SAPIENZA
UNIVERSITÀ DI ROMA

Collaborative Filtering Recommender System for Tourism Social Network

Faculty of Information Engineering, Informatics, and Statistics

Corso di Laurea Magistrale in Data Science

Candidate

Mateus De Souza
ID number 1927607

Thesis Advisor

Prof. Chatzigiannakis Ioannis

Co-Advisor

Dr. Maryam Kamal

Academic Year 2021/2022

Thesis defended on 24th March of 2023
in front of a Board of Examiners composed by:

Prof. Brutti Pierpaolo (chairman)
Prof. Baiocchi Andrea
Prof. Chatzigiannakis Ioannis
Prof. Galasso Fabio
Prof. Lembo Domenico
Prof. Marcucci Juri
Prof. Scardapane Simone

Collaborative Filtering Recommender System for Tourism Social Network
Master's thesis. Sapienza – University of Rome

This thesis has been typeset by LaTeX and the Sapthesis class.

Author's email: mateus.de.souza@accenture.com

Abstract

Recommender systems are tools that interact with complex information spaces to provide a suggestion of new content based on the interest of the user. They provide a personalized view of the information, so that the feedback to a user is personal and individualized. For the matter of the development of a recommender engine, techniques of artificial intelligence, such as machine learning, neural networks, deep-learning, data-mining, user-modeling, topic-modeling and constraint satisfaction, among others, are used to give the most reliable recommendation for users. Personalized recommendations are an important part of many online e-commerce applications and social-network interactions. They are responsible for a tremendous increase in the revenue of online companies and the engagement of users in social network platforms. For this reason, the present thesis provides the development of a recommender system for the tourism social network Kuriu, to enhance the user experience on the platform.

Contents

Chapter 1

Introduction

1.1 Research Motivation

Based on the development of different mathematical tools, the process of recommending products to clients, can be done by different approaches. In order to give a coherent suggestion to a given user of the touristic destination, it is significant to take in consideration all the previous information of the customer, not only regarding about his active choice of places, but also how he interacts inside the network.

Looking for the development of a trustful recommender system, an alternative algorithm is proposed in this document. The goal of the author was the creation of an engine to support the experience of traveling and to make the decision process of the platform users easier.

In the following document the research done describes the mentioned engine, its difficulties, and the mechanism behind it. Also, the particularities of the data used for the development were mentioned, together with the evaluation parameters focused on the coherence and the diversity.

1.2 Document Outline

This document is divided into 8 chapters. In the chapter 1 the motivation behind the development of a tourism recommender engine is exposed.

To afford a good comprehension of the algorithm used and its implications in the world of online tourism platforms, the next 3 chapters gave a briefly introduction of the engine behind the algorithm and its real world applications. As follows, in the chapter 2 the theoretical part of search engines was described, in 3 the different architectures available and real examples of applications were discussed, and, finally, in the chapter 4 the algorithms used to extract information from text data.

The practical part is divided in three parts. In the chapter 5 the data used for developing, training and evaluating the algorithm is described. The algorithm itself is described in the chapter 6. And the exposition of the results and conclusion are respectively in the chapters 7.1.3 and 8.

Chapter 2

Recommender systems

2.1 Preliminaries and Basic Concepts

During the last decades, the development of the internet and the online world have made the recommender systems one of the biggest impacts in our daily life. With the rise of the big web services, specially in the area of e-commerce and online advertisement, the recommender systems are something unavoidable.

Explaining hastily, a recommender system is an algorithm which suggests relevant items to users. For example, some product, depending on the interests of the user, some movie, depending on the past movies watched by the user, a new place to visit, based on the last tickets bought by the user and so on. [18]

In the case of industries, the recommender system is critical, because they can generate huge income advantage when they are well engined and could be critical to the competitiveness of the industry in a given market. However, the goal of a recommender is the same, there are different approaches to the development of one.

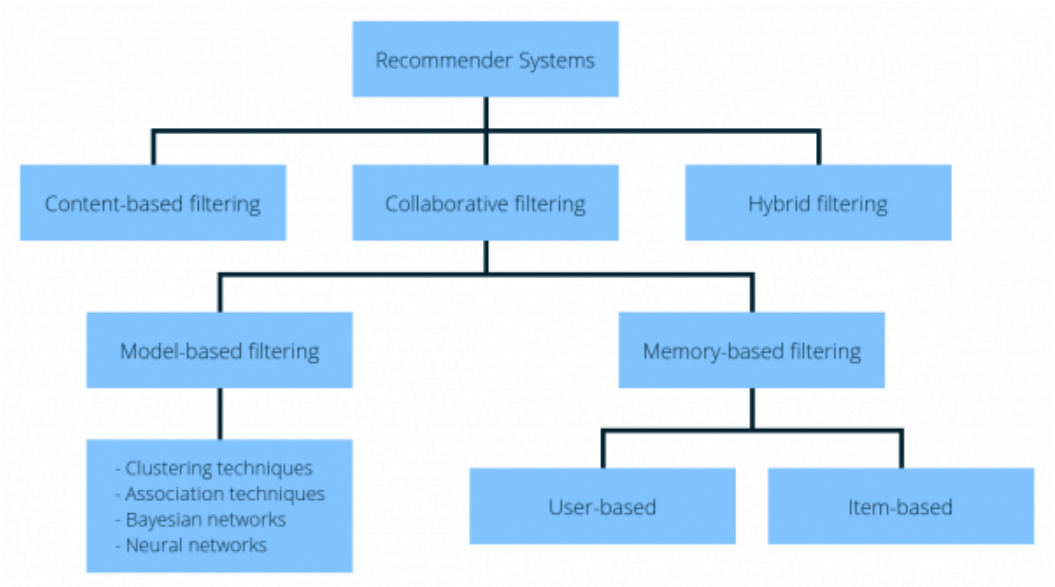


Figure 2.1. Different categories of Recommender Systems

So, as the importance of those algorithms is undeniable for the era of online world, it is necessary, for any online business or social network, to study and develop

better and more accurate recommender systems, in order to be in evidence on the online market. [14]

2.2 Collaborative Filtering

Collaborative filtering is a method of recommender system that is based on the past interactions between users and items in order to produce the recommendation of new products. [23]

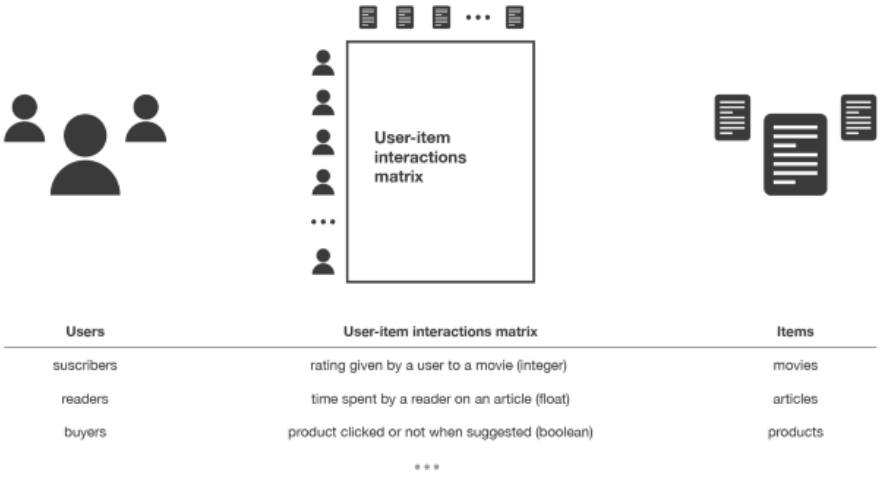


Figure 2.2. Idea of a Collaborative Filtering [18]

Then, the idea of the collaborative filter is that the data, storage in the "user-item interactions matrix", is enough to create similar profiles for the users/items and make predictions based on those estimated similarities.

The class of collaborative filtering algorithms is divided into memory based and model based approaches. The memory based ones uses the interactions between users-items, assuming no model for it, and searching for proximity users/items around a specific user/item target and recommend the most popular items among those in the proximity. In contrast to the memory based algorithm, the model based approach, assumes the existence of a model that explains the interaction between user and item and tries to use this model to predict new recommendations for the users. The selection of the model creates the different types of model-based algorithms, such as clusters, deep-learning models, neural-networks, matrix decomposition models and other types of classification algorithms.

The advantage of collaborative approaches is that they are not data exclusively, so as they don't require any specific information about the users of the items they can be flexible to adjust for numerous situations. Furthermore, the more user-item interactions, the better it is for the algorithm to give more accurate recommendations because it gives more margin for the learning process.

As collaborative filters only consider past interactions to make the recommendations, they suffer from a "cold start problem", where it is hard to recommend something to a new user, or relate a new item to be recommended for a user, because they have few interactions to be efficiently handled. To avoid this problem, we have few techniques that will be discussed further in this thesis.

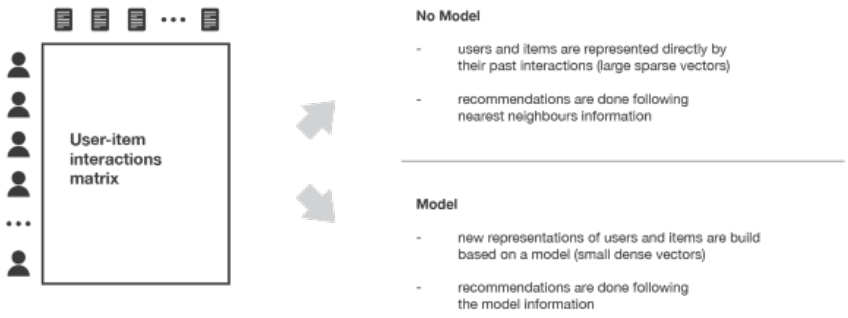


Figure 2.3. Different types of Collaborative Filtering [18]

2.2.1 Memory based collaborative approaches

As briefly mentioned before, the memory based approach only use information from the user-item interaction matrix, assuming no model to produce new recommendation.

In the case of the user-user memory based approach, we have that the idea is to identify users with most similar "profile" in order to suggest items for those similar users. This method is also known as "user-centered" because it is a nearest neighbors approach towards users.

In the example of a user-user, the idea is to analyze the interaction matrix and use some measure of similarity in the rows of the matrix to create the similarity between all users, after this step a k-nearest-neighbours is used to suggest the most popular items among the items not seeing already by our reference user. [16]

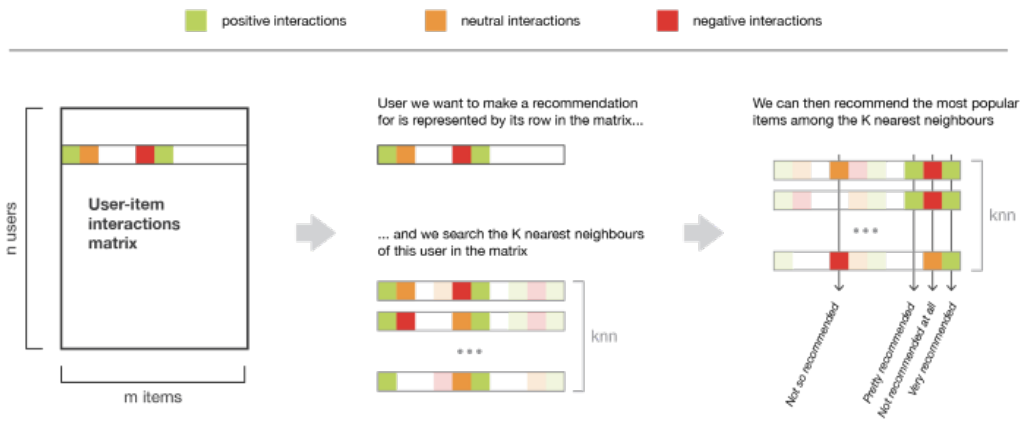


Figure 2.4. Memory based collaborative filter user-user approach [18]

In this approach, one thing to be considered to avoid problems in the recommendation is the number of interactions of the user, because this could lead to a very high similarity when the user hasn't being active enough to be considered as a valid candidate.

In the case of the item-item memory based approach, the idea of finding similar

items based on how the users have interacted with them. Important to know that, for the recommendation system to be a good tool, the interaction of those users need to be positive to be valid. In this method, the same idea of distances between similar items is applicable, so we call it "item-centered".

Assuming a recommendation to a given user in the item-item memory based method, we first assume the item that the user liked the most and calculate a similarity metric for all the items in the interaction matrix. Once all the similarities are calculated, we apply the same k-nearest-neighbours for the given item and recommend it to the user.

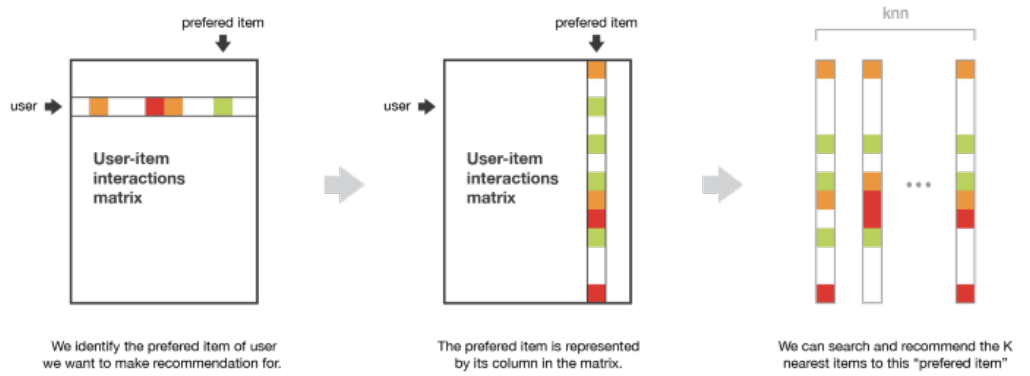


Figure 2.5. Memory based collaborative filter item-item approach [18]

It is important to notice that in this approach is easy to recommend other items to users by not choosing the favorite one for each user, but instead, choosing another liking item and follow the calculation using this new one as a reference.

To conclude, in the figure below we can see the bigger differences in the two different approaches for the n memory based collaborative filters:

Figure 2.6. Differences in memory based collaborative filters [18]

2.2.2 Model based collaborative approaches

In model based collaborative approaches, the matrix of interactions between users and items is supposed to be represented by a model, where the prediction of new items for users can be made by following this hypothetical model. The most common technique for model based collaborative filtering is to use a factorization of the interaction matrix in a user-factor matrix and a factor-item matrix that describe the whole model base on the values of those factors and which it is used for the recommendations. [27]

As the method itself rely on a hypothetical model, it is clear that the recommendation maybe differ from the "best possible recommendation for a given user", but as a model, it is fair enough to give a recommendation close to the best one.

2.3 Content-Based

Unlike collaborative methods, the content based approaches use additional information about the users/items to create recommendations. It is an approach assuming a model only the data itself, to create recommendation on the similarity of other features present in the user interaction with the system or about the user himself. [1]

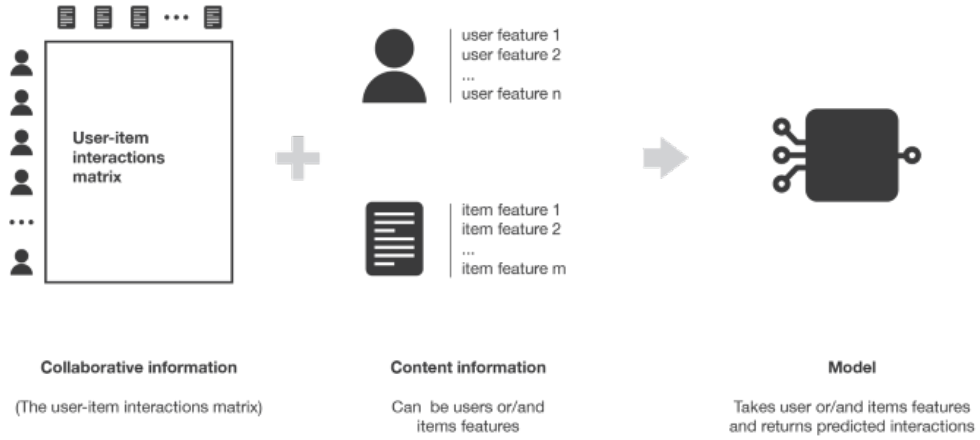


Figure 2.7. Idea of Content-Based [18]

The advantage of the content-based algorithm is that it doesn't suffer for the "cold start" problem, because the new users/items can be described and classified by their characteristics (features of the data).

2.3.1 Probabilistic Model

In the case of a content-based recommender system, the prediction can be obtained by estimating $\frac{P(A|B)}{P(\overline{A}|B)}$ using the probability distributions based on the Bayes Rule.
Applying the Bayes Rule:

$$P(A|B) = P(A)\frac{P(B|A)}{P(B)}$$

$$P(\overline{A}|B) = P(\overline{A})\frac{P(B|\overline{A})}{P(B)}$$

So, we have that:

$$\frac{P(A|B)}{P(\overline{A}|B)} = \frac{P(A)P(B|A)}{P(\overline{A})P(B|\overline{A})}$$

Where:
$P(A)$ and $P(\overline{A}) = (1 - P(A))$ are the prior-distributions computed using the data.
$P(B|A)$ and $P(B|\overline{A})$ are likelihoods assumed to follow Gaussian distributions, with parameters to be determined also from data.

So, the idea of the probabilistic model is to train a Bayesian classifier in order to predict items for users. It is important to mention that the parameters to be estimated depend only on data interaction related to the considered item. [18]

2.3.2 Vector Space Model

The recommender system in the vector space approach is a problem of optimization, where a linear regression is done in a user-centered environment.

$$X_i = \arg\min_{X_i} \frac{1}{2} \sum_{i,j \in E} [X_i Y_j^T - M_{i,j}]^2 + \frac{\lambda}{2}(\sum_k X_{i,k}^2)$$

This approach differs from the collaborative filtering because, in the matrix decomposition, the algorithm needs to learn latent representations for both users and items, whereas the content-based vector space method builds a model upon human defined features. [18]

2.4 Hybrid

Both methods, collaborative filtering and contend-based, have their weak points. So, a good way to approach the development of a recommendation system is to use a hybrid setup, where both methods are used together to achieve a better result. [8]

There are many approaches when building a hybrid recommendation system, but the main ones are Weighted, Switching, Mixed, Feature Combination, Feature Augmentation, Cascade and Meta-Level.

2.4.1 Weighted

In a weighted recommendation system, the goal is to use different models to interpret the date and do a linear combination of the output of those models, where the output of the prediction is a combination of the output of the different methods. [8]

2.4.2 Switching

The switching hybrid model works by introducing an additional layer on the recommendation model, where it chooses the right model to be applied on the data, based on the user profile or other features. [8]

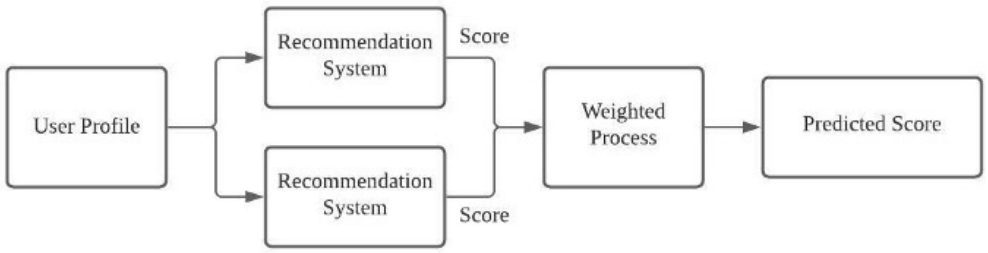


Figure 2.8. Weighted Hybrid Recommendation System [8]

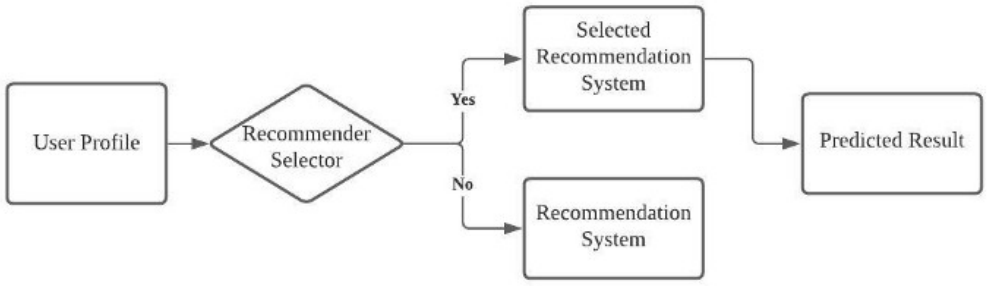


Figure 2.9. Switching Recommendation System [8]

2.4.3 Mixed

The mixed model is a combination between the switching and the weighted model, where the candidates are separated in different sets first and after are trained in different approaches, with the final output being a linear combination of the output from the different methods. This method is able to make many recommendations simultaneously with a target on performance enhancing. [8]

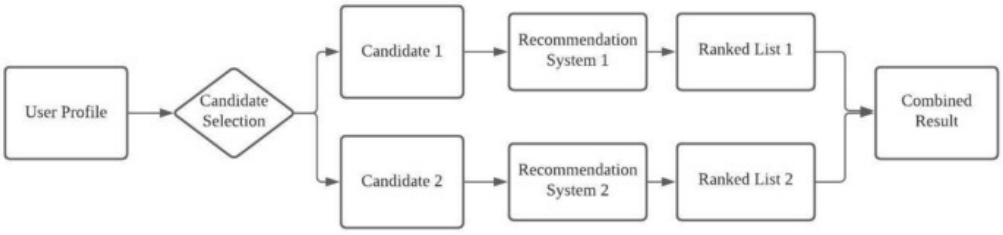


Figure 2.10. Mixed Recommendation System [8]

2.4.4 Feature Combination

The idea behind the feature combination is to use an auxiliary recommendation system to produce some features which are propagated into the main recommendation system, where the final output of the recommendation is produced. [8]

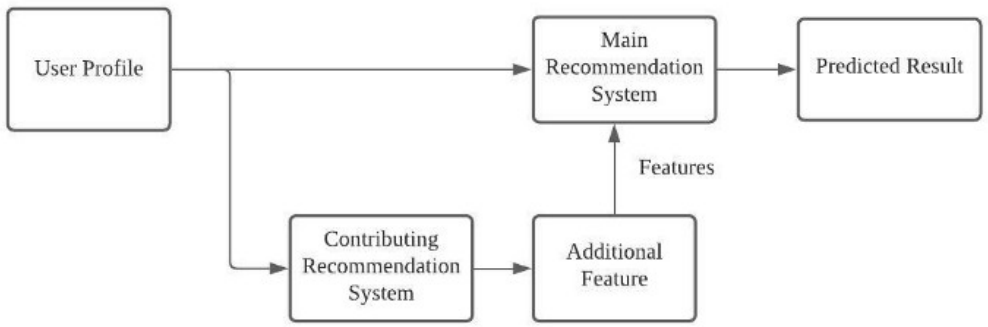


Figure 2.11. Feature Combination Recommendation System [8]

2.4.5 Feature Augmentation

In the feature augmentation, we have a contributing recommendation model generating a classification for the user/item profile, which is further used to produce the final prediction with a main recommender system. The advantage of the feature augmentation is that it can improve the recommendation model without changing it structure. [8]

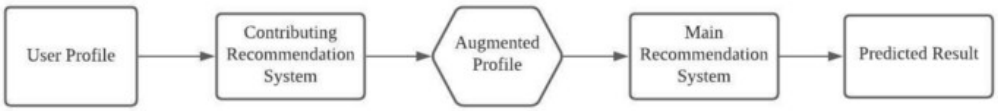


Figure 2.12. Feature Augmentation Recommendation System [8]

2.4.6 Cascade

In practice, cascade models are used to solve some minor issues from the output of a main recommendation system. For example, the main recommendation system produces an output, where it is refined by a second recommendation system, responsible to deal with some missing data issue. [8]

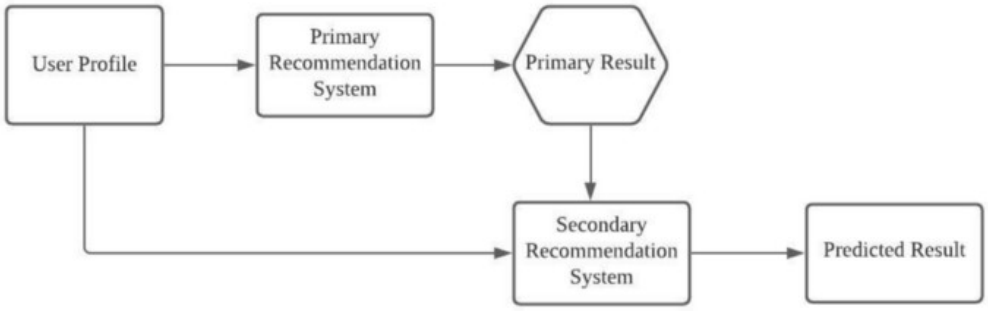


Figure 2.13. Cascade Recommendation System [8]

2.4.7 Meta-Level

The meta-level approach is similar to the feature augmentation, but with the difference on the dataset, because in the meta-level the original dataset is learned by a contributing model and then is passed to be processed in the main recommendation system. [8]

2.5 Evaluation of a recommender system

To be able to decide which algorithm fits the best for a given situation, we need to have some tool to evaluate the performance of the recommendation system. In the case of the recommender systems, the evaluation can be made by a well-defined metric or by the human judgement. [18]

2.5.1 Metrics based evaluation

For the example of a recommender system model based, the numeric values of probabilities can be used inside an error measurement metric in the way that the model is trained in part of the dataset and tested in the remaining part. In a classification approach, where some threshold is used to binarize the data, other measures can also be evaluated, such as accuracy, precision, and the recall in the output of the model.

Finally, in the case of a memory based collaborative filtering, such as a “user-user” or an “item-item” interaction, a different metric can be used but also taking in consideration the k-nearest-neighbours behind the algorithm. The idea is to see how much an item already associated to a user can be recommended to this user, so in that case we can have some precision metric for the algorithm.

2.5.2 Human based evaluation

When designing a recommender system, we want it to be also a bit unpredictable and with a certain explainability of why it is recommending something to the user.

The explainability of a recommender is a key point to the success of the algorithm, because in the case of a recommendation without the understanding why is it relevant to a given user would create a loose of confidence of the user in the recommender. So as an alternative to this phenomenon, it is interesting to support the user with some feedbacks of why a certain item is recommended to him.

So, as diversity and explainability are intrinsically difficult to evaluate, one good alternative would be the human evaluation of the recommender. This makes the evaluation a real time task, where depending on the action of the user to a given recommendation, the algorithm is self evaluated to improve the algorithm and would give an on-time feedback of the whole recommendation engine.

Chapter 3

Commercially Adopted Recommender Systems

3.1 Deep Learning Based Models

Traditionally, recommender system are based on clustering, KNN and matrix factorization. However, as the deep learning is showing success in different areas, the tendency came also for the recommendation systems. In fact, big tech companies such as YouTube, Amazon, Netflix, TripAdvisor, and others, choose the implementation of deep learning algorithms in combination with the traditional methods because of their efficiency. [6]

3.2 TripAdvisor Recommender Systems

TripAdvisor, the largest travelling website nowadays, provides a platform for billions of users to review, book and search trips and experiences around the world. Their volume of experiences is around the number of 160000, which leads to a huge number of information regarding clients.

As the number of available experiences grow rapidly in the platform, it came out the need to produce personalized recommendations for the users, in order to increase their satisfaction and provide travelers an easy way to find relevant experiences. The problem with the amount of information and the personalized recommendation is that, to process all this information, TripAdvisor needs to use Deep Learning models in its recommender system. [21]

The deep learning recommender system of TripAdvisor is divided in three major components: data collection, entity embeddings, and the neural network architecture.

3.2.1 Data Collection

The collection of the data is important for a supervised learning setting, because it defines the way the model is going to be trained. In the algorithm of TripAdvisor, the collection of data relies on the interaction of the user with the buttons “check availability” and “booking”, where different weights are assigned to each one.

So, in the end, the data collection is a full view over the navigation of the user on the website of TripAdvisor giving different weights for different inputs given by the user to different experiences.

3.2.2 Entity Embeddings

The embedding is a popular technique to convert words and phrases into vectors, which can be processed. It is a technique adopted in recommender system as a form of learning the representation of items and users.

In the following model, each entity is represented as a 100-dimensional vector within the same embedding space, containing points of interests and bookable experiences. First, the embeddings are trained using the "StarSpace" package of Facebook AI Research on the page view logs. After the pre-training, the embeddings encoded entities such as location and category information. The pre-trained data is passed to other downstream tasks (sort orders and landing page recommendations) in order to be initialized for the consumption of the RFY model. [21]

It is important to notice that, this initialization with pre-trained weights and then fine-tune of the embeddings gave a better performance to the model in comparison with a random initialization.

3.2.3 Neural Network Architecture

As represented in the figure below, the RFY model starts with an aggregation of the user browsing history and taking the weighted average of the 100-dimensional item embeddings. After we have two fully connected layers with the final softmax output on 64,000 class probabilities, where each of those represents one experience that can be recommended. [21]

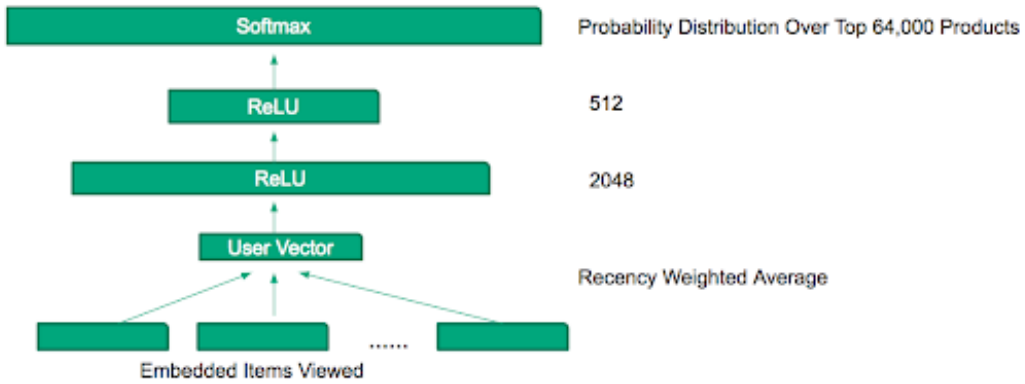


Figure 3.1. RFY model of Trip Advisor [21]

We use the exponential recency weighted average formula to aggregate the user browsing history:

$$x_n = \sum_{i=1}^{n} w_i x_i$$

$$w_i = \alpha(1-\alpha)^{n-1}$$

The principal assumption of the RFY model is that the most recent browsing data contributes the most to the prediction of the next action of the user. So, a simple model is able to enhance the speed of the prediction, compromising the accuracy for not increasing the number of neurons before the softmax.

3.3 Airbnb Experience Recommender Systems

Airbnb Experience is a two-sided travel e-commerce platform where users can host tours or activities and book them as a guest. As recommender systems are dependent on abundant user listing interactions, the Airbnb has a weak point in respect of the application of traditional recommender systems approaches, because it is an emerging business and it is not full of data yet.

Instead of wait for data to be accumulated, the Airbnb Experience works on listing and estimating the guest preferences with limited data, by extending the knowledge graph and utilizing location features, such as city-specific concepts, which enables to better characterize inventories.

Then, collaborative filtering, graphical models, or deep learning methods can then be applied by using both user-item interaction data and attributes of users. However, due to the limited amount of user-item interaction data, and the exponential increase of feature space that can be caused by categorical features, classic learning methods can be prone to overfitting. [26]

3.4 Amazon Recommender Systems

3.4.1 Amazon: E-commerce

The Amazon recommender system is based on an item-to-item collaborative filter, where the recommendation algorithm review the visitor's recent purchase history and, for each purchase, pulled up a list of related items. The items which appeared repeatedly across the lists were genuine candidates for a recommendation for the user, but also the candidates which appeared fewer, would be considered as a recommendation but with a different weight. [**?**]

The particularity of the algorithm of Amazon relies on the relatedness metric used. It is a conditional probability metric where, item B is related to item A if purchasers of A are more likely to buy B than the average Amazon customer is. The greater the difference in probability, the greater the items' relatedness. It is important to notice that the heavy buyers are not considered for the calculation of the metric, because of the tendency to introduce bias on the likelihood. [11]

3.4.2 Amazon Prime: Streaming

Amazon, one of the biggest e-commerce platform nowadays, also has a streaming platform called Prime, which uses a model-based recommender system. As the recommendation is often modeled as a matrix completion problem, where a big matrix is writing by a decomposition of a smaller matrix, the Amazon recommender system uses deep learning in order to solve the problem of matrix completion. [11]

The deep neural network contains a lot of processing nodes, arranged into layers where the data processed and passed through different layers until reach the final layer, that output the result of the computation. Once the deep neural network is ready, it needs to be trained with samples of inputs and outputs, which are responsible to readjust the settings of the neural network until the average discrepancy between the top layer output and the desirable output result is minimum. [27]

In the case of prime, the autoencoder is used to calculate the matrix completion problem. It is a neural network trained to output the same data it takes as input, with, in-between of the input and output layers, a bottleneck, a layer with relatively few nodes — in this case, only 100, versus tens of thousands of input and output nodes. The peculiarity of the model of Prime, is that the autoencoder need to be

trained correctly, in other words, to be fed with the input data in chronologically order, because the age of the input film, when used for training, was an important feature when it comes to a movie recommender. [11]

3.5 Netflix Recommender Systems

Netflix is a subscription service that offers movies, documentaries, and series for its clients. To help find shows and new movies for the users, Netflix developed a recommender system able to enhance the variety of interests of the customer of its services. [15]

When a client accesses the Netflix platform, the recommender system estimates the likelihood that the user will watch a particular title based on factors as:

- Interactions with the service, such as viewing history and title rating.
- Members with similar tastes and preferences.
- Genre, category, actors, release year and other information about the title.
- Time of the day the user uses the platform.
- The device used to access the platform.
- How long the user interact with the service.

This information is used as input of the algorithm of the recommender system, but it is also known that Netflix avoid using personal information such as age and gender as part of its decision-making process.

To avoid the problem of a "cold start", once a user add a new account to his subscription, he is asked to choose some titles that may be interesting. This is used to start the process of recommendation and as long as he uses the platform, the algorithm gets improved and new titles are suggested, and it creates an environment of real-time feedback for the recommender. [15]

By the time, the whole architecture of Netflix homepage is configured to adjust to user preferences so that, every row and category is ranked and personalized to give the user the best experience of joy in the platform.

3.6 Online and Offline Evaluation Parameters

The key point of a successful recommendation system rely on the choice of evaluation metric for it. There are three main methods of evaluation: offline evaluations, online evaluations, and user studies. But for the scope of this project we will focus on the offline and the online ones. As the evaluation show that results from offline evaluations sometimes contradict results from online evaluations, we will further discuss their efficiency and the pros and cons of each of those. [5]

3.6.1 Online Evaluation Parameters

Online evaluations come from online advertisement and e-commerce environment. Their main focus is to measure the acceptance rates of the recommendations in real-time, by mostly considering the ration of clicked recommendations (click-through rate) or ratio of download/bought items. The main idea behind those metrics is that they have a positive correlation with the user satisfaction. [5]

The assumption is that when the user interacts with a recommendation item, it shows that he is satisfied with what was recommended for him and this can be doing in an online environment where the interaction is captured by real time actions from the user.

3.6.2 Offline Evaluation Parameters

Offline evaluation measures the accuracy of a recommender based on a ground-truth. The goal here is to give a measure by the use of metrics such as accuracy, precision, recall, F-measure, mean reciprocal rank (MRR), normalized discounted cumulative gain (nDCG), mean absolute error, and root-mean-square error in order to evaluate aspects such as novelty or serendipity of recommendations. [5]

As the ground-truth plays an important role in the offline evaluation, we have two types of datasets that can serve as ground-truth:

- "Explicit ground-truths" refers to the situation where the information of how the users liked certain items is available. So the evaluation relies on the fact that by taking out some data, the recommend system alone would be able to predict it and then by comparing the real value with the prediction, the recommend system is evaluated. [5]
- "Inferred ground-truths" are based on a personal collection of items, where the recommendation is compared with. The assumption relies on the fact that the prediction to be valid should be inside the collection, so the more recommended items are inside the inferred ground-truth collection, the better is the system. [5]

Chapter 4

Natural Language Processing

Natural Language processing is a field of Artificial Intelligence and Linguistics, with the aim of making computers understand the statements or words written or spoken by humans.

Actually, the approaches to NLP are based on machine learning, where patterns in natural language data are analyzed, and using them, the algorithm improves the computer program's language comprehension. [12]

Nowadays, the NLP is applied in chatbots, smartphones, search engines, banking, translation and many other business applications. So, because of this widely application, it is a field in constant development and study.

4.1 Topic Modeling

Topic modeling is an unsupervised artificial intelligence approach that reads documents, search for words and phrases patterns, and automatically clusters those terms in a way which best represents the set. It is called an "unsupervised" learning method because it doesn't require a preexisting training data categorized by humans.

This method extract needed attributes from a bag of words, because in NLP, each word in the corpus is treated as a feature. As a result, the feature reductions allow focusing on the relevant material of the text, rather than waste time sifting through all the data.

The main point of this method is to give a classification output through an unsupervised learning atmosphere, where each text is classified in a bucket defined by the topic created by the method. It is different for a classical supervised learning classification approach because of the use of transformers, which are responsible for labeling the elements of the text data before the topic modeling. [24]

4.2 Top2Vec

The algorithm of Top2Vec starts with the use of Doc2Vec to generate a semantic space, a vector space where the distance among the vector indicates the semantic similarity between them. [9]

About Doc2Vec, it is a high level modification of Word2Vec, which create document/sentence/paragraph embedding. Unlike the Word2Vec, the Doc2Vec incorporates a paragraph vector along with word vectors during training phase and can be trained in two different ways:

- Paragraph Vector with Distributed Memory: Given a paragraph vector and context vector, it predicts target word.
- Distributed Bag of Words: Given a paragraph vector, it predicts the context words.

Considering those two methods of training for the Doc2Vec, as a matter of performance enhancing, the model Top2Vec utilizes the Distributed Bag of Words version of the Doc2Vec method to create the semantic space.

Unlike the LDA algorithm, where topics are sampled from a discrete space, the semantic space in Top2Vec is a continuous representation of topics consisting of word and document vectors. Those jointly embedded word and document in the semantic space are so interpreted as, areas with high concentration of documents can be thought of having similar topics and can be best represented by nearby embedded words. [19]

The search for such dense areas in the semantic space face a first problem in the use of traditional clustering methods because of the high dimensionality of the space. So, to avoid this issue, first a dimensionality reduction process is applied in the semantic space, normally using the Uniform Manifold Approximation and Projection for Dimension Reduction (UMAP), because it is good at preserving local and global structure, and then, the HDBSCAN algorithm is used for the clustering. [3]

So, after the UMAP and HDBSCAN identify the document dense clusters, Topic vectors are simply calculated as the centroid of these clusters.

The point of Top2Vec is that, as the document and word vector are jointly represented in the semantic space, therefore, once the topic is identified, it is easy to find the nearby word vectors that best represent the topic. So, in the end we have a word distribution for every topic created, and each of the documents can be assigned to a topic. [4]

Another important point about the Top2Vec algorithm is that it doesn't require the remove of stop words before the application of the algorithm. This happening because such words may appear in almost all documents, so they will be equidistant for all topics and shall not appear as nearest words to any topic.

Also, as the algorithm gives a continuous representation for the topics in the semantic space, this allows the reduction of the number of topics by taking a weighted arithmetic mean of the topic vector of the smallest topic and its nearest topic vector, each weighted by their topic size. After each merge, the topic sizes are recalculated for each topic. [10]

4.3 Recommender Systems with NLP

Recommender systems (RS) have evolved into a fundamental tool for helping users make informed decisions and choices, especially in the era of big data in which customers have to make choices from many products and services. A lot of RS models and techniques have been proposed and most of them have achieved great success. Among them, the content-based RS and collaborative filtering RS are two representative ones. Their efficacy has been demonstrated by both research and industry communities. [13]

The advent of Big Data, makes the recommendation system deal with different types of data structures. The use of unstructured data, such as text data, creates difficult in the processing of information to feed a recommender engine. [20]

Because of this issue, the Natural Language Processing is used to, first make unstructured text data familiar to the machine, and capable to be processed and

interpreted, so that the recommender can be used to generate recommendations to the users. So, the overview of the process is, to make the machine able to understand the unstructured data for, after, be able to use the classical recommender algorithms to recommend products for users. [22]

4.4 Methods of Evaluation for Topic Modelling

Statistical top modeling is a tool used for analyzing large unstructured text collections. To date, however, there have not been any specifically addressing on the evaluation of topic models.

The evaluation of a model is an important issue, because the unsupervised nature of topic models makes model selection difficult. For some applications, the performance can be evaluated, but there is not an universal method, that measures the generalization capability of a topic modeling as a good choice of model. [10]

Many evaluation metrics have been defined to evaluate the effectiveness of retrieval and search result in diversification systems. However, it is often unclear which evaluation metric should be used to analyze the performance of retrieval systems given a specific task. [2]

For the matter of controlling the performance of the model, the coherence of topics and the diversity are pointed as measurements of good topic modeling. For this purpose, in the present work the Inverted Rank-Biased Overlap and the Topic Diversity measures are using to quantify the diversity of the topics, and the Coherence metrics from Gensim are used to evaluate the coherence of the topics.

Interpretation of the meatrics:

- Topic Diversity: Is defined here as the percentage of unique words in the top 25 words of all topics, where, a diversity close to 0 represents a redundant topic, and those close to 1 indicate more varied topics. [7]
- IRBO: Is a measure of disjointedness between topics weighted on word rankings, based on the top-N words. So, as in the Topic Diversity, the higher these metrics are, the better. [7]
- Coherence: It is a number that represents the overall topic's interpretability and is used to assess the topic's quality. What a Topic Coherence Metric assesses is how well a topic is 'supported' by a text set (called reference corpus). It uses statistics and probabilities drawn from the reference corpus, especially focused on the word's context, to give a coherence score to a topic. [17]
- Hit Rate: In a recommender framework, the hit ration is the fraction of items for which the correct answer is included in the recommendation list. As one can see, the bigger the recommendation list, the bigger the ratio becomes, because of the higher chance that the correct answer is included in the recommendation list. [25]

Chapter 5

Datasets

5.1 Easy Tour Dataset

The development of the project was first based on the dataset of Easy Tour. The data gathered was used as the benchmark for the algorithm structure, and it is divided in two collections of data:

5.1.1 Easy Tour Reviews

The reviews data contains the writing review of each user to a given attraction. It is important to know that, in the Easy Tour Dataset, the users, the location, and the reviews have an ID, which was crucial for the development of the algorithm.

	id	userId	title	description	tags	categories	viewCounter	placeId	city	province	region	state
0	62349ff2272ae5672fbc3135	632	Big Buddha, Phuket TH	Phuket è conosciuta per essere la meta predile...	['Phuket', 'Buddha', 'bianco', 'Panorama', 'bu...	['Viaggi']	18	ChIJOSmXz2AvUDARXCxlBBs7y38	Tambon Karon	Amphoe Mueang Phuket	Chang Wat Phuket	Thailandia
1	6234aa65272ae5672fbc34c2	632	Wat Chalong, il tempio più venerato di Phuket TH	Phuket non è solo spiagge, ci sono anche altri...	['tempio', 'Watchalong', 'TempioBuddista', 'bu...	['Viaggi']	16	ChIJYSMtg7svUDARtkIKwhnTa48	Tambon Chalong	Amphoe Mueang Phuket	Chang Wat Phuket	Thailandia
2	620aa536ff8ae67f37cafb06	442	Accarezzare le tigri	So che dietro queste cose c'è spesso del marci...	['tigri', 'TigerKingdomPhuket', 'isola', 'anim...	['Viaggi']	426	ChIJ0xtyOmgwUDARKUcz6kOyIEQ	Phuket	Kathu District	Phuket	Thailandia
3	624e5982ec9f8559ff529ce8	571	Yangon, la capitale storica del Myanmar	Probabilmente Yangon sarà il punto d'inizio se...	['Myanmar', 'Birmania', 'Yangon', 'città', 'As...	['Viaggi']	6	ChIJaxk-Ip6UwTARtAsI-HHS-1Y	Yangon	NaN	Regione di Yangon	Myanmar (Birmania)
4	620abb33ff8ae67f37cafe3c	442	Freedom beach	Forse la spiaggia più bella di tutta l'isola d...	['Freedombeach', 'spiaggia', 'phuket', 'thaila...	['Viaggi']	424	ChIJBVMebEoIUDARduwCCVdtNxE	Tambon Patong	Amphoe Kathu	Chang Wat Phuket	Thailandia
...	...	...	...	...	...	...	...	...	...	...	...	...
5719	6213b3460f41f86438b07d9c	429	Machu Picchu, meraviglia del mondo	È l'alba, ma la realtà è che in questo viaggio...	['Perù', 'MachuPicchu', 'meraviglia', 'meravig...	['Viaggi']	47	ChIJVVVVlV-abZERJxqgpA43EDo	NaN	Provincia di Urubamba	Regione di Cusco	Perù
5720	61dd867f9c823e129b1f2cc2	429	I segreti di Moray	Sembra la prova schiacciante dell'esistenza de...	['Moray', 'terrazze', 'Incas', 'Perù', 'agrico...	['Arta aperta']	95	ChIJyZGitKXDbZERg4qm1pVfvBQ	Maras	Urubamba	Cuzco	Perù
5721	61e0146111b3bf4d3185ad5d	427	Machu Picchu, tra storia e meraviglia	Sono passati dieci (!) anni dalla mia esperien...	['Machupicchu', 'Perù', 'archeologia', 'storia...	['Cultura']	63	ChIJNf_BulOabZERgIGI1iKpiYQ	NaN	Provincia di Urubamba	Regione di Cusco	Perù
5722	6273a0a9dd67f76198e5ac192	606	Come arrivare a Machu Picchu: modalità e tempi	Sembrerà un argomento banale: vuoi che sia dif...	['machupicchu', 'Perù', 'peru', 'visitperu', '...	['Viaggi']	7	ChIJVVVVlV-abZERJxqgpA43EDo	NaN	Provincia di Urubamba	Regione di Cusco	08680
5723	625f18770f9a6017435172 52	579	Le terrazze di Moray, nella Valle Sacra degli ...	Le terrazze di Moray sono un sito archeologico...	['coltivazione', 'inca', 'agricoltura', 'natura']	['Cultura']	6	ChIJyZGitKXDbZERg4qm1pVfvBQ	Maras	Urubamba	Cuzco	Peru

Figure 5.1. Sample of Easy Tour Reviews Data

The data structure represented in the figure 5.1 contains the following columns:

- **id:**
  Identification of the review.
- **userId:**
  Identification of the author of the review.
- **title:**
  Title of the review.
- **description:**
  The corpus of the review.
- **tags:**
  Keywords of the review.

- **categories:**
  Category of the attraction
- **viewCounter:**
  Counter of the viewers of the review.
- **placeId:**
  Identification of where the attraction took place.
- **city:**
  City of where the attraction took place.
- **province:**
  Province of where the attraction took place.
- **region:**
  Region of where the attraction took place.
- **state:**
  State of where the attraction took place.

5.1.2 Easy Tour Likes

The likes data is the representation of the instantaneous feedback of users to a given review. In this data, we have significant information about how the users act inside the network. How they perceive what was written about a certain attraction and their judgement of the information which it contains.

	documentId	authorId	type	bored	inspire	lol	love	useful	willdoit	wow	yum
0	622c3b8ac992ca28f33c7f21	606	['Useful']	0	0	0	0	1	0	0	0
1	620d214425398565dadb6984	180	['Useful']	0	0	0	0	1	0	0	0
2	620d214425398565dadb6984	531	['Useful']	0	0	0	0	1	0	0	0
3	621e537d82ed1d1eef3a394f	531	['Useful']	0	0	0	0	1	0	0	0
4	621e537d82ed1d1eef3a394f	550	['Useful']	0	0	0	0	1	0	0	0
...	...	...	...	...	...	...	...	...	...	...	...
2537	62767583e44220507a306b0e	304	['Inspire']	0	1	0	0	0	0	0	0
2538	62767583e44220507a306b0e	688	['Useful']	0	0	0	0	1	0	0	0
2539	6273a0a9dd67f6198e5ac192	399	['Useful']	0	0	0	0	1	0	0	0
2540	61e0146111b3bf4d3185ad5d	480	['Useful']	0	0	0	0	1	0	0	0
2541	622f3149ef19af4233396c98	550	['Useful']	0	0	0	0	1	0	0	0

Figure 5.2. Sample of Easy Tour Likes Data

The likes data have the following columns, described below:

- **documentId:**
  Identification of the review, which was liked.
- **authorId:**
  Identification of the like's author.

- **type:**
  The type of reaction given by the user to a review.
- **bored, inspire, lol, love, useful, willdoit, wow, yum:**
  By encoding the "type" of the like, we have a flag in the following categories to indicate the type of the like.

5.2 Exploration of the Easy Tour Dataset

The review data contains a total number of 5724 reviews, produced by 156 users. Those quantities give us the ratio 36.7 reviews per user, which is a good ratio.

In fact, if the distribution of reviews was uniform, it would mean that the engagement of all the users in the platform was very high, but indeed, when the real numbers are analyzed, we can see that:

User ID	Number of Reviews
158	114
435	105
304	73
680	73

From 5724 reviews, 4 users were the authors of 365 of those, which means that 2,56% have written 6,38% of the reviews.

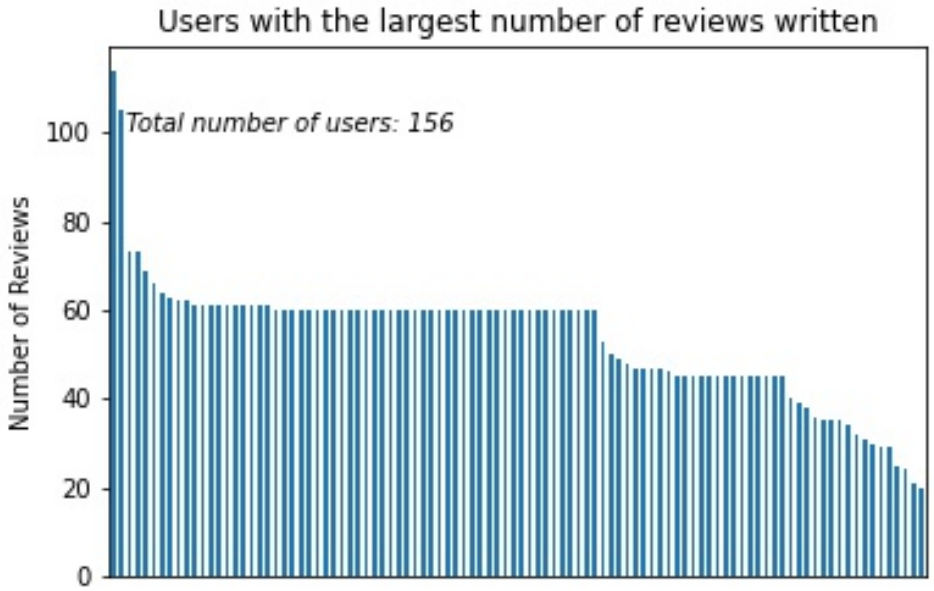


Figure 5.3. Users with the largest reviews done in Easy Tour Dataset

The figure 5.3 shows the distribution of the wrote reviews per user and gives a clear vision of how it is distributed.

Analyzing the attractions of the dataset, we have, 3167 different experiences, with the most reviewed ones represented above:

Place ID	Number of Reviews	City
ChIJu46S-ZZhLxMROG5lkwZ3D7k	80	Roma
ChIJ6$_p$622$YIOxMRfriMZcxDOtI$	48	Napoli
EihWaWEgRG9tZW5pY28gRm9ud...	43	Napoli
ChIJrdbSgKZWKhMRAyrH7xd51ZM	24	Firenze

By analyzing those numbers, we can see that 4 attractions, over 3167, were responsible for 195 reviews. Those numbers, combined, give us the information that 0,12% of the attractions concentrate 3,4% of the reviews, which can be seen as the most reviewed places and principal touristic destinations.

Another interesting point is that, some attractions maybe are in the same travel destination. So, as we can see above, 2 of the 4 most reviewed attractions are in Napoli, which give us an idea of the importance of analyzing not only the ID of the place, but also directly the city of the experience in order to give a meaningful recommendation as a travel destiny.

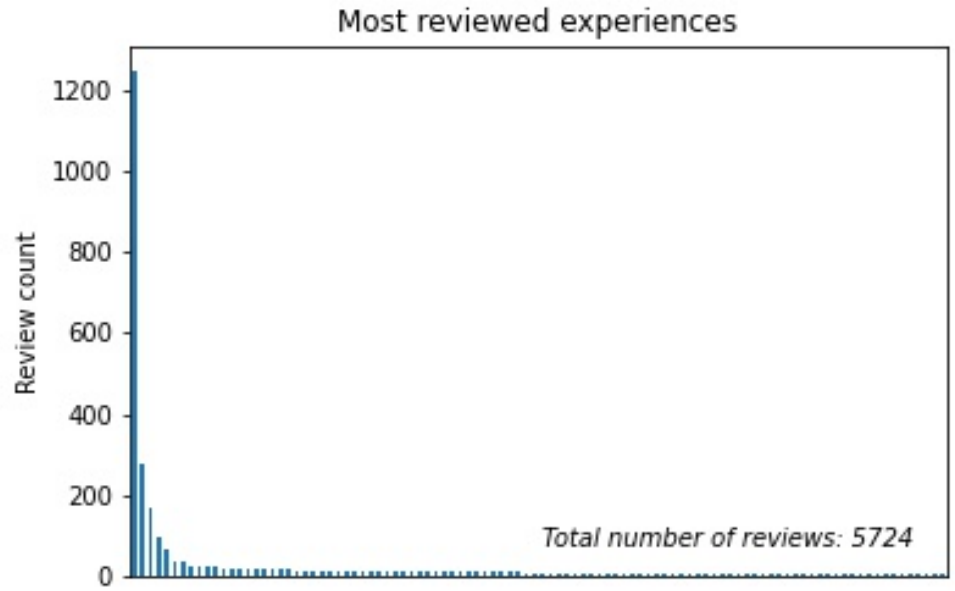


Figure 5.4. Experiences with the largest reviews done in Easy Tour Dataset

In the figure 5.4 it is clearly shown how the reviews are concentrated in few destinations.

5.3 Trip Advisor Dataset

For the aim of this project, data from the biggest online platform of tourism was gathered, where the cities of Barcelona, Berlin, London, Madrid, New York, Paris, and Rome were used.

The data consists of reviews of activities done by users, where each one gives a rate for the activity, followed by a written description of the review. As in the figure 5.5 we can see a sample of the data collected from Trip Advisor from activities of the Spanish city, Madrid:

	userName	userUrl	reviewDate	usertext	review	host_id	review_id	usertext_processed
0	Rori F	https://www.tripadvisor.com/Profile/Greyhoundm...	September 9, 2022	My sister and I booked our full day trip to vi...	5.0	TZ917HUF8HP1	CF5D3AGFD1PEB5GIR7	['sister', 'booked', 'full', 'day', 'trip', 'v...
1	Maps516996	https://www.tripadvisor.com/Profile/Maps516996	July 3, 2022	My daughters and I thoroughly enjoyed our full...	5.0	TZ917HUF8HP1	X9ENIHJ8DJZ37R8PLM	['thoroughly', 'full', 'day', 'tour', 'avila',...
2	Diana M	https://www.tripadvisor.com/Profile/dianam770	February 28, 2022	Very nice tour of Avila and Segovia. The guide...	4.0	TZ917HUF8HP1	P7Z22GGTQUX1LXMMX0	['tour', 'avila', 'segovia', 'guida', 'knowled...
3	Dantecass	https://www.tripadvisor.com/Profile/Dantecass	October 30, 2022	I'll start with the "cons" because I felt ther...	3.0	TZ917HUF8HP1	NIWFY6KEORG7U3OW9U	['start', 'cons', 'felt', 'mora', 'cons', 'pro...
4	neillay51	https://www.tripadvisor.com/Profile/neillay51	October 30, 2022	The trip was not well organised. The microphon...	2.0	TZ917HUF8HP1	MC8LA0P1WZH36OK01R	['trip', 'well', 'microphone', 'receiving', 'd...
...	...	...	...	...	...	...	...	...
15160	עידו ש	https://www.tripadvisor.com/Profile/B3216CK_	July 3, 2019	Perfect trip, Benjamin was a great guide, funn...	5.0	RBZMVY2K8HDM	QFJGLO2DII3WMZREGK	['trip', 'benjamin', 'great', 'funny', 'kind',...
15161	LB1	https://www.tripadvisor.com/Profile/leesa_barker	September 24, 2018	A good mix of history, scandal and humour. My ...	5.0	RBZMVY2K8HDM	U22GUVK57CY29TOAV7	['good', 'mix', 'history', 'scandal', 'humour'...
15162	Shelley O	https://www.tripadvisor.com/Profile/shelleyo249	May 29, 2018	This was a great activity - we were a large gr...	5.0	H2UUM0HKMH01	6PFZG2I9PGIDXBLTHT	['great', 'activity', 'largire', 'group', 'spl...
15163	RashmiOberoi	https://www.tripadvisor.com/Profile/RashmiOberoi	October 26, 2017	We had a great time. The tour is a great way t...	5.0	H2UUM0HKMH01	2HDAN1F8B4C6KUNPN6	['great', 'tour', 'great', 'way', 'see', 'beau...
15164	Seth M	https://www.tripadvisor.com/Profile/sethm955	January 23, 2020	I was the only one on the tour, and I only got...	3.0	H2UUM0HKMH01	1N2TOBRJJ4Y1S4YIVR	['one', 'tour', 'got', 'tapas', 'end', 'expect...

15165 rows × 8 columns

Figure 5.5. Sample of Trip Advisor Madrid Dataset

The dataset used is a result of a web-scrapping process, where the data collected is stored in the follow columns:

- **userName:**
  The identification of the author of the given review.
- **userUrl:**
  The link to the author profile.
- **reviewDate:**
  The date of the publication of the review.
- **usertext:**
  The written review done by the user.
- **review:**
  The review grade of the experience according to user.
- **host_id:**
  The identifier index of the experience.
- **review_id:**
  The identifier index of the review done by user.
- **usertext_processed:**
  The written review done by the user, processed and tokenized for the input model.

The data set has the same structure mentioned above in all the cities used in our analysis.

5.4 Exploration of the Trip Advisor Dataset

According to each city of the dataset, the distribution of reviews per user is different, so as the distribution of reviews per experience. This main aspect of the

data is given by differences in the touristic points visited, because as noticed, each city has a different number of reviews and users who were in a given city.

So, because of those discrepancies, the dataset was chucked in different cities and explored. The investigated points were the total number of reviews, the total number of active users who wrote a review about an experience, the best experiences based on the reviews score and the most engaged users in the platform in respect to a specific city.

5.4.1 Berlin

Here in the figure 5.6 we can see a sample of the dataset of the city, which is further analyzed below:

	userName	userUrl	reviewDate	usertext	review	host_id	review_id	usertext_processed
0	Frances G	https://www.tripadvisor.com/Profile/francesg622	November 8, 2022	We were in Berlin for 3 days and were keen to ...	5.0	PYYAMK3VL8S2	RMTPNLU8CX3SALS5HV	['berlin', 'keen', 'discover', 'much', 'could'...
1	MariaLR	https://www.tripadvisor.com/Profile/marialr2001	September 30, 2022	My friends and I enjoyed the tour very much! W...	5.0	PYYAMK3VL8S2	7UFFBEZDZUU7ODJV3W	['friends', 'tour', 'much', 'appreciated', 'un...
2	jbushman2017	https://www.tripadvisor.com/Profile/jbushman2017	June 27, 2022	I love walking tours and have taken many. This...	5.0	PYYAMK3VL8S2	6MYDVU3FSQZCZUQ6T2	['love', 'walking', 'tours', 'taken', 'tour', ...
3	india	https://www.tripadvisor.com/Profile/Trek062916...	November 3, 2021	We were lucky enough to have Campbell as our t...	5.0	PYYAMK3VL8S2	88GE6JPM8DNZ2ESGW8	['lucky', 'enough', 'campbell', 'tour', 'guide...
4	Maria S	https://www.tripadvisor.com/Profile/LouiseAussie	June 9, 2022	This tour led by our very knowledgeable and en...	5.0	PYYAMK3VL8S2	E6S1IO8OL69RYOKZCQ	['tour', 'led', 'knowledgeable', 'entertaining...
5	Joanna D	https://www.tripadvisor.com/Profile/joannadT23...	October 22, 2022	Our tour guide, Scott was amazing! He was very...	5.0	PYYAMK3VL8S2	KQB1ZIB4HFZDDTKHHX	['tour', 'scott', 'amazing', 'knowledgeable', ...
6	RedcliffCathy	https://www.tripadvisor.com/Profile/RedcliffCathy	June 16, 2022	We met at the assigned spot, that seems like a...	4.0	PYYAMK3VL8S2	F45TVBT16R5A47F4NH	['met', 'spot', 'seems', 'like', 'tours', 'sta...
7	Joseph D	https://www.tripadvisor.com/Profile/R5420FQjos...	July 8, 2022	Thoroughly enjoyable tour of Berlin, with a wi...	5.0	PYYAMK3VL8S2	CY2JKALL225FKAHSS2	['enjoyable', 'tour', 'berlin', 'wide', 'rangi...
8	ELT	https://www.tripadvisor.com/Profile/ELT1234	October 8, 2021	I did a half day tour with friends on my first...	5.0	PYYAMK3VL8S2	VBOCP9XNNJJ3A45AHO	['half', 'day', 'tour', 'friends', 'first', 'b...
9	shauychan8	https://www.tripadvisor.com/Profile/shauychan8	May 9, 2022	Glen from England was my guide for this group ...	5.0	PYYAMK3VL8S2	A93S2NB6WMZBBVZLS9	['glen', 'england', 'guida', 'group', 'walking...

Figure 5.6. Sample of Trip Advisor Berlin Dataset

The dataset scrapped consists of reviews from the year of 2018 until 2022, which shows that Berlin is a very active touristic destiny. Counting with a number of 226 experiences analyzed in the data set, we can see that 12696 users were responsible for making 14985 reviews for those 226 experiences, which gives a number of 1,18 reviews per user and 66,3 reviews per experience.

The figure 5.7a represents the best experiences in the platform Trip Advisor for the city of Berlin. As we can see below, the experiences are identified by an ID number and with the correspondent score given by all the users who put a review on that experience.

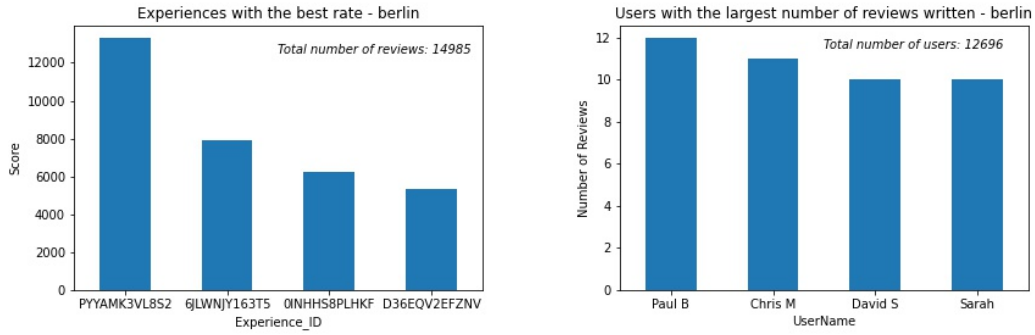
The most scored experiences in Berlin are identified by the IDs "PYYAMK3VL8S2", "6JLWNJY163T5", "0INHHS8PLHKF" and "D36EQV2EFZNV". And the number of reviews and scores can be identified in the table:

Experience ID	Number of Reviews	Total Score
PYYAMK3VL8S2	2695	13322.0
6JLWNJY163T5	1605	7903.0
0INHHS8PLHKF	1257	6239.0
D36EQV2EFZNV	1089	5339.0

Also, another important point to be analyzed is the number of reviews per user in the figure 5.7b, where we can see the engagement of the users in the platform.

UserName	Number of Reviews
Paul B	12
Chris M	11
David S	10
Sarah	10

We can see that, for the city of Berlin, the top 4 users with the bigger engagement are responsible for 0,29% of the reviews, where those are: Paul B, Chris M, David S and Sarah.



(a) Experiences with best scores in Trip Advisor Berlin Dataset

(b) Users with the largest reviews done in Trip Advisor Berlin Dataset

Figure 5.7. Exploration of Berlin Dataset

5.4.2 Paris

The dataset of Paris consists of reviews from the year of 2015 until 2022. It is composed by a number of 858 experiences, and we can see that 28680 users were responsible for making 38707 reviews, which gives a number of 1,35 reviews per user and 45,2 reviews per experience.

	userName	userUrl	reviewDate	usertext	review	host_id	review_id	usertext_processed
0	terryp0060	https://www.tripadvisor.com/Profile/terryp0060	March 16, 2022	My wife and I enjoyed a wonderful afternoon wi...	5.0	V85F7KJ0V1GY	MJQWD9G1RVGW13MFHY	['wife', 'enjoyed', 'wonderful', 'afternoon', ...
1	Amy	https://www.tripadvisor.com/Profile/Explore665...	June 22, 2021	Best experience of our vacation in Paris! Thie...	5.0	V85F7KJ0V1GY	KXQ3HON0UFIWHTJ7GR	['experience', 'vacation', 'paris', 'thierry',...
2	kwilk	https://www.tripadvisor.com/Profile/kwilk	January 21, 2020	We had an excellent time today with Thierry at...	5.0	V85F7KJ0V1GY	2ZY160OYQ74ES1OGLI	['time', 'today', 'thierry', 'wine', 'pairing'...
3	Levi N	https://www.tripadvisor.com/Profile/LeviN_11	April 5, 2022	We signed up for this wine and cheese pairing ...	4.0	V85F7KJ0V1GY	SMIE8CD8LGC0CLLA1N	['signed', 'wine', 'pairing', 'based', 'recomm...
4	kristiesue2	https://www.tripadvisor.com/Profile/kristiesue2	October 3, 2022	This was a great way to start our Paris vacati...	5.0	V85F7KJ0V1GY	01KIMQ5X6R0UXZM8B8	['great', 'way', 'start', 'paris', 'vacation',...
5	□ □□□□	https://www.tripadvisor.com/Profile/frimanal	February 26, 2020	A must stop while in Paris. Teirry is very pro...	5.0	V85F7KJ0V1GY	VJ2M1O9RPCF10MYOAN	['stop', 'paris', 'teirry', 'proffesional', 'i...
6	Kevin G	https://www.tripadvisor.com/Profile/J7158MBkeving	March 8, 2020	Bought as suprise for my wife's birthday in sh...	5.0	V85F7KJ0V1GY	1P4ISTMYCB04SJ67UK	['bought', 'wife', 'birthday', 'short', 'excel...
7	ttrice93	https://www.tripadvisor.com/Profile/ttrice93	January 31, 2019	My best friend and I were so excited about thi...	5.0	V85F7KJ0V1GY	SFWFX9QODDSD8BHI4Y	['friend', 'excited', 'wine', 'tasting', 'expe...
8	Kristen B	https://www.tripadvisor.com/Profile/kristenbN4...	April 24, 2018	Very cool little place and experience. Great h...	5.0	V85F7KJ0V1GY	2ZZM6QOZQB1MAFUBVG	['place', 'experience', 'great', 'history', 'c...
9	Manoj Prithiani	https://www.tripadvisor.com/Profile/retronomical	November 15, 2019	I wasn't sure about opting for this one howeve...	5.0	V85F7KJ0V1GY	RK7WI394OLSDFC5JQY	['sure', 'opting', 'one', 'turned', 'thierry',...

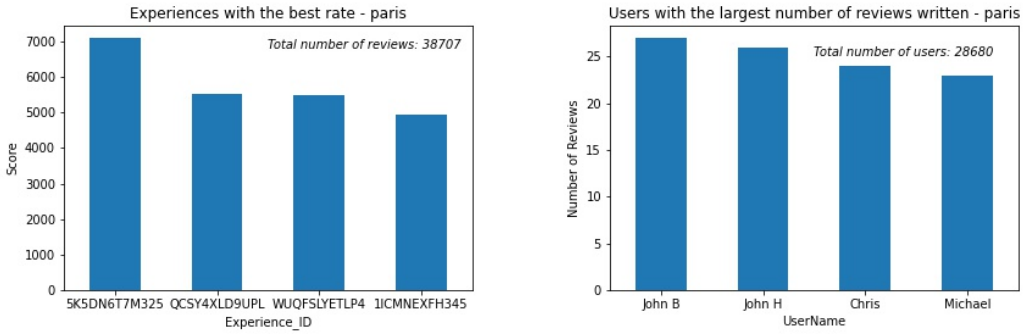
Figure 5.8. Sample of Trip Advisor Paris Dataset

The figure 5.9a represents the best experiences in the platform Trip Advisor for the city of Paris. The most scored experiences in Paris are identified by the IDs “5K5DN6T7M325”, “WUQFSLYETLP4”, “QCSY4XLD9UPL” and “FWI0TXOAB266”. And the number of reviews and scores can be seen in the table below:

Experience ID	Number of Reviews	Total Score
5K5DN6T7M325	1495	7102.0
WUQFSLYETLP4	1400	5528.0
QCSY4XLD9UPL	1250	5504.0
FWI0TXOAB266	1104	4949.0

Another point to be analyzed is the number of reviews per user in the figure 5.9b, where we can see the engagement of the users in the platform.

UserName	Number of Reviews
John B	27
John H	26
Chris	24
Michael	23



(a) Experiences with best scores in Trip Advisor Paris Dataset

(b) Users with the largest reviews done in Trip Advisor Paris Dataset

Figure 5.9. Exploration of Paris Dataset

We can see the top 4 users with the bigger engagement are responsible for 0,26% of the reviews, where those are: John B, John H, Chris and Michael.

5.4.3 Rome

The dataset of Rome has reviews from the year of 2015 until 2022. It contains a number of 1612 experiences, reviewed by 46178 users with a total of 68336 reviews, which gives a number of 1,48 reviews per user and 42,4 reviews per experience.

	userName	userUrl	reviewDate	usertext	review	host_id	review_id	usertext_processed
0	Silvia H	https://www.tripadvisor.com/Profile/621silviah	October 21, 2022	Great experience!! Eni, our guide was fantasti...	5.0	3T7V6VVPP5CU	YOW7DQ6FD0I5CYCX7C	['guida', 'fantastic', 'absolutely', 'tour', '...
1	Argesa H	https://www.tripadvisor.com/Profile/argesah	September 23, 2022	It was a really nice tour. Valentina our tour ...	5.0	3T7V6VVPP5CU	VLPTTCHKXPL4VISPDA	['really', 'tour', 'valentina', 'tour', 'guida...
2	Camilla OBrien	https://www.tripadvisor.com/Profile/Travelingw...	July 1, 2022	Our tour guide Valentina was extremely enthusi...	5.0	3T7V6VVPP5CU	VARODXBENW3Z3FIP9Q	['tour', 'guida', 'valentina', 'extremely', 'e...
3	Michele S	https://www.tripadvisor.com/Profile/T7701LFmic...	June 23, 2022	Eni, our guide was fantastic!!! So knowledgeab...	5.0	3T7V6VVPP5CU	9HHV9D8RNM4W3XPQVM	['guida', 'fantastic', 'knowledgeable', 'answe...
4	Melisa K	https://www.tripadvisor.com/Profile/melisak905	May 25, 2022	Amazing experience! I absolutely recommend thi...	5.0	3T7V6VVPP5CU	FADSZRN2MATB9G6WGQ	['amazing', 'absolutely', 'tourwe', 'visited',...
5	Paul S	https://www.tripadvisor.com/Profile/paulsl8407VT	August 4, 2022	We have finally done it! The Secrets of Rome t...	5.0	C845Z4DMX8DB	KR18PVSTUITHQFOAUG	['finally', 'done', 'secrets', 'rome', 'tour',...
6	Danielle L	https://www.tripadvisor.com/Profile/daniellelK...	February 22, 2022	We've been to Rome a few times so we wanted to...	5.0	C845Z4DMX8DB	56NHBIATGTUGYYP07F	['wanted', 'explore', 'something', 'new', 'dem...
7	amybdent	https://www.tripadvisor.com/Profile/amybdent	June 27, 2018	My husband Scott, and I visited Rome in May of...	5.0	C845Z4DMX8DB	OKYBXN0NNA4A1L3F9G	['husband', 'scott', 'may', 'booked', 'tour', ...
8	Kyle A	https://www.tripadvisor.com/Profile/Exploratio...	February 20, 2020	My wife, 7-year-old daughter, and myself LOVED...	5.0	88UXOWMYTWSF	1PN13XFG1934JFL1TR	['wife', '7year', 'old', 'daughter', 'tours', ...
9	Carol A	https://www.tripadvisor.com/Profile/cja010511	February 17, 2020	We moved to Italy a few months ago, and it was...	5.0	88UXOWMYTWSF	KQ86OPNGG612CFN8O3	['moved', 'italy', 'months', 'time', 'visit', ...

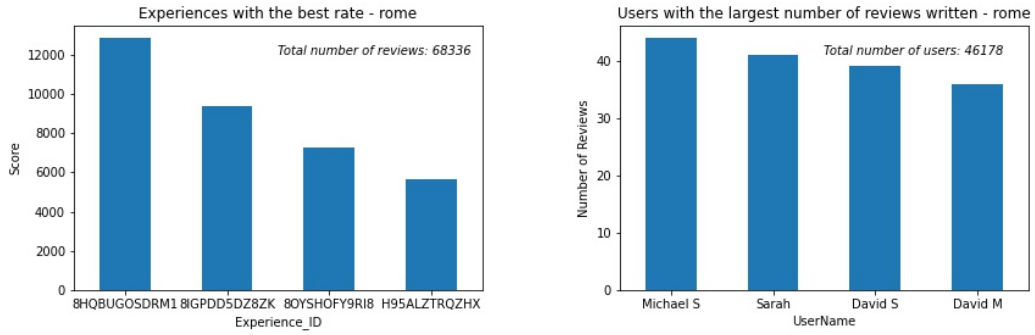
Figure 5.10. Sample of Trip Advisor Rome Dataset

In the figure 5.11a is represented the best experiences in the platform for the city of Rome.

The most scored experiences are identified by the IDs "8HQBUGOSDRM1", "8IGPDD5DZ8ZK", "8OYSHOFY9RI8" and "H95ALZTRQZHX". And the number of reviews and scores can be seen in the table below:

Experience ID	Number of Reviews	Total Score
8HQBUGOSDRM1	2799	12844.0
8IGPDD5DZ8ZK	2060	9336.0
8OYSHOFY9RI8	1518	7227.0
H95ALZTRQZHX	1130	5617.0

Another point, is the number of reviews per user in the figure 5.11b, where we can see the engagement of the users in the platform.



(a) Experiences with best scores in Trip Advisor Rome Dataset

(b) Users with the largest reviews done in Trip Advisor Rome Dataset

Figure 5.11. Exploration of Rome Dataset

UserName	Number of Reviews
Michael S	44
Sarah	41
David S	39
David M	36

We can see the top 4 users with the bigger engagement are responsible for 0,23% of the reviews, where those are: Michael S, Sarah, David S and David M.

5.4.4 New York

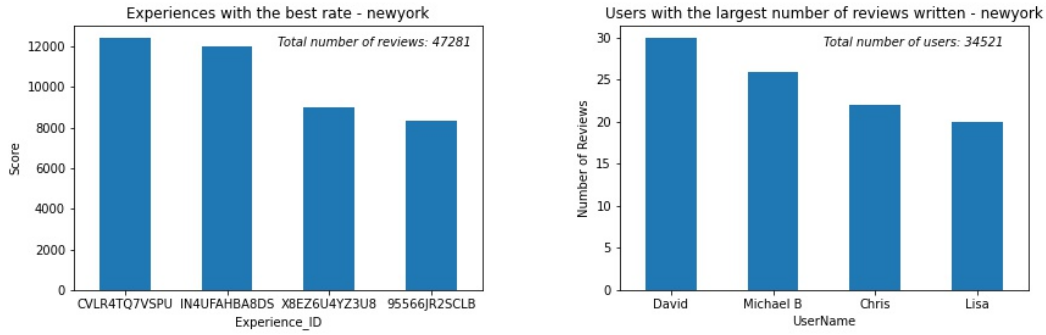
The New York dataset has reviews from the year of 2015 until 2022, and it contains a number of 660 experiences, reviewed by 34521 users with a total of 47281 reviews, which gives a number of 1,37 reviews per user and 71,6 reviews per experience.

	userName	userUrl	reviewDate	usertext	review	host_id	review_id	usertext_processed
0	Clara D	https://www.tripadvisor.com/Profile/clarad516	May 15, 2022	This was our first time in New York and so we ...	5.0	IN4UFAHBA8DS	QKFG6FKT6XIPH0VL8S	['first', 'time', 'new', 'york', 'wanted', 'to...
1	Paul S	https://www.tripadvisor.com/Profile/paulsV9	September 12, 2022	A good tour that was made great by our guide S...	5.0	IN4UFAHBA8DS	9EXQVVRGCS7MLOPDVU	['good', 'tour', 'made', 'great', 'guida', 'st...
2	Aly M	https://www.tripadvisor.com/Profile/alym645	April 5, 2022	I've been to NYC quite a few times, but it was...	5.0	IN4UFAHBA8DS	MK2USVXJ96XNLVEA6F	['nyc', 'times', 'husbands', 'first', 'since',...
3	Justine N	https://www.tripadvisor.com/Profile/justinen741	October 29, 2022	We had Dana as our tour guide and she was sooo...	5.0	IN4UFAHBA8DS	HAISTQG86B5UVU3OCO	['dana', 'tour', 'guida', 'much', 'made', 'sur...
4	Donise E	https://www.tripadvisor.com/Profile/d0nisee	April 13, 2022	We took the upgraded tour with the smaller gla...	5.0	IN4UFAHBA8DS	6F7R1KGY1HU8MNA8B5	['took', 'tour', 'smaller', 'glass', 'bus', 'c...
5	UncleDawg01	https://www.tripadvisor.com/Profile/UncleDawg01	March 10, 2022	This is the first thing you are going to say w...	5.0	IN4UFAHBA8DS	MHQP76OVNI6A207NSC	['first', 'thing', 'going', 'say', 'sit', 'plu...
6	Cormiermg1	https://www.tripadvisor.com/Profile/Cormiermg1	September 30, 2022	What a wonderful day!! Saw and learned so much...	5.0	IN4UFAHBA8DS	1X7U1EEEJKV79AWAPO	['wonderful', 'day', 'learned', 'much', 'guida...
7	goegypt	https://www.tripadvisor.com/Profile/goegypt	November 2, 2022	My husband and I got to see so much of NYC on ...	5.0	IN4UFAHBA8DS	QBEFZT6MKBWV2NOZTV	['husband', 'got', 'see', 'much', 'nyc', 'full...
8	Lon S	https://www.tripadvisor.com/Profile/lons646	November 5, 2022	I would highly recommend this bus trip early i...	5.0	IN4UFAHBA8DS	1GKC0M3UDKKLWIMN7J	['would', 'highly', 'bus', 'experience', 'grea...
9	Kim A	https://www.tripadvisor.com/Profile/kimaU7972YS	July 26, 2022	This was a great tour to see an overview of Ne...	5.0	IN4UFAHBA8DS	2H0L9XX05KE2K5X4D1	['great', 'tour', 'overview', 'new', 'york', '...

Figure 5.12. Sample of Trip Advisor New York Dataset

In 5.13a is represented the best experiences for the city of New York, where the most scored experiences are “CVLR4TQ7VSPU”, “IN4UFAHBA8DS”, “X8EZ6U4YZ3U8” and “95566JR2SCLB”:

Experience ID	Number of Reviews	Total Score
CVLR4TQ7VSPU	2553	12436.0
IN4UFAHBA8DS	2494	12006.0
X8EZ6U4YZ3U8	1826	9007.0
95566JR2SCLB	1742	8346.0



(a) Experiences with best scores in Trip Advisor New York Dataset

(b) Users with the largest reviews done in Trip Advisor New York Dataset

Figure 5.13. Exploration of New York Dataset

The number of reviews per user in the figure 5.13b gives an idea of the engagement of the users, and we can see the top 4, which are responsible for 0,21% of total reviews.

UserName	Number of Reviews
David	30
Michael B	26
Chris	22
Lisa	20

5.4.5 Barcelona

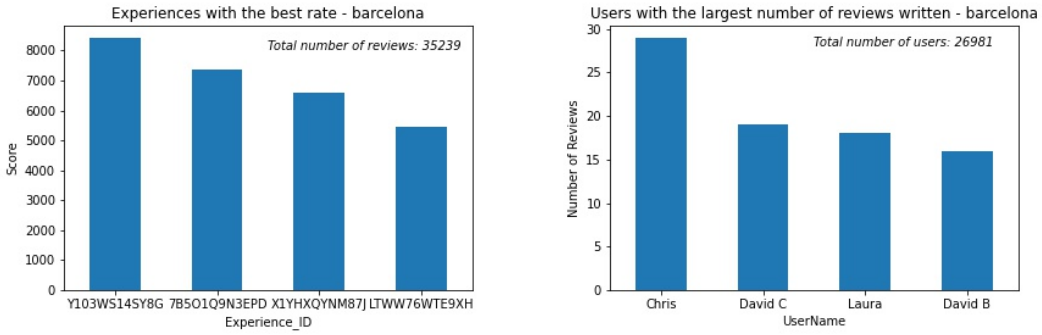
The Barcelona dataset has reviews from the year of 2009 until 2022, and it contains a number of 696 experiences, reviewed by 26981 users with a total of 35239 reviews, which gives a number of 1,31 reviews per user and 50,6 reviews per experience.

	userName	userUrl	reviewDate	usertext	review	host_id	review_id	usertext_processed
0	sallyrowena	https://www.tripadvisor.com/Profile/voyageroxf...	October 29, 2022	This was the highlight of our stay in Barcelon...	5.0	Z49BXJAEF3LT	KZ0J3V9YMD22YUHB54	['highlight', 'barcelona', 'good', 'considerin...
1	Miggsoo3	https://www.tripadvisor.com/Profile/Miggsoo3	July 8, 2022	Very informative tour Alberto was extremely kn...	5.0	Z49BXJAEF3LT	T4KX1R1QAJTATI37EL	['tour', 'extremely', 'knowledgeable', 'accomm...
2	John F	https://www.tripadvisor.com/Profile/U3804TLjohnf	April 8, 2022	Suzie is fantastic! Without a knowledgeable gu...	5.0	Z49BXJAEF3LT	AAHT1TTNNZ9XKAEJG8	['suzie', 'fantastic', 'without', 'knowledgeab...
3	Alina	https://www.tripadvisor.com/Profile/oprinaalina	April 10, 2022	Because of transportation, difficult in an unk...	2.0	Z49BXJAEF3LT	IBQHHTJECFV19I2B3U	['transportation', 'difficult', 'city', 'ten',...
4	Carolyn J	https://www.tripadvisor.com/Profile/CarolynJ472	October 30, 2022	I have done many tours in many lands. Most of ...	5.0	Z49BXJAEF3LT	TR25C3M5BTUK2KJZON	['done', 'tours', 'many', 'pleasant', 'experie...
5	A N	https://www.tripadvisor.com/Profile/AndreaN62	May 21, 2022	Fabulous tour with our lovely friendly guide. ...	4.0	Z49BXJAEF3LT	2IND3TL5WU7R7Z8KQ5	['fabulous', 'tour', 'lovely', 'friendly', 'gu...
6	Ada N	https://www.tripadvisor.com/Profile/AdaN280	April 23, 2022	Our tour guide, Violetta was outstanding. She ...	5.0	Z49BXJAEF3LT	26D02I5RDB3DQADNSD	['tour', 'outstanding', 'much', 'history', 'ba...
7	KasiaMarc	https://www.tripadvisor.com/Profile/KasiaMarc	March 2, 2022	Olga Escribano was our guide on this tour and ...	5.0	Z49BXJAEF3LT	BD2DI3Z2EY7KWOSXCU	['guida', 'tour', 'wow', 'amazing', 'true', 'j...
8	Margot R	https://www.tripadvisor.com/Profile/mmroussel33	October 30, 2022	She was very knowledgeable and detail oriented...	4.0	Z49BXJAEF3LT	XMDO42QO4IONGG68HM	['knowledgeable', 'oriented', 'great', 'older'...
9	A Kim	https://www.tripadvisor.com/Profile/HaasTravel	November 16, 2021	I just took a tour of Park Guell and La Sagrad...	5.0	Z49BXJAEF3LT	O2OJXAXYYHAR6NN7UR	['tour', 'park', 'guell', 'sagrada', 'familia'...

Figure 5.14. Sample of Trip Advisor Barcelona Dataset

In 5.15a is represented the best experiences for the city of Barcelona, and in the figure 5.15b we have the engagement of the users.

Experience ID	Number of Reviews	Total Score
Y103WS14SY8G	1774	8427.0
7B5O1Q9N3EPD	1480	7366.0
X1YHXQYNM87J	1355	6594.0
LTWW76WTE9XH	1108	5434.0



(a) Experiences with best scores in Trip Advisor Barcelona Dataset

(b) Users with the largest reviews done in Trip Advisor Barcelona Dataset

Figure 5.15. Exploration of Barcelona Dataset

Below, we can see that the top 4 users are responsible for 0,23% of total reviews.

UserName	Number of Reviews
Chris	29
David C	19
Laura	18
David B	16

5.4.6 London

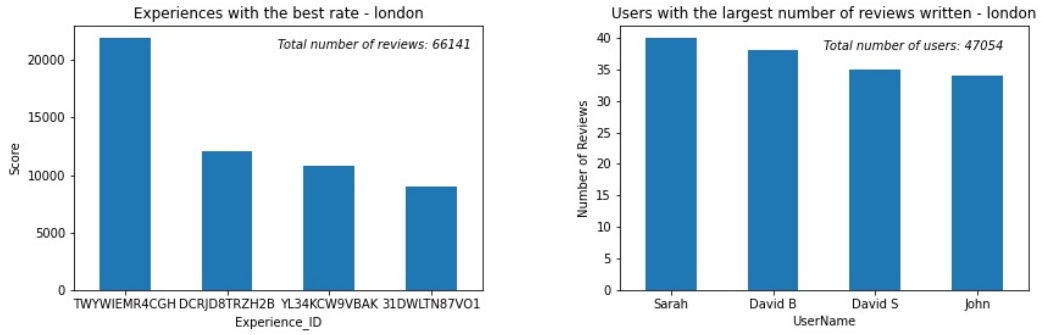
	userName	userUrl	reviewDate	usertext	review	host_id	review_id	usertext_processed
0	Elyse W	https://www.tripadvisor.com/Profile/elysew2017	January 30, 2022	Francis took us on a private tour and it was a...	5.0	NW9HICV3N330	KRDI9NYA5LU5TDCQMI	['francis', 'took', 'tour', 'highlight', 'trip...
1	Shirley Paulson	https://www.tripadvisor.com/Profile/shirlpaulson	April 17, 2022	Frances was the perfect guide for the three of...	5.0	NW9HICV3N330	GU48AQMYGBW91TB173	['frances', 'guida', 'three', 'asked', 'engage...
2	Terry W	https://www.tripadvisor.com/Profile/1976TerryW	October 29, 2021	Had a great afternoon touring around Westminst...	5.0	NW9HICV3N330	FOP5KWYI6LULVDJ9YN	['great', 'afternoon', 'touring', 'around', 'w...
3	Martin	https://www.tripadvisor.com/Profile/Mar1980Mar	November 9, 2021	This was one of the best tours I have ever bee...	5.0	NW9HICV3N330	MU670VMMJX723746II	['one', 'tours', 'ever', 'friend', 'saw', 'hig...
4	June B	https://www.tripadvisor.com/Profile/U1434YZjuneb	February 13, 2022	Urban Saunters was easy to work with and we ha...	5.0	NW9HICV3N330	FJS849GP1HZDSXH5HS	['urban', 'saunters', 'easy', 'work', 'blast',...
5	HAL	https://www.tripadvisor.com/Profile/68786	July 26, 2022	We ended up being a very small tour. This cove...	4.0	NW9HICV3N330	XJ0E4P3NFM37T6LRQD	['ended', 'tour', 'covered', 'westminister', '...
6	Susan G	https://www.tripadvisor.com/Profile/888SusanG888	January 15, 2022	Robert was lovely and we could have listened t...	5.0	NW9HICV3N330	VFVCBH1FNDR4OKZT1B	['robert', 'lovely', 'could', 'listened', 'tou...
7	Graham W	https://www.tripadvisor.com/Profile/W8639PIgra...	December 17, 2021	Really enjoyed my private WW2 walking tour wit...	5.0	NW9HICV3N330	RPGOIHTTXHEZKDFZLA	['really', 'ww2', 'walking', 'tour', 'richard'...
8	Marek M	https://www.tripadvisor.com/Profile/L7726HJmarekm	June 12, 2021	David had a wealth of knowledge about Westmins...	5.0	NW9HICV3N330	973TW0YIRR7XQB9LGS	['david', 'wealth', 'knowledge', 'westminster'...
9	Sightsee596913	https://www.tripadvisor.com/Profile/Sightsee59...	June 9, 2021	Francis was absolutely fantastic. Very pleasan...	5.0	NW9HICV3N330	WWM8EI7RQID0KQ2LGE	['francis', 'absolutely', 'pleasant', 'manner'...

Figure 5.16. Sample of Trip Advisor London Dataset

London has reviews from the year of 2013 until 2022. It contains 920 experiences, reviewed by 47054 users with a total of 66141 reviews, which gives a number of 1,40 reviews per user and 71,9 reviews per experience.

Experience ID	Number of Reviews	Total Score
TWYWIEMR4CGH	5667	21923.0
DCRJD8TRZH2B	2540	12036.0
YL34KCW9VBAK	2165	10758.0
31DWLTN87VO1	1806	8992.0

In 5.17a is represented the best experiences and in the figure 5.17b we have the engagement of the users, where the top 4 users are responsible for 0,22% of total reviews.



(a) Experiences with best scores in Trip Advisor London Dataset

(b) Users with the largest reviews done in Trip Advisor London Dataset

Figure 5.17. Exploration of London Dataset

UserName	Number of Reviews
Sarah	40
David B	38
David S	35
John	34

5.4.7 Madrid

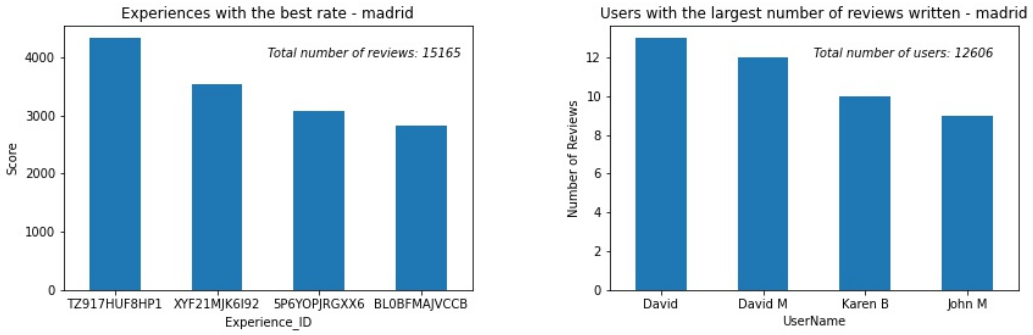
	userName	userUrl	reviewDate	usertext	review	host_id	review_id	usertext_processed
0	Rori F	https://www.tripadvisor.com/Profile/Greyhoundm...	September 9, 2022	My sister and I booked our full day trip to vi...	5.0	TZ917HUF8HP1	CF5D3AGFD1PEB5GIR7	['sister', 'booked', 'full', 'day', 'trip', 'v...
1	Maps516996	https://www.tripadvisor.com/Profile/Maps516996	July 3, 2022	My daughters and I thoroughly enjoyed our full...	5.0	TZ917HUF8HP1	X9ENIHJ8DJZ37R8PLM	['thoroughly', 'full', 'day', 'tour', 'avila',...
2	Diana M	https://www.tripadvisor.com/Profile/dianam770	February 28, 2022	Very nice tour of Avila and Segovia. The guide...	4.0	TZ917HUF8HP1	P7Z22GGTQUX1LXMMX0	['tour', 'avila', 'segovia', 'guida', 'knowled...
3	Dantecass	https://www.tripadvisor.com/Profile/Dantecass	October 30, 2022	I'll start with the "cons" because I felt ther...	3.0	TZ917HUF8HP1	NIWFY6KEORG7U3OW9U	['start', 'cons', 'felt', 'mora', 'cons', 'pro...
4	neillay51	https://www.tripadvisor.com/Profile/neillay51	October 30, 2022	The trip was not well organised. The microphon...	2.0	TZ917HUF8HP1	MC8LA0P1WZH36OK01R	['trip', 'well', 'microphone', 'receiving', 'd...
5	Jennie H	https://www.tripadvisor.com/Profile/721jennieh	October 4, 2022	.Great cities to visit. Buses very comfortable...	3.0	TZ917HUF8HP1	9NI5P9R4TSWM12QVVZ	['cities', 'visit', 'buses', 'comfortable', 'g...
6	Malcolm	https://www.tripadvisor.com/Profile/MKTMS	October 15, 2022	Supply earpieces that can be inserted comforta...	1.0	TZ917HUF8HP1	5O0QKB0RD8WPF33WI1	['inserted', 'comfortably', 'canal', 'ones', '...
7	Quiz	https://www.tripadvisor.com/Profile/cbquiz	May 1, 2022	Rafael did an excellent job with this tour he ...	5.0	TZ917HUF8HP1	0NIDO5U6H2TRNNLKMX	['excellent', 'job', 'tour', 'knowledgeable', ...
8	Lina	https://www.tripadvisor.com/Profile/3grayboots	September 20, 2022	We had a great time touring Ávila and Segovia ...	5.0	TZ917HUF8HP1	Y80FCAVOU5JCKWKVAA	['great', 'touring', 'ávila', 'segovia', 'tour...
9	Ruijian He	https://www.tripadvisor.com/Profile/Ruijian	June 8, 2022	5/5 for our tour guide Rafa who was amazing at...	4.0	TZ917HUF8HP1	54NXUIL44L0816MFWA	['tour', 'guida', 'rafa', 'amazing', 'job', 'd...

Figure 5.18. Sample of Trip Advisor Madrid Dataset

Madrid has reviews from the year of 2014 until 2022 with a total of 361 experiences reviewed 15165 times by 12606 users, which gives a number of 1,20 reviews per user and 42,0 reviews per experience.

Experience ID	Number of Reviews	Total Score
TZ917HUF8HP1	903	4330.0
XYF21MJK6I92	742	3532.0
5P6YOPJRGXX6	629	3065.0
BL0BFMAJVCCB	617	2814.0

In 5.19a is represented the best experiences and in the figure 5.19b we have the engagement of the users, where the top 4 users are responsible for 0,29% of total reviews.



(a) Experiences with best scores in Trip Advisor Madrid Dataset

(b) Users with the largest reviews done in Trip Advisor Madrid Dataset

Figure 5.19. Exploration of Madrid Dataset

UserName	Number of Reviews
David	13
David M	12
Karen B	10
John M	9

Chapter 6

Hybrid Recommendation System: The Top G Algorithm

As the algorithm responsible for the recommendation is a hybrid recommender engine, which means it is composed by a model based approach, followed by a memory based one.

6.1 Part I: Model Based

For the first part of the algorithm, the python library Top2Vec [4] was used. The benefits of its usage were the automatic finding of the number of topics, no requirements for a stop-word list, no need for stemming or lemmatization, and the presence of built-in functions which helped with the search for documents, topics, and word vectors.

The primary assumption made by the Top2Vec algorithm is that semantic similar documents indicates an underlying topic, so the first step lies to create joint embedding of documents and word vectors using Doc2Vec, Universal Sentence Encoder or BERT Sentence Transformer. 6.1

Once the documents and words are embedded in the vector space with lower dimensionality, with the help of the method UMAP, represented in the figure **??**, the algorithm finds dense clusters of documents and identify which words make those documents clustered together. 6.3.

After the analysis is done by searching for the dense areas in the vector space with the method HDBSCAN, each dense area calculates their centroid of document vectors in original dimension and then define it as the topic vector 6.4.

Once the topic vectors are calculated, the search turned to find the n-closest word vectors to the resulting topic vector and the words that attracted the documents together inside a topic are called the topic words. 6.5.

The idea behind the Top2Vec approach applied to the touristic review dataset is that, authors of similar reviews tend to have similar opinion about visited places. So, the first point is to create topics in order to identify hubs of common users, based on their opinion.

The usage of the reviews to train the embedding model and creating the topics, is different from creating them using the users, as a sort of pre-defined profile for each user. The idea behind the Top2Vec is to group reviews based on context of the written content, which means that, one user can be assigned to different topics at the same time, and it would give a different spectrum of the user activity on the social network.

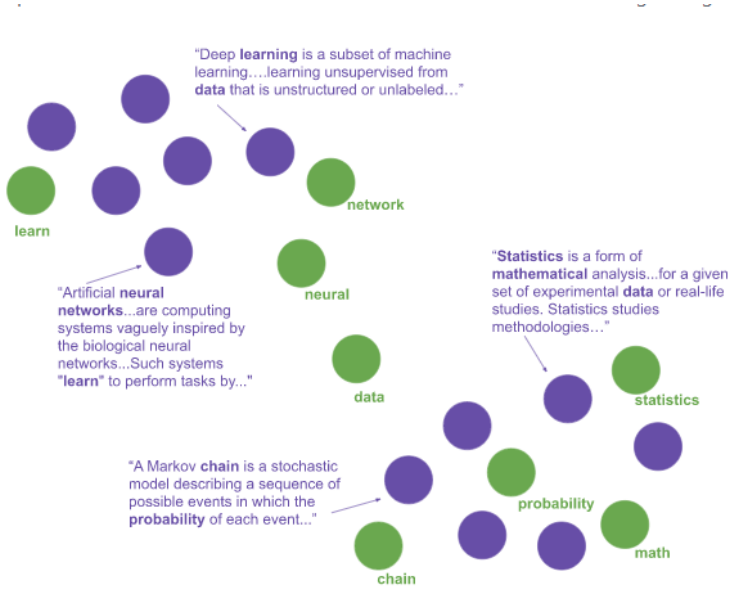


Figure 6.1. Embedded Document and Word Vectors [4]

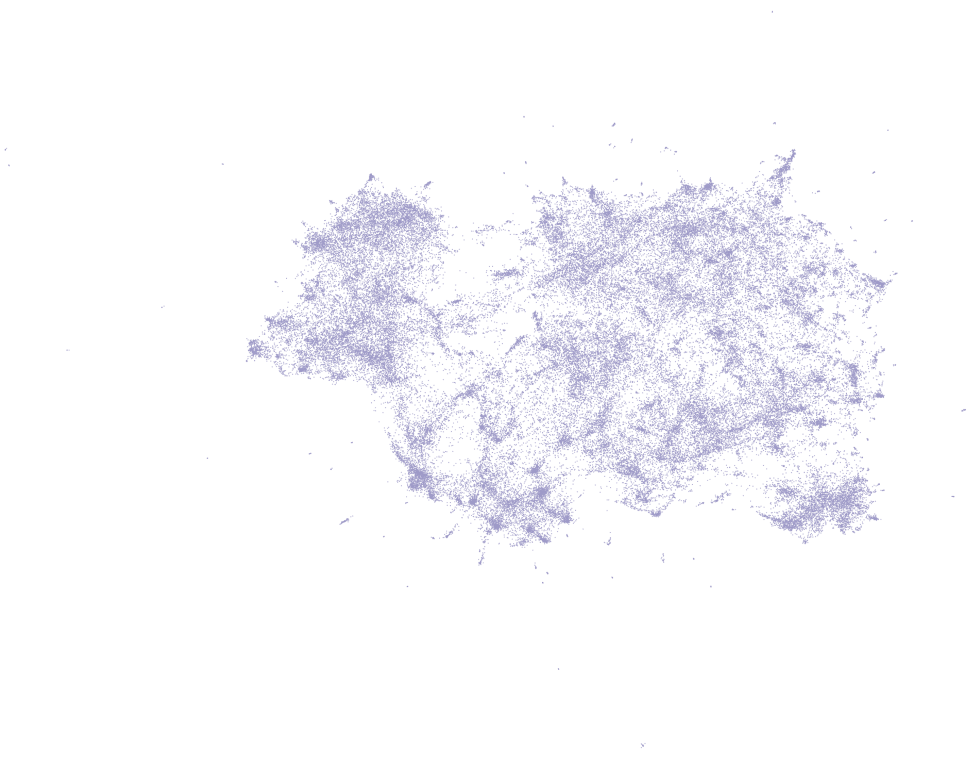


Figure 6.2. Document Vectors in High Dimensional Space [4]

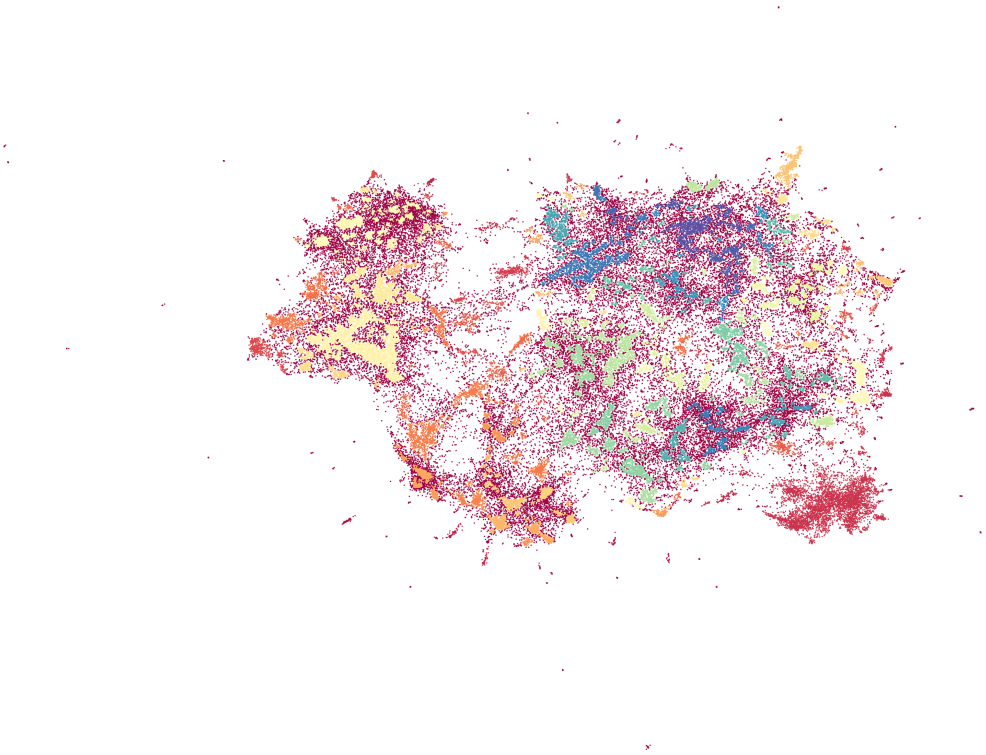


Figure 6.3. Dense Areas of Documents with HDBSCAN [4]

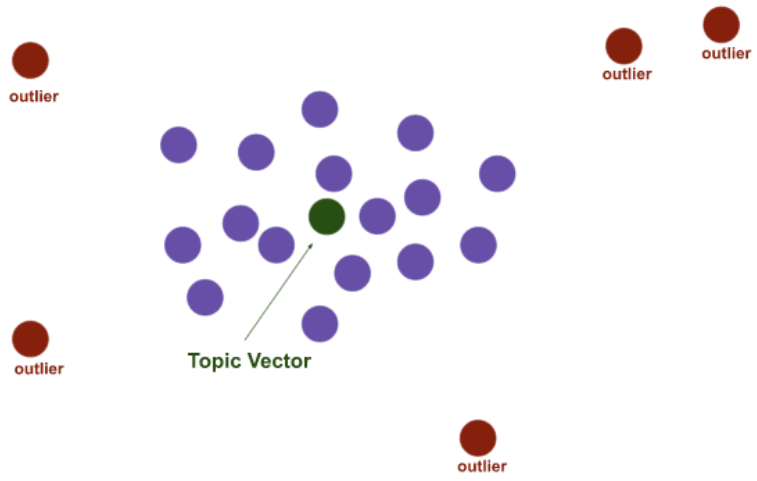


Figure 6.4. Calculating the Topic Vector [4]

The fact that a user can belong to different hubs at the same time creates the possibility of a way bigger list of recommendations and avoid part of the cluster specificity problem.6.6

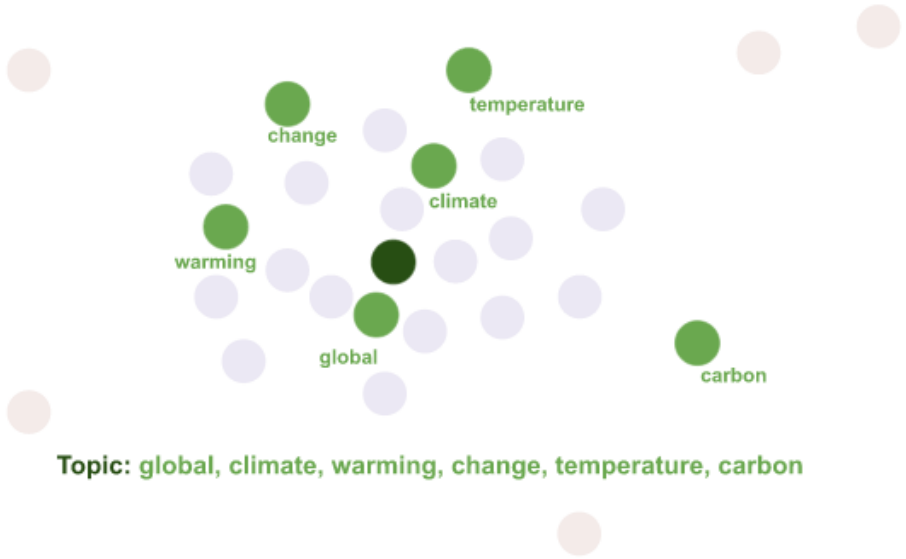


Figure 6.5. Getting the Topic Words [4]

After the creation of the topics, an auxiliary structure for the identification of the relationship between user, places, and reviews is defined by the following code 6.7.

Once the auxiliary structures are ready, the algorithm 6.8 identify internal hubs of recommendation for each topic and assign them to the users inside the topics, taking in consideration two rules:

- The user shouldn't have reviewed the following recommended place.
- The same place couldn't be recommended twice for the same user inside the cluster.

By the end of the topics processed by the algorithm, the recommendation from different topics are collected based on each user, and the partial recommendation list is created.

6.2 Part II: Memory Based

If the algorithm counted only on the model based approach, the tendency would be that, by the time, the topics wouldn't have a big variety and the recommendation list of the users would take a long time to be significantly updated with new destinations.

To introduce of some randomness in the algorithm, we use the "Likes dataset" 5 to extract some insights from the interests of the user.

Before feed the dataset into the algorithm, the likes are encoded into different categories. The idea is to pre-process the reaction of the users into different groups, so the interaction can be filtered into positive and negative one. This step, represented in 6.10 is crucial to give a trustful recommendation.

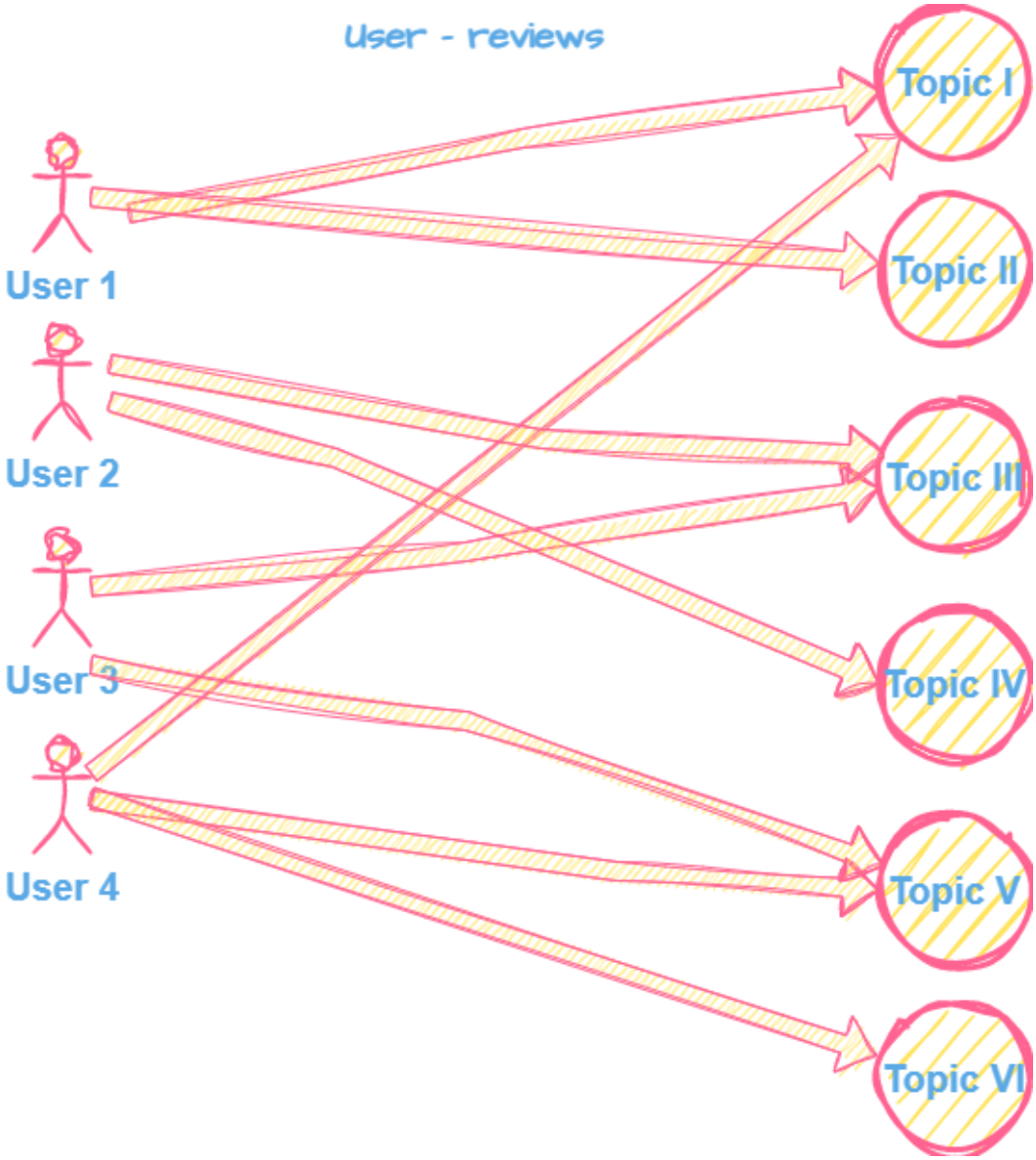


Figure 6.6. User Distribution Over Topics

The idea is that, each user have his own hub based on what he had interact with. Once those hubs are done for each user, a filter is applied, to keep only the positive reactions towards reviews, in order to take into consideration only the good feedbacks. 6.11

The result of this memory based algorithm is a list of destinations positively evaluated by the user based on the description of different users of the network. Once the new partial recommendation list is done, another filter is applied to filter possible duplicates.

▾ Dictionary: User - Reviews

```
[ ]  1 key_user = df['userId'].unique()
     2 user_review_dict=dict()
     3 for i in key_user:
     4   out = df[df['userId']==i]
     5   review_id = out['id']
     6   list_reviews = list(review_id)
     7   user_review_dict[i]=list_reviews
```

▾ Dictionary: Place - Reviews

```
[ ]  1 key_place = df['placeId'].unique()
     2 place_review_dict=dict()
     3 for i in key_place:
     4   out = df[df['placeId']==i]
     5   place_id = out['id']
     6   list_places = list(place_id)
     7   place_review_dict[i]=list_places
```

▾ Dictionary: User - Places

```
[ ]  1 key_user_place = df['userId'].unique()
     2 user_place_dict=dict()
     3 for i in key_user_place:
     4   out = df[df['userId']==i]
     5   places_id = out['placeId']
     6   list_places_user = list(places_id)
     7   user_place_dict[i]=list_places_user
```

Figure 6.7. Auxiliary Dictionaries of the Dataset

6.3 Part III: Hybrid Weighted Configuration

To obtain the final recommendation for each user, the model based recommendation list and the memory based recommendation list are combined in a weighted configuration, as mentioned in the chapter 7.1.3.

For a matter of practicality, the linear combination is done with a unitary angular coefficient and the combined results are again filtered to avoid the occurrence of duplicated values on the recommendation list for each user. 6.12 Finally, the recommendation given to a user is sorted from his own recommendation list.

Exploring Topics

```
[ ]  1 def get_place_and_user_from_cluster(review_ids):
      2   places_of_cluster=list()
      3   users_of_cluster=list()
      4   reccomendations_per_user = dict()
      5   for doc in review_ids:
      6     places_of_cluster.append((df[df['id']== doc]['placeId'].values[0]))
      7     users_of_cluster.append((df[df['id']== doc]['userId'].values[0]))
      8   places_of_cluster = list(dict.fromkeys(places_of_cluster))
      9   users_of_cluster = list(dict.fromkeys(users_of_cluster))
     10   for user in users_of_cluster:
     11     places_visited_for_user_x = user_place_dict[user]
     12     possible_places_to_visit = list(set(places_of_cluster) - set(places_visited_for_user_x))
     13     reccomended_places=random.sample(possible_places_to_visit,10)
     14
     15     recommendation_list = reccomended_places
     16     recommendation_list=list()
     17     for recommendation in reccomended_places:
     18       try:
     19         val = df_locations[df_locations['placeId'] ==recommendation]['city'].values[0]
     20       except:
     21         val='nan'
     22       recommendation_list.append(val)
     23     recommendation_list = list(dict.fromkeys(recommendation_list))
     24     try:
     25       recommendation_list.remove('nan')
     26     except:
     27       recommendation_list
     28     if user in reccomendations_per_user.keys():
     29       reccomendations_per_user[user].append(recommendation_list)
     30     else:
     31       reccomendations_per_user[user]=recommendation_list
     32   return reccomendations_per_user
```

Figure 6.8. Model Based Recommendation Algorithm

In case of further interactions inside the network, or new users, the algorithm will update the already existent model taking in consideration the chronology of the data. The reason behind this is because, as noticed in the development of the recommender system of Netflix [15], the time of the data is important for the recommendation, but not because of the destination itself, but because of the tendencies of market and the taste of users, which are biased by season and year.

Memory Based algorithm part

```
[14]  1 likes_path = "/content/drive/MyDrive/thesis-data/kuriu-likes-2022-06JUN-18.json"
      2 df_likes = pd.read_json(likes_path)
```

```
1 df_likes
```

	documentId	lastUpdate	authorId	userName	type
0	622c3b8ac992ca28f33c7f21	2022-03-19T07:56:20.868Z	606	DaleOnTravel	Useful
1	620d214425398565dadb6984	2022-02-16T16:14:34.113Z	180	giusepigno	Useful
2	620d214425398565dadb6984	2022-03-02T12:50:07.738Z	531	Lontra_Scivolosa	Useful
3	621e537d82ed1d1eef3a394f	2022-03-01T18:25:49.536Z	531	Lontra_Scivolosa	Useful
4	621e537d82ed1d1eef3a394f	2022-03-03T09:17:48.329Z	550	Giada_Mille_Esperienze	Useful
...	...	...	...	...	...
2537	62767583e44220507a306b0e	2022-05-14T15:57:12.829Z	304	EliVi	Inspire
2538	62767583e44220507a306b0e	2022-06-01T10:54:45.668Z	688	EsoticaTravel	Useful
2539	6273a0a9dd67f6198e5ac192	2022-05-05T21:13:07.761Z	399	MCExp	Useful
2540	61e0146111b3bf4d3185ad5d	2022-01-14T16:58:21.739Z	480	Mary90	Useful
2541	622f3149ef19af4233396c98	2022-03-14T19:35:50.657Z	550	Giada_Mille_Esperienze	Useful

2542 rows × 5 columns

Figure 6.9. Likes Dataset

PRE PROCESSING LIKES

```
[ ]  1 def lower_list_type(l):
     2     if type(l)!=list:
     3         return []
     4     res = [x.lower() for x in l]
     5     return res
     6
     7 def encoding_types(df):
     8   one_hot_enc_categories = df["type"].apply(lambda x: lower_list_type(x))
     9   df["type2"] = one_hot_enc_categories
    10   mlb = MultiLabelBinarizer()
    11   schede_with_categories = df.join(pd.DataFrame(mlb.fit_transform(df.pop('type2')),index=df.index,columns=mlb.classes_))
    12   return schede_with_categories
```

Figure 6.10. Category Encoding

▾ Create the list of likes for users, search and create recommendations

```
[ ]  1 rec_memory_based=dict()
     2 for index, row in df_likes.iterrows():
     3   if (row['bored']!='1'):
     4     doc_id=row['documentId']
     5     like_author=row['authorId']
     6     author_liked=df_reviews.loc[df_reviews['id']==doc_id]['userId'].values[0]
     7     places_from_liked_author=user_place_dict[author_liked]
     8     try:
     9       places_from_liked_author.remove('nan')
    10     except:
    11       places_from_liked_author
    12     places_from_liked_author = list(dict.fromkeys(places_from_liked_author))
    13     if like_author in rec_memory_based.keys():
    14       rec_memory_based[like_author].append(places_from_liked_author)
    15     else:
    16       rec_memory_based[like_author]=places_from_liked_author
```

```
[ ]  1 def from_id_to_places(rec_memory_based,df_locations):
     2   output_dic=dict()
     3   for key in rec_memory_based:
     4     lista1=rec_memory_based[key]
     5     aux=list()
     6     for element in lista1:
     7       try:
     8         val=df_locations[df_locations['placeId']==element]['city'].values[0]
     9       except:
    10         val='nan'
    11       aux.append(val)
    12
    13     aux = list(dict.fromkeys(aux))
    14     try:
    15       aux.remove('nan')
    16     except:
    17       aux
    18     output_dic[key]=aux
    19   return output_dic
```

Figure 6.11. Memory Based Algorithm

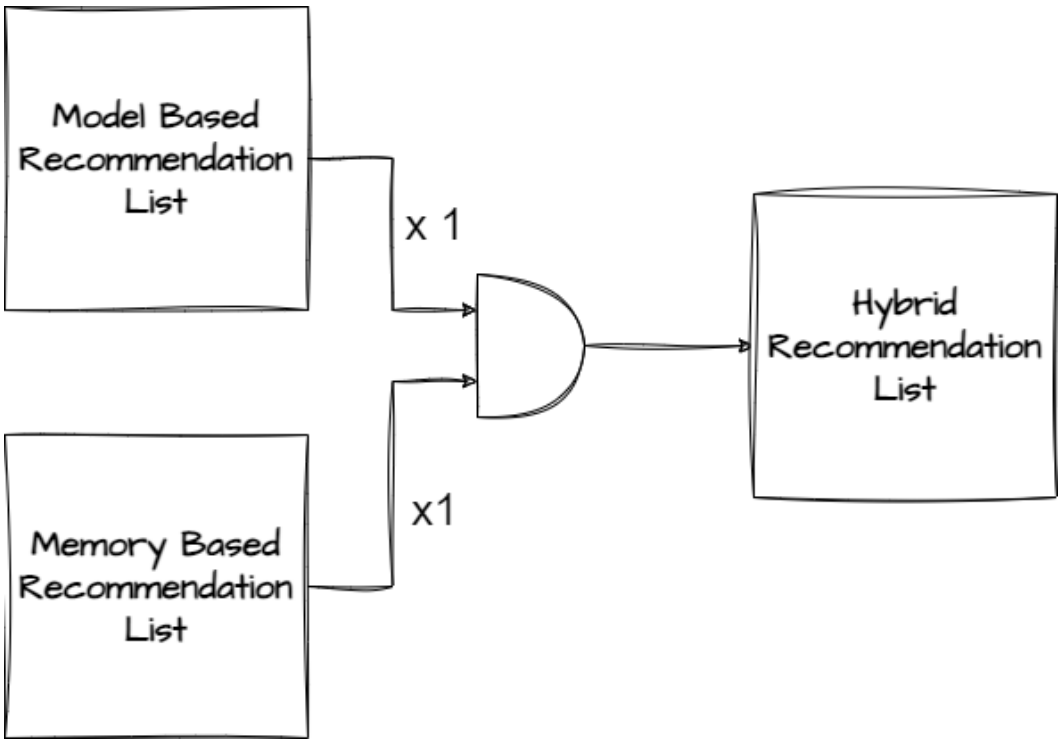


Figure 6.12. Hybrid Recommendation List

Chapter 7

Results

7.1 Methodology

7.1.1 Data Collection

We collected data from two sources: the Easy Tour platform, where written reviews from attractions and likes (clickstream data) were collected based on users interactions, and the Trip Advisor platform, where only the user written reviews were available.

For the Easy Tour Dataset, as described in the section 5, 5724 reviews and 2541 clickstream data were collected. In other hand, in the Trip Advisor Dataset we have a total of 7 different data from the cities of Paris, Rome, Madrid, New York, Berlin, London, and Barcelona, counting a total of 285854 reviews.

7.1.2 Qualitative Analysis

By the result of exploring the different datasets, we attain to develop a qualitative analysis by evaluating the behavior of the recommendation system in the 4 most engaged users in the datasets, so we can see the performance of the algorithm by calculating the Hit Rate for those users.

Using the most engaged users from the dataset for evaluating the recommender system algorithm can provide valuable insights into the quality of recommendations. Engaged users are those who have actively participated in social interactions and have left reviews and ratings about their travel experiences. These users are likely to be more representative of the average user behavior and preferences and can help to evaluate the effectiveness of the recommendation system algorithm in providing relevant and personalized recommendations.

By selecting the four most engaged users from the dataset, we are essentially selecting a sample that is likely to represent the most active and opinionated users. This can help you identify potential biases in the algorithm and evaluate its effectiveness in providing recommendations that cater to the diverse preferences and interests of different users.

Qualitative analysis of the interactions and reviews from these four users can help you identify patterns in their behavior and preferences. This can help in understanding whether the recommendation system algorithm is effectively capturing the different factors that influence user preferences, such as the content written in the reviews or the content of the reviews which they interact with.

Furthermore, by comparing the recommendations provided by the algorithm to the actual behavior and preferences of these four users, we can assess the accuracy

and relevance of those recommendations. This can help in identifying areas for improvement and fine-tune the algorithm to better serve the needs and preferences of the users.

7.1.3 Quantitative Analysis

As we are using a Hybrid approach in the Easy Tour Dataset and, because of the lack of clickstream data for the Trip Advisor Dataset, we use a Model Based approach. In order to evaluate the performance of the algorithm, we use several metrics to evaluate the effectiveness of our recommendation system based on internal topic metrics, such as the topic diversity, coherence and IRBO and external topic metric, the Hit Rate.

Topic Diversity:

```
[ ]
def topic_diversity_score(topics, topk):
  unique_words = set()
  for topic in topics.values():
      unique_words = unique_words.union(set(topic))
  td = len(unique_words) / (topk * len(topics))
  return td
```

```
[ ]
def get_count(t:tuple):
  return t[1]

def topic_diversity(docs, labels=None, topics_list=None, topk=25, print_scores=False):
  if topics_list is None:
    topics_topk = get_topics(labels, docs, topk)
  else:
    topics_topk = dict()
    for id,t in enumerate(topics_list):
      topics_topk[id] = t[:topk]
  topics_diversity = dict()
  for k1 in topics_topk.keys():
    words = topics_topk[k1]
    not_unique_words = 0
    for w in words:
      for k2 in topics_topk.keys():
        if k2 == k1:
          continue
        if w in topics_topk[k2]:
          not_unique_words += 1
          break
    percentage = (len(words) - not_unique_words)/len(words)
    topics_diversity[k1] = percentage
  td_score = topic_diversity_score(topics_topk, topk)
  if print_scores:
    print("Topic diversity score is: {}".format(td_score))
  return topics_diversity, td_score
```

Figure 7.1. Topic Diversity Code

Topic diversity is a metric that measures the extent to which a recommendation system suggests products from a wide range of topics. We calculated topic diversity by dividing the number of unique recommended products in the topic modeling algorithm by the total number of recommended products, 7.1. A higher topic diversity score indicates that the recommendation system is suggesting products from a wide range of topics.

Coherence:

```
[ ]  1 def coherence(docs, dictionary, labels=None, topics_list=None, print_scores=False):
     2   if topics_list is None:
     3     topics = get_topics(labels, docs)
     4     topics_keys = list(topics.keys())
     5     topics_list = [topics[k] for k in topics_keys]
     6
     7   c_npmi = CoherenceModel(topics=topics_list,texts=docs,dictionary=dictionary,coherence="c_npmi")
     8   c_umass = CoherenceModel(topics=topics_list,
     9                            texts=docs,
    10                            dictionary=dictionary,
    11                            coherence="u_mass")
    12   c_uci = CoherenceModel(topics=topics_list,
    13                            texts=docs,
    14                            dictionary=dictionary,
    15                            coherence="c_uci")
    16   c_v = CoherenceModel(topics=topics_list,
    17                            texts=docs,
    18                            dictionary=dictionary,
    19                            coherence="c_v")
    20   #results
    21   coherence_c_npmi, coherence_c_uci, coherence_c_umass, coherence_c_v = c_npmi.get_coherence(), c_uci.get_coherence(), c_umass.get_coherence(), c_v.get_coherence()
    22   coherence_c_npmi_topics, coherence_c_uci_topics, coherence_c_umass_topics, coherence_c_v_topics = c_npmi.get_coherence_per_topic(), c_uci.get_coherence_per_topic(), c_umass.get_coherence_per_topic(), c_v.get_coherence_per_topic()
    23   if print_scores:
    24     print("c_npmi: {}\nc_uci: {}\nc_umass: {}\nc_v: {}".format(coherence_c_npmi, coherence_c_uci, coherence_c_umass, coherence_c_v))
    25     print("-"*100)
    26     print("c_npmi for each topic: {}\nc_uci for each topic: {}\nc_umass for each topic: {}\nc_v for each topic: {}".format(coherence_c_npmi_topics, coherence_c_uci_topics, coherence_c_umass_topics, coherence_c_v_topics))
    27   return coherence_c_npmi, coherence_c_uci, coherence_c_umass, coherence_c_v
```

Figure 7.2. Coherence Code

Coherence is a metric that measures the extent to which the recommended products are related to each other. We calculated coherence by analyzing the similarity between the recommended products based on their attributes, for our particular case, the description of reviews written by the users. A higher coherence score indicates that the recommended products are more closely related to each other.

IRBO:

```
[ ]  1 def get_word2index(list1, list2):
     2     words = set(list1)
     3     words = words.union(set(list2))
     4     word2index = {w: i for i, w in enumerate(words)}
     5     return word2index
     6
     7 def irbo(topics, weight=0.9, topk=10):
     8     if topk > len(topics[0]):
     9         raise Exception('Words in topics are less than topk')
    10     else:
    11         collect = []
    12         for list1, list2 in combinations(topics, 2):
    13             word2index = get_word2index(list1, list2)
    14             indexed_list1 = [word2index[word] for word in list1]
    15             indexed_list2 = [word2index[word] for word in list2]
    16             rbo_val = rbo.rbo(indexed_list1[:topk], indexed_list2[:topk], p=weight)[2]
    17             collect.append(rbo_val)
    18         return 1 - np.mean(collect)
```

Figure 7.3. IRBO Code

IRBO (Item Recommendation-Based Optimization) is a metric that evaluates the effectiveness of a recommendation system based on the proportion of recommended items that are actually selected by users. We calculated IRBO by dividing the number of recommended products that were clicked on or highly rated by the user by the total number of recommended products. A higher IRBO score indicates that the recommendation system is suggesting products that are more relevant to users.

Hit Rate:

```
[ ]  1 def hit_rate(rec_all,rec_80):
     2   all_keys1=list(rec_all.keys())
     3   all_keys2=list(rec_80.keys())
     4   common=len(list(set(all_keys1).intersection(all_keys2)))
     5   hr_list=list()
     6   for key in rec_80:
     7     if key in all_keys1:
     8       eighty = rec_80[key]
     9       all = rec_all[key]
    10       HR=len(list(set(all).intersection(eighty)))/len(all)
    11       hr_list.append(HR)
    12       print(HR)
    13   mean_hr_algorithm = sum(hr_list)/len(hr_list)
    14   return mean_hr_algorithm
```

Figure 7.4. Hit Rate Code

Hit Rate measures the proportion of recommended products that are effectively recommended to a user. We calculated Hit Rate by dividing the number of recommended products presented in the trained algorithm with part of the data by the total number of recommended products with all the data applied to the training for the algorithm. Using this metric, we can test how robust is the recommendation algorithm according to the size of the sample data used to train it.

By using these metrics in our analysis, we were able to gain a sympathetic understanding of the effectiveness of our recommendation system. Specifically, we evaluated the system based on its ability to suggest products from a wide range of topics, the coherence of its recommendations, its ability to suggest products that were actually relevant to users, and the proportion of recommended products that were clicked on or highly rated by the user.

7.2 Qualitative Results

The evaluation metrics for the models were calculated in two different environments: the first one with 100% of data and the second with 80%. The idea was to observe the behavior of the model in respect to the size of the data used to train and in a successive step, evaluate the recommendation of the whole dataset, using the Hit Rate measure. The Hit Rate analysis, calculated with 25 recommendations per user, was done also to compare the model based approach algorithm to the hybrid approach in the case of the Easy Tour Dataset and Trip Advisor Dataset.

7.3 Qualitative Results

In order to provide a qualitative result of the algorithm, the top users, mentioned in the section 5, of each dataset were analyzed and evaluated using the Hit Rate. The goal of this analysis was to identify how robust the algorithm is to give recommendations based on 80% of the total data. Also, it is important to mention that

Dataset - 100% of Data	Number of Topics	Topic Diversity	Coherence	IRBO
Berlin	106	0.262	0.446	0.974
Barcelona	278	0.212	0.484	0.987
Madrid	109	0.249	0.466	0.968
London	527	0.188	0.493	0.992
Paris	291	0.223	0.497	0.987
Rome	560	0.177	0.497	0.993
New York	376	0.198	0.470	0.990
Easy Tour Model	51	0.752	0.475	0.995
Easy Tour Hybrid	51	0.752	0.475	0.995

Table 7.1. Results of model evaluation metrics with 100% of Data

Table 7.2. Results of model evaluation metrics with 80% of Data

Dataset - 80% of Data	Number of Topics	Topic Diversity	Coherence	IRBO
Berlin	77	0.284	0.436	0.972
Barcelona	209	0.2395	0.500	0.985
Madrid	81	0.283	0.472	0.967
London	430	0.193	0.502	0.991
Paris	223	0.233	0.495	0.984
Rome	463	0.182	0.508	0.992
New York	295	0.209	0.468	0.988
Easy Tour Model	34	0.791	0.459	0.994
Easy Tour Memory	34	0.791	0.459	0.994
Easy Tour Hybrid	34	0.791	0.459	0.994

Table 7.3. Hit Rate Comparison

Dataset	Hit Rate Model Based	Hit Rate Memory Based	Hit Rate Hybrid
Berlin	0.531	0.768	0.645
Barcelona	0.432	0.759	0.619
Madrid	0.487	0.742	0.589
Rome	0.383	0.730	0.588
Paris	0.431	0.743	0.609
London	0.401	0.784	0.660
New York	0.429	0.763	0.644

all Hit Rates were calculated using the number of 25 recommendations per user.

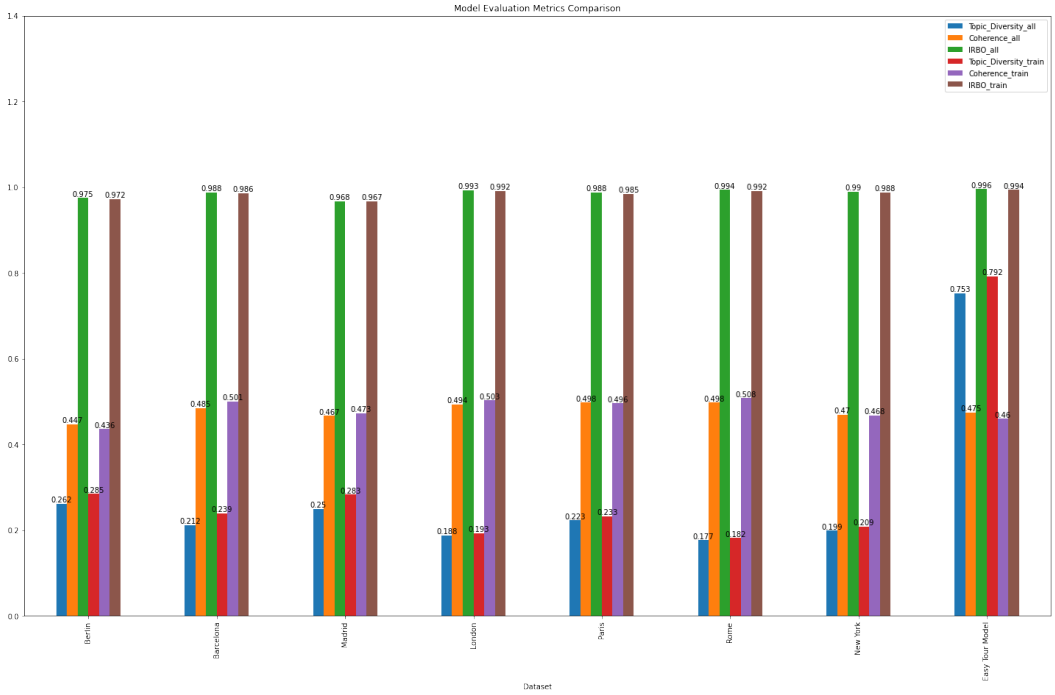


Figure 7.5. Results of model evaluation metrics

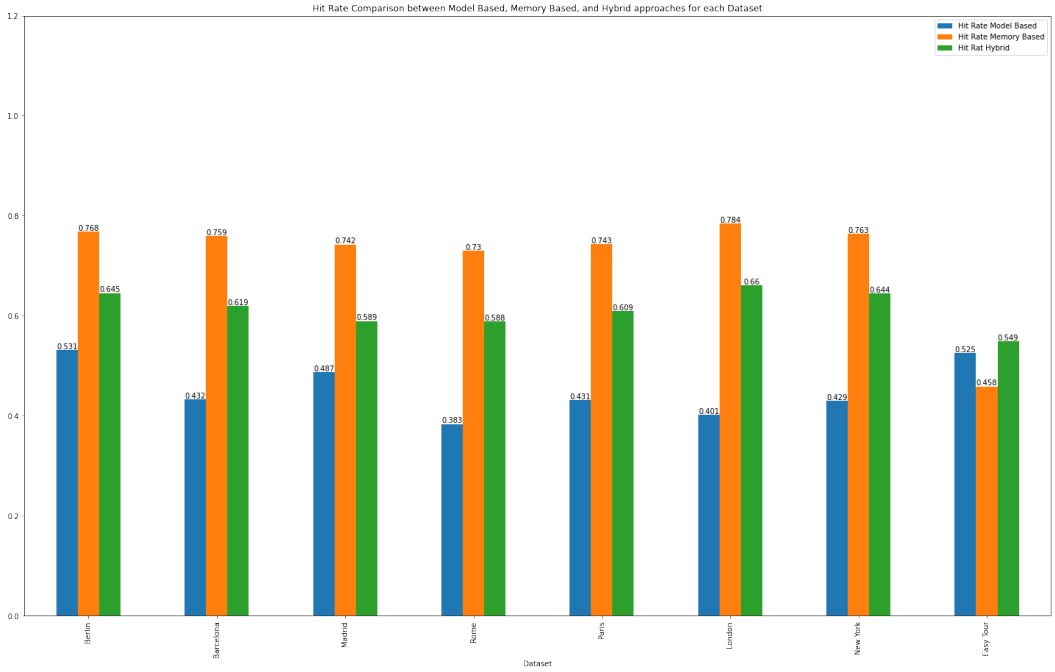


Figure 7.6. Hit Rate results for different recommendation system engines in different datasets

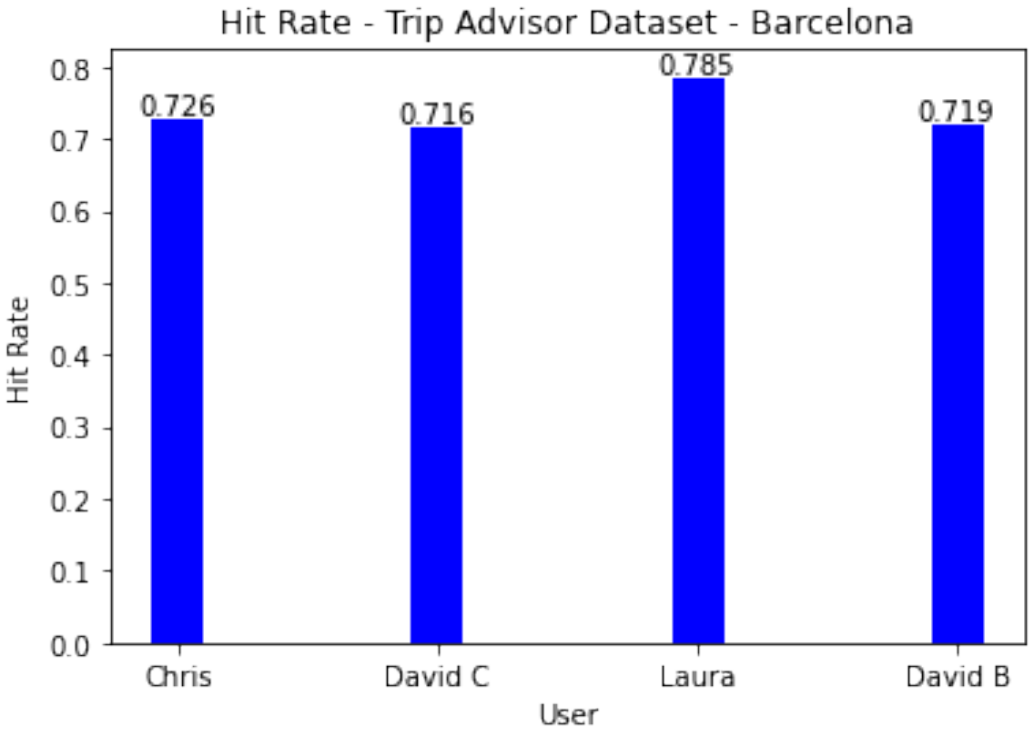


Figure 7.7. Hit Rate of the 4 most engaged users of the Trip Advisor Barcelona Dataset

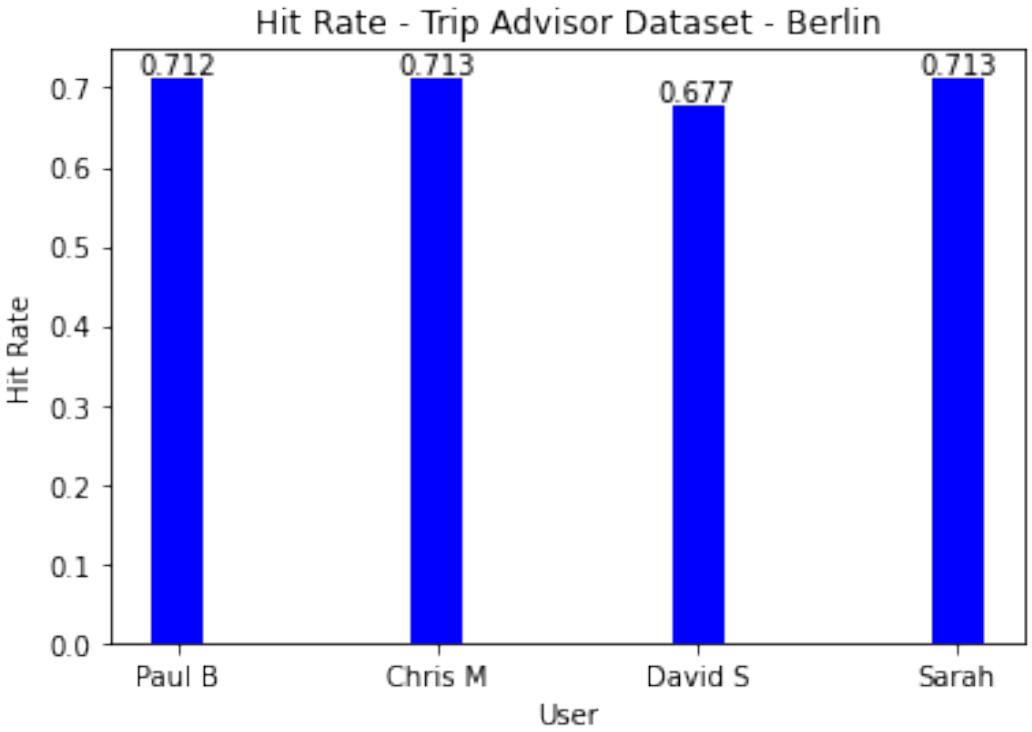


Figure 7.8. Hit Rate of the 4 most engaged users of the Trip Advisor Berlin Dataset

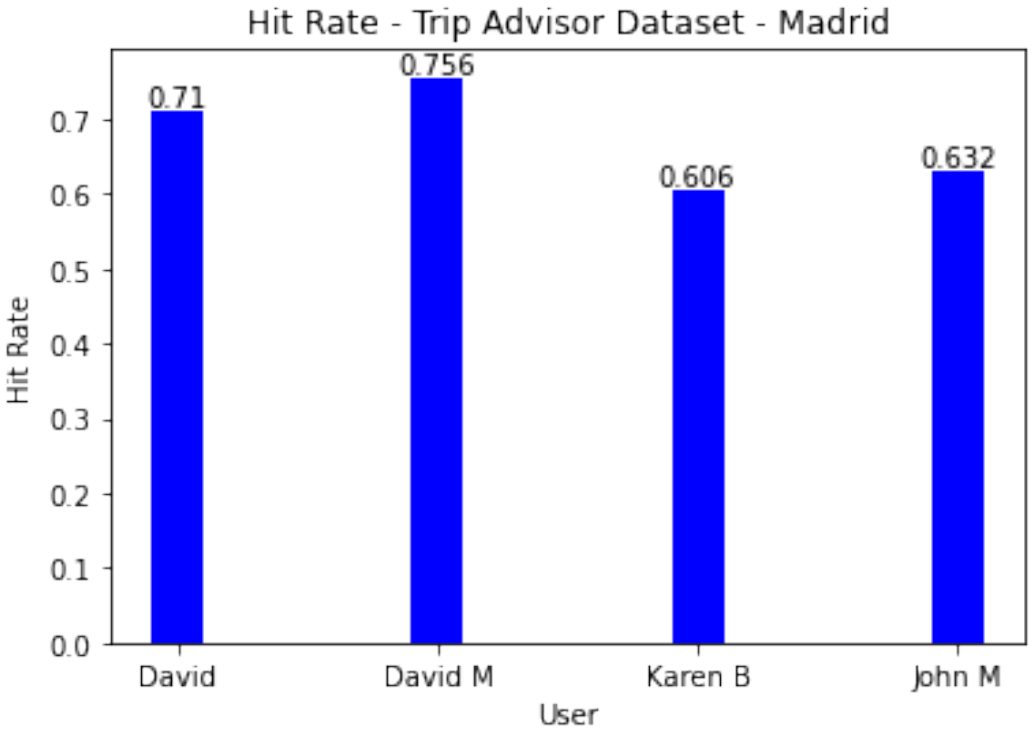


Figure 7.9. Hit Rate of the 4 most engaged users of the Trip Advisor Madrid Dataset

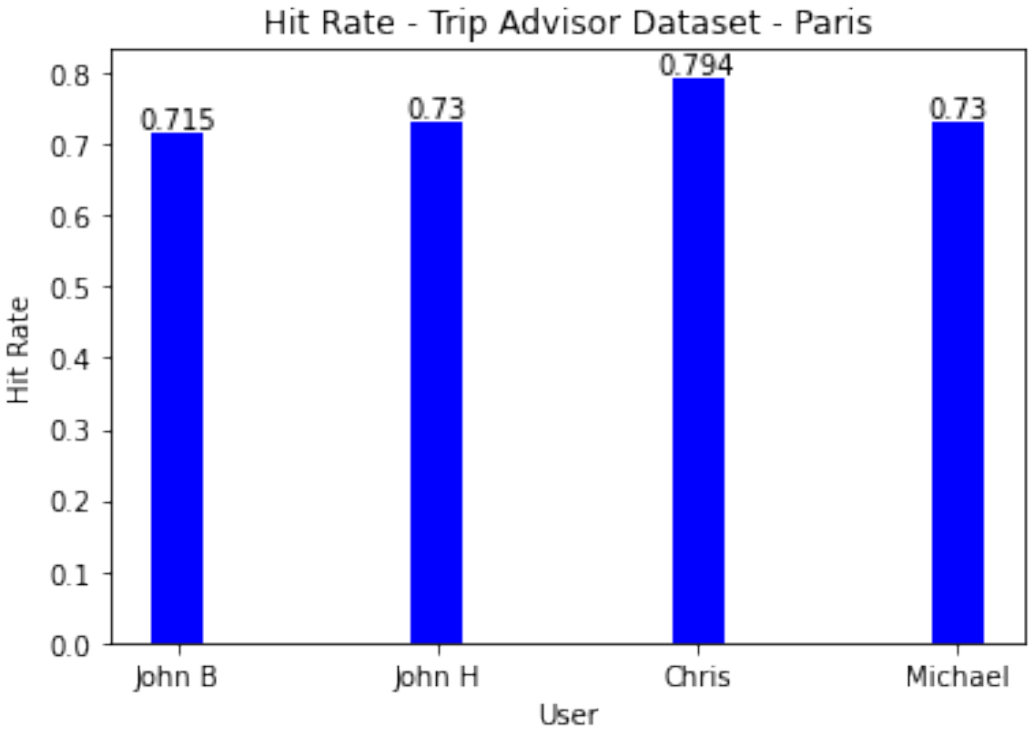


Figure 7.10. Hit Rate of the 4 most engaged users of the Trip Advisor Paris Dataset

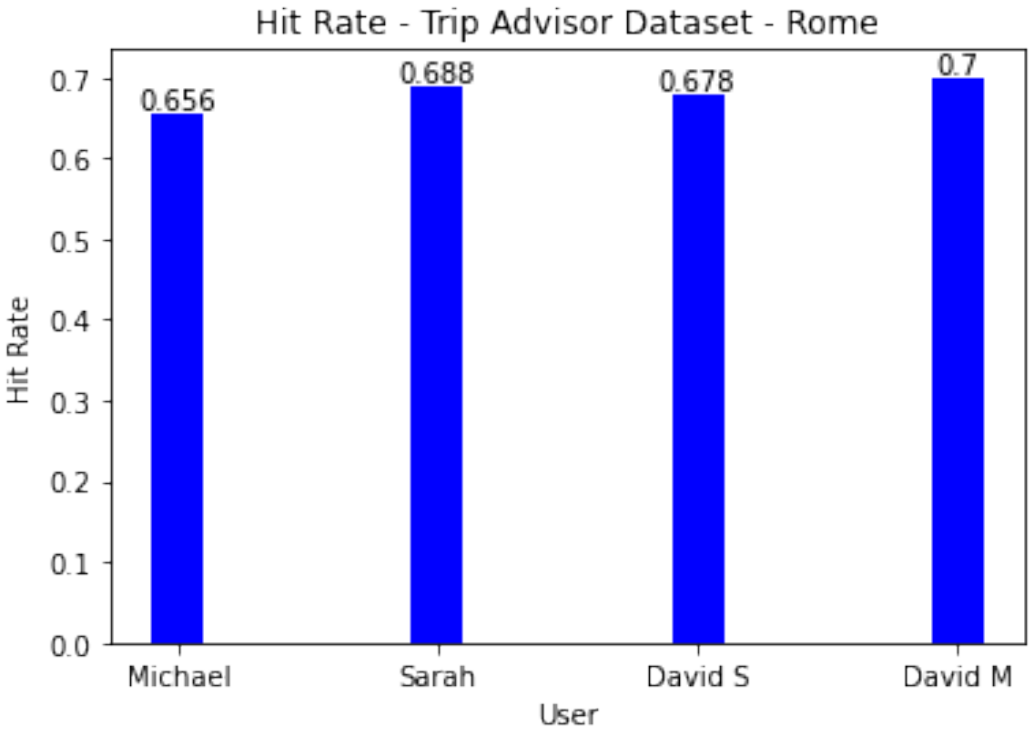


Figure 7.11. Hit Rate of the 4 most engaged users of the Trip Advisor Rome Dataset

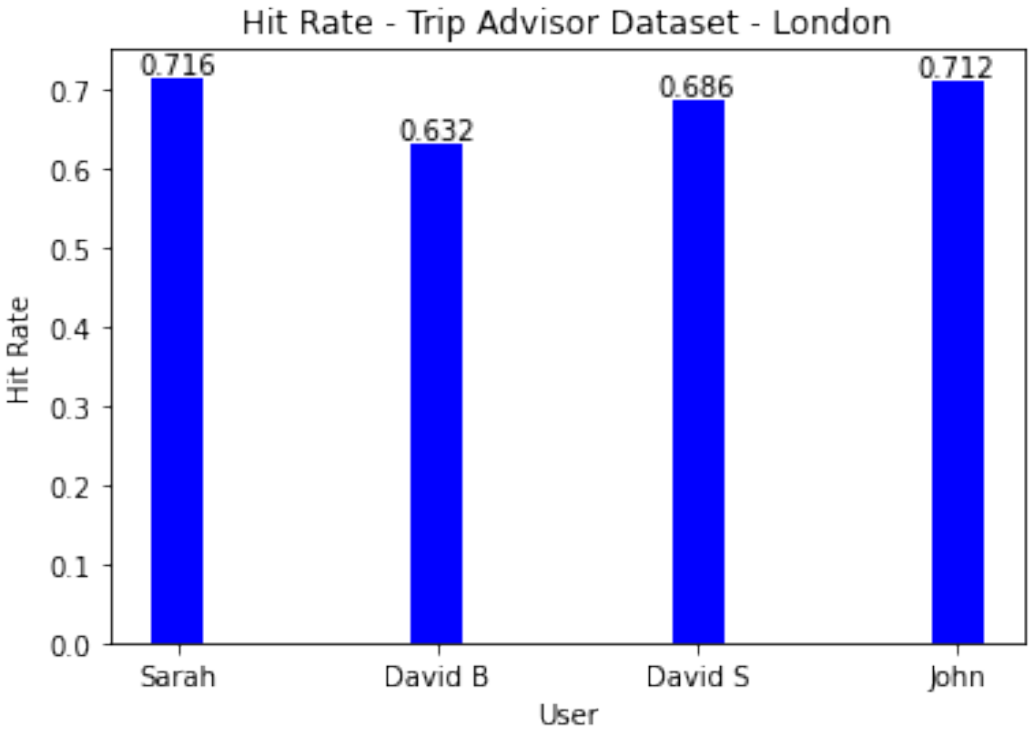


Figure 7.12. Hit Rate of the 4 most engaged users of the Trip Advisor London Dataset

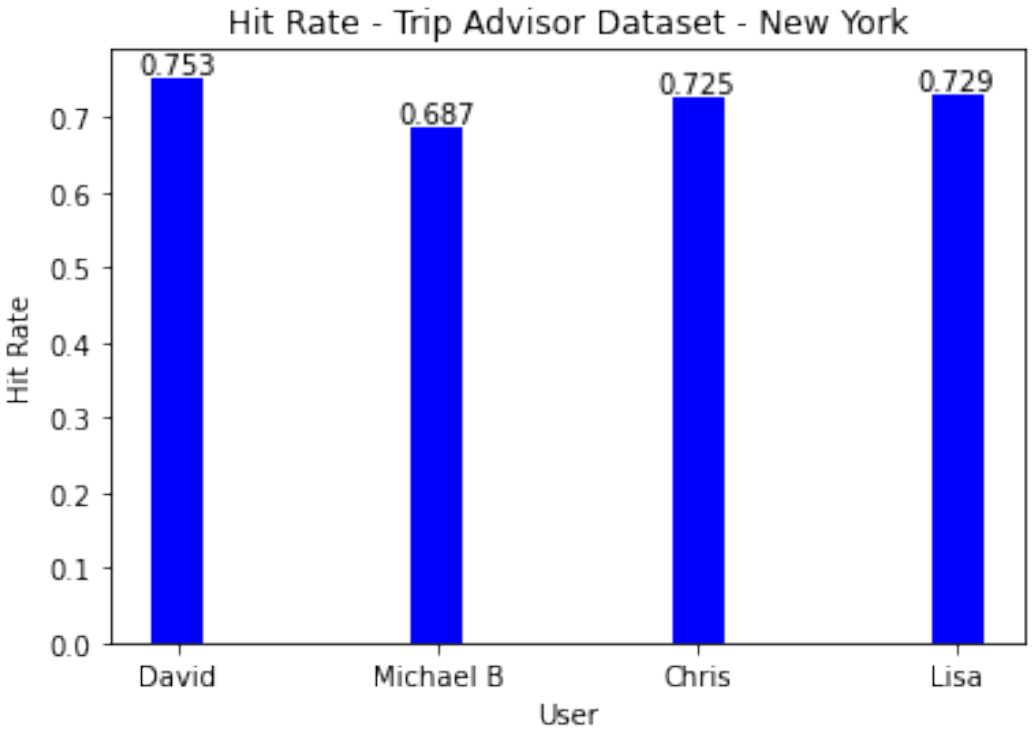


Figure 7.13. Hit Rate of the 4 most engaged users of the Trip Advisor New York Dataset

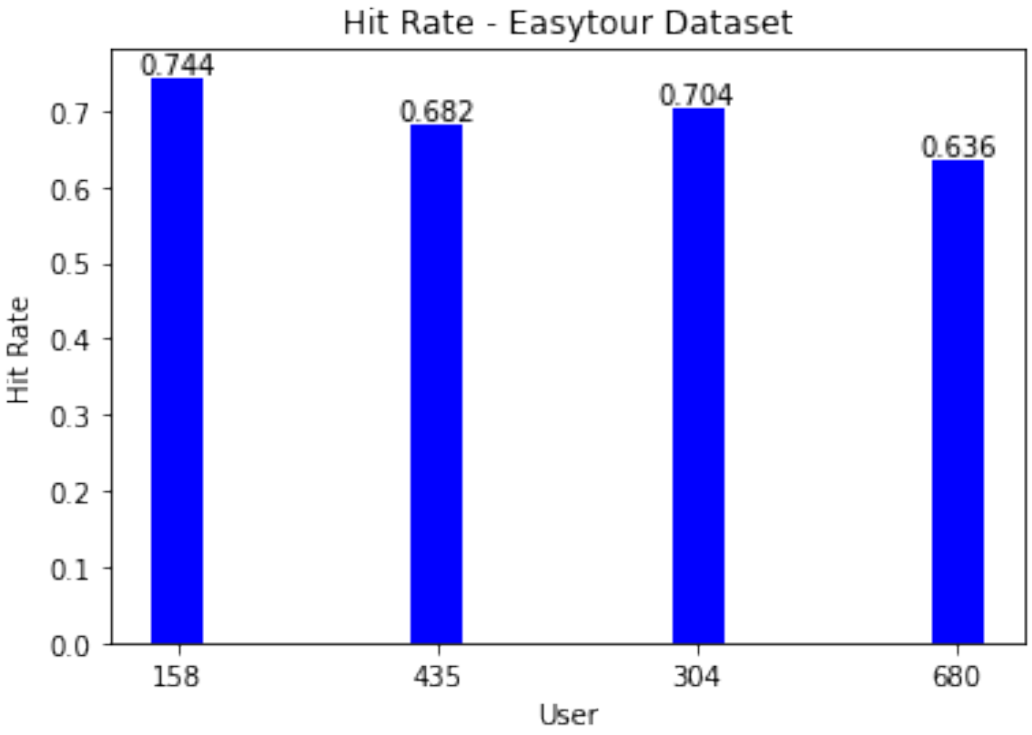


Figure 7.14. Hit Rate of the 4 most engaged users of the Easy Tour Dataset in Model Based Approach

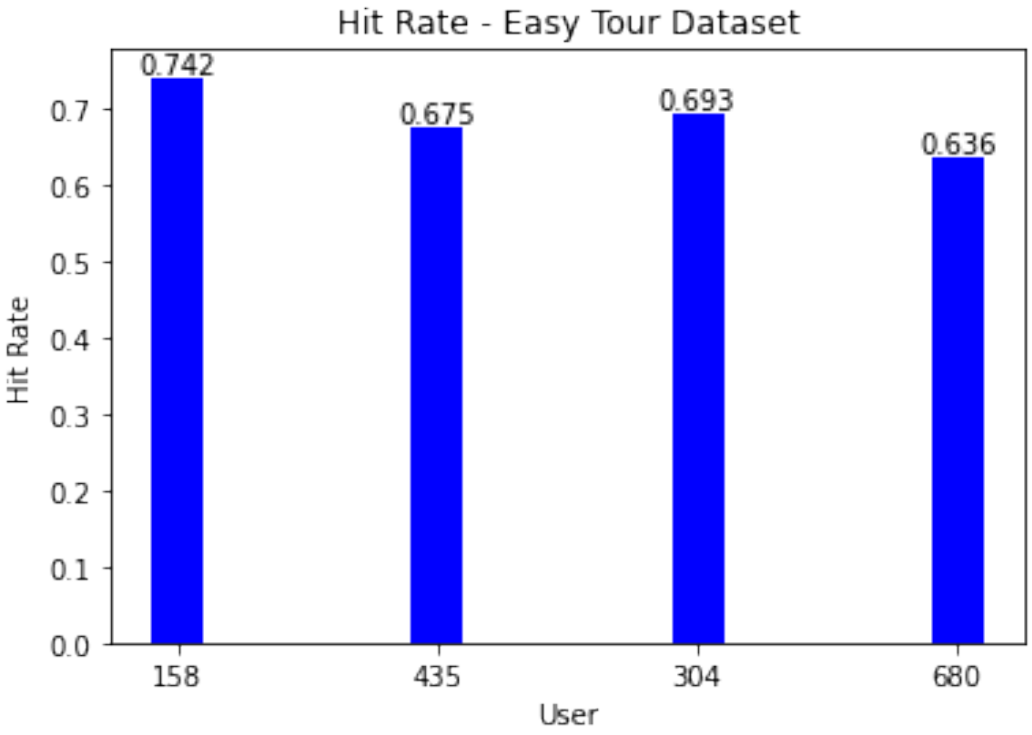


Figure 7.15. Hit Rate of the 4 most engaged users of the Easy Tour Dataset in Hybrid Approach

Chapter 8

Conclusions

The main purpose of the thesis is to demonstrate that recommender systems are an important tool to be implemented in tourism digital platforms. In the network reality of millions of destinations, the use of a recommender engine can improve its customer's digital experience, making it personalized. By doing so, the probability of buying a product or visiting a specific place shown by the platform would increase, and the users would be even more hooked in the social network platform.

The results obtained by the proposed algorithm were evaluated in both datasets and, analyzing the metrics, it was observed that the overall Hit Rate was higher in the hybrid approach when compared to the model based approach when applied in the same dataset. The result was 4.7% better and when confronted with the singular Hit Rate of the most engaged users of the Dataset, the surprise was that the individual Hit Rate was higher in the pure model based approach.

Despite the fact that for some Datasets, the memory based approach had a greater Hit Rate, the results for the recommendation were given by a naive recommendation approach, which leads to recommend the obvious. So, even though the result was expected for the Trip Advisor Dataset, the same phenomenon was not seen in the Easy Tour. The reason behind this was because of the nature of the data and how it was distributed.

As mentioned in the section 3, the quality of the recommendation depends on the quantity of data generated by the user, which means that, a user, with a poor interaction in the network, won't have an accurate recommendation in the beginning of the use of the platform. In fact, by analyzing the dichotomy of the results (overall and individual for the Easy Tour Dataset), we can conclude that the hybrid approach, by adding the memory based component based on the user interactions in the network, was able to give a better contribution for the recommendation of the users with fewer reviews written, and consequently made the quality of the recommendation higher for the whole set of users.

When training the model with a reduced data, the algorithm tended to perform poorly. In the case of the evaluation metrics, the observed results were that, for smaller datasets, as the Easy Tour, the model generated fewer topics, and consequently a higher Topic Diversity rate. But, in the other hand, the bigger datasets haven't the advantage of a bigger cohesion or IRBO. The results shown that the cohesion and the IRBO were kept, independent of the dataset analyzed and the Hit Rate for both datasets, Trip Advisor and Easy Tour were quite similar for the pure model based approach.

Bibliography

[1] Charu C Aggarwal. Content-based recommender systems. In *Recommender systems*, pages 139–166. Springer, 2016.

[2] Enrique Amigó, Damiano Spina, and Jorge Carrillo-de Albornoz. An axiomatic analysis of diversity evaluation metrics: Introducing the rank-biased utility metric. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, pages 625–634, 2018.

[3] Dimo Angelov. Top2vec - documentation. https://github.com/ddangelov/Top2Vec.

[4] Dimo Angelov. Top2vec: Distributed representations of topics. *arXiv preprint arXiv:2008.09470*, 2020.

[5] Joeran Beel and Stefan Langer. A comparison of offline evaluations, online evaluations, and user studies. 12 2014.

[6] Jesus Bobadilla, Santiago Alonso, and Antonio Hernando. Deep learning architecture for collaborative filtering recommender systems. *Applied Sciences*, 10(7):2441, 2020.

[7] Ginevra Carbone and Gabriele Sarti. Etc-nlg: End-to-end topic-conditioned natural language generation. *Italian Journal of Computational Linguistics*, 6:61–77, 12 2020.

[8] Jeffery chiang. 7 types of hybrid recommendation system. https://medium.com/analytics-vidhya/7-types-of-hybrid-recommendation-system-3e4f78266ad8.

[9] Adji B Dieng, Francisco JR Ruiz, and David M Blei. Topic modeling in embedding spaces. *Transactions of the Association for Computational Linguistics*, 8:439–453, 2020.

[10] Roman Egger and Joanne Yu. A topic modeling comparison between lda, nmf, top2vec, and bertopic to demystify twitter posts. *Frontiers in Sociology*, 7, 2022.

[11] Larry Hardesty. The history of amazon's recommendation algorithm. https://www.amazon.science/the-history-of-amazons-recommendation-algorithm.

[12] Julia Hirschberg and Christopher D Manning. Advances in natural language processing. *Science*, 349(6245):261–266, 2015.

[13] Adam Louly. Nlp based recommender system without user preferences. https://towardsdatascience.com/nlp-based-recommender-system-without-user-preferences-7077f4474107.

[14] Linyuan Lü, Matúš Medo, Chi Ho Yeung, Yi-Cheng Zhang, Zi-Ke Zhang, and Tao Zhou. Recommender systems. *Physics reports*, 519(1):1–49, 2012.

[15] Netflix. How netflix's recommendations system works. https://help.netflix.com/en/node/100639.

[16] Alexandre Escolà Nixon. Building a memory based collaborative filtering recommender. https://towardsdatascience.com/how-does-collaborative-filtering-work-da56ea94e331.

[17] João Pedro. Understanding topic coherence measures. https://towardsdatascience.com/understanding-topic-coherence-measures-4aa41339634c.

[18] Baptiste Rocca. Introduction to recommender systems. https://towardsdatascience.com/introduction-to-recommender-systems-6c66cf15ada.

[19] sagar pundir. Top2vec: New way of topic modelling. https://towardsdatascience.com/top2vec-new-way-of-topic-modelling-bea165eeac4a.

[20] Oren Sar Shalom, Haggai Roitman, and Pigi Kouki. Natural language processing for recommender systems. In *Recommender Systems Handbook*, pages 447–483. Springer, 2022.

[21] Peimeng Sui. Personalized recommendations for experiences using deep learning. https://www.tripadvisor.com/engineering/personalized-recommendations-for-experiences-using-deep-learning/.

[22] Katarzyna Anna Tarnowska and Zbigniew Ras. Nlp-based customer loyalty improvement recommender system (clirs2). *Big Data and Cognitive Computing*, 5(1):4, 2021.

[23] Emmanouil Vozalis and Konstantinos G Margaritis. Analysis of recommender systems algorithms. In *The 6th Hellenic European Conference on Computer Mathematics & its Applications*, pages 732–745, 2003.

[24] Hanna M Wallach. Topic modeling: beyond bag-of-words. In *Proceedings of the 23rd international conference on Machine learning*, pages 977–984, 2006.

[25] Benjamin Wang. Ranking evaluation metrics for recommender systems. https://towardsdatascience.com/ranking-evaluation-metrics-for-recommender-systems-263d0a66ef54.

[26] Liang Wu and Mihajlo Grbovic. How airbnb tells you will enjoy sunset sailing in barcelona? recommendation in a two-sided travel marketplace. SIGIR '20, New York, NY, USA, 2020. Association for Computing Machinery.

[27] Shuai Zhang, Lina Yao, Aixin Sun, and Yi Tay. Deep learning based recommender system: A survey and new perspectives. *ACM Computing Surveys (CSUR)*, 52(1):1–38, 2019.