can simply describe what they're looking for, such as open views or cozy seating for cafes, or specific amenities like WiFi and parking for hotels. Leveraging deep learning-based recommender systems, the chatbot swiftly generates tailored recommendations, saving users time and effort. Furthermore, we would rather provide a broad grasp of the topic and pave

the way for academics who are willing to work on deep learning for recommender systems rather than delving into implementation specifics while reviewing the publications. Our work allows scholars interested in this topic to understand main effects of utilizing deep learning techniques in recommender systems. This review study focuses on understanding the motivation of using each deep learning-based method in recommender systems. Moreover, it aims to project insights on providing deep learning-based solutions to current challenges of recommender systems. Based on (Al, 2019) paper, tracking data from the user's smartphone, which becomes the same version of unstructured data which can be sorted, segmented and produced. Most of the companies use the name "user behavior", "user satisfaction", "persona" to create a model of each user's preferences and need to grab as much attention to the specific product, item as possible. Product and Item contain very general meaning, it can be digital product, gadget, platform and even produced data as well. There are a lot of companies which collect data and Provide services relating to the same data (segments, types of people, people who desire short/long videos, products related to specific fields.), which is a much harder process for the companies with a small capacity of IT related field teams. The transformed data become an analytical tool for them to understand what type of people mostly use, watch the specific content, what they want for the next stage. The prediction model also is the most powerful field of ML which by using vector databases can understand which society strives to see/watch. Recently large language models have been integrated into recommender systems, opening up new possibilities for improving conventional recommendation processes (Di Palma et al. 2023). With their natural language processing capabilities, these models significantly improve the system's capacity to handle and analyze a wide variety of data kinds. Retrieval-augmented recommender systems, in particular, integrate the best aspects of generative and retrieval-based models to offer a strong framework for handling scenarios with data sparsity and the cold-start issue. This innovative method makes use of LLMs' knowledge of context and adaptive learning capabilities to provide highly relevant recommendations that are precisely tailored to the preferences and actions of the user. By doing so, it addresses the critical limitations of previous systems, mainly their inability to dynamically adapt to new user inputs or to efficiently handle new items without extensive retraining. The similar to the chroma DB part I assigned in my project and use of small-scale data to create embeddings for my DB and request is the paper (Varswani et al. 2018). Recurrent models especially factor computation along the symbol positions of the input and output sequences. giving the state steps to the computation period, which generates a sequence of hidden layers/states. This sequential part precludes parallelization within testing models. The project of Varswani as well used the form of embeddings to decode the dataset from traditional style to the compressed embeddings which will solve the issue of availability, scalability of data. Recommendation system is the same thing which has the structure of a fast and accessible environment.

Hotel Recommendation Assistant

What are your hotel preferences?

Search

AI assistant: Based on the detailed descriptions and considering you're looking for a hotel close to the center in Rome suitable for a couple, I would recommend the following options from the data provided:

1. "Rome - Charming Boutique Studio" (View Listing: <https://www.airbnb.com/rooms/128340>)
 - "Type": Entire rental unit
 - "Price": \$132.00 per night
 - "Location": The apartment is located in the "Borac" neighborhood, known for its narrow cobblestone alleys, craftsmen's shops, art galleries, and proximity to major archaeological sites. It's just a few minutes' walk from the Colosseum and the Roman Forum, making it an excellent central location.
 - "Amenities": This studio offers a cozy and comfortable setting with a queen bed, air conditioning, a fully equipped kitchen, and a private bath. The host, Serena, is noted for being exceptionally helpful and responsive.
 - "Host": Serena is a Superhost, ensuring a high-quality stay.
 - "Why it's suitable for couples": Its intimate size, modern comforts, and quiet courtyard setting make it ideal for couples looking for a romantic getaway. The location in the heart of Rome is perfect for exploring the city by foot.
2. "Rome Studio Rental, Colosseum" (View Listing: <https://www.airbnb.com/rooms/240602>)
 - "Type": Entire vacation home
 - "Price": \$61.00 per night
 - "Location": This studio boasts a prime location a couple of blocks from the Colosseum.

You: Give me a hotel close to the center in Rome for couple

Fig. 1. Query Example 1

Hotel Recommendation Assistant

What are your hotel preferences?

Search

AI assistant: Based on the detailed descriptions and considering you're looking for a hotel close to the center in Rome suitable for a couple, I would recommend the following options from the data provided:

1. "Rome - Charming Boutique Studio" (View Listing: <https://www.airbnb.com/rooms/128340>)
 - "Type": Entire rental unit
 - "Price": \$132.00 per night
 - "Location": The apartment is located in the "Borac" neighborhood, known for its narrow cobblestone alleys, craftsmen's shops, art galleries, and proximity to major archaeological sites. It's just a few minutes' walk from the Colosseum and the Roman Forum, making it an excellent central location.
 - "Amenities": This studio offers a cozy and comfortable setting with a queen bed, air conditioning, a fully equipped kitchen, and a private bath. The host, Serena, is noted for being exceptionally helpful and responsive.
 - "Host": Serena is a Superhost, ensuring a high-quality stay.
 - "Why it's suitable for couples": Its intimate size, modern comforts, and quiet courtyard setting make it ideal for couples looking for a romantic getaway. The location in the heart of Rome is perfect for exploring the city by foot.
2. "Rome Studio Rental, Colosseum" (View Listing: <https://www.airbnb.com/rooms/240602>)
 - "Type": Entire vacation home
 - "Price": \$61.00 per night
 - "Location": This studio boasts a prime location a couple of blocks from the Colosseum.

You: Give me a hotel close to the center in Rome for couple

Fig. 2. Query Example 2

V. CONCLUSION AND FUTURE USE

To conclude this project aimed at creating a highly personalized recommendation system based on the specific preferences that a user will give as an input. The model currently works on a 10-row dataset, that is stored in vector db, in our case chromaDB, and contains complete information about every hotel including both numeric data like price, id, number of bedrooms and textual data like room type, property type, hotel description, host description, neighborhood description, reviews, amenities. To make the retrieval from the vector db smooth langchain is used and the main model works with retrieval augmented generation, that is it takes the data from vector db, gets prompt for gpt model and gets user input and having all the data and logic to work generated highly personalized recommendation of hotels. Currently the data is narrowed down too much and in the future a lot of work can be done to find a way to store more information and from multiple cities/countries to make the recommendation more accurate and applicable worldwide. Another objective would be to not use ready made data but to have a scraping script that will update every day and only in terms of "recommending good hotels based on prompt" overperform chat gpt and other booking websites on the internet like Booking.com and Airbnb.com.

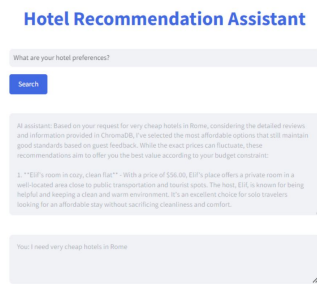


Fig. 3. Enter Caption

in neural information processing systems (pp. 5998-6008).https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee2Paper.pdf

VI. REFERENCES

- [1] Al, A. D. E. a. a. D. E. (2019). An analytical study and visualisation of human activity and content-based recommendation system by applying ML automation. *International Journal of Mechanical and Production Engineering Research and Development*, 9(3), 75–88. <https://pure.eur.nl/ws/portalfiles/portal/36755878/26715561832658.IJMPERDJUN20198.pdf>
- Batmaz, Z., Yurekli, A., Bilge, A., Kaleli, C. (2018). *A review on deep learning for recommender systems: challenges and remedies*. *Artificial Intelligence Review*, 52, 1–37. Available at : https://daiwk.github.io/assets/Batmaz2018_Article_A_Review_On_Deep_Learning_For_Recomm.pdf
- [3] Hsain, A., Housni, H. E. (2024, March 16). Large language model-powered chatbots for internationalizing student support in higher education. *arXiv.org*. <https://arxiv.org/abs/2403.14702>
- [4] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... Polosukhin, I. (2017). Attention is all you need. *Nature Communications*, 8, Article 14049.
- [5] Deldjoo, Y., He, Z., McAuley, J., Korikov, A., Sanner, S., Ramisa, A., Vidal, R., Sathiamoorthy, M., Kasirzadeh, A., Milano, S. (2024, March 31). A review of modern recommender systems using Generative Models (Gen-RECSYS). *arXiv.org*. <https://arxiv.org/abs/2404.00579>
- [6] Zheng, G., Terry, J. M., Belgrader, P., Ryvkin, P., Bent, Z., Wilson, R. J., Ziraldo, S. B., Wheeler, T. D., McDermott, G. P., Zhu, J., Gregory, M., Shuga, J., Montesclaros, L., Underwood, J. G., Masquelier, D. A., Nishimura, S. Y., Schnall-Levin, M., Wyatt, P. W., Hindson, C. M., . . . Bielas, J. H. (2017). Massively parallel digital transcriptional profiling of single cells. *Nature Communications*, 8(1). <https://doi.org/10.1038/ncomms14049>
- [7] Di Palma, D. (2023). Retrieval-augmented Recommender System: Enhancing Recommender Systems with Large Language Models. In *Proceedings of the Seventeenth ACM Conference on Recommender Systems (RecSys '23)*, ACM, New York, NY, USA, 8. https://www.researchgate.net/publication/373931475_Retrieval-augmented_Recommender_System_Enhancing_Recommender_Systems_with_Large_Language_Models
- [8] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., Polosukhin, I. (2017). Attention is all you need. In *Advances*