FEDERAL STATE EDUCATIONAL INSTITUTION

OF HIGHER EDUCATION

NATIONAL RESEARCH UNIVERSITY

HIGHER SCHOOL OF ECONOMICS

**Saint Petersburg School of Economics and Management**

**Department of Management**

Denisova Nadezhda

Trush Georgy Dmitrievich

**Recommendation System for Travellers Based on TripAdvisor.com Data**

Bachelor’s thesis

In the field 38.03.02 ‘Management’

Educational programme ‘Management’

|  |  |
| --- | --- |
| Reviewer  Position, degree  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Initials Last name | Academic supervisor  Associate Professor  E.A. Antipov  Initials Last name |

Saint Petersburg 2020

**Table of contents**

[**Abstract** 3](#_Toc41836788)

[**1. Introduction** 4](#_Toc41836789)

[**2. Theoretical foundation** 10](#_Toc41836790)

[**2.1 Definition of main terms and concepts** 10](#_Toc41836791)

[**2.2 Recommender system approaches: core algorithms and respective applications** 12](#_Toc41836792)

[**2.3 Overview of popular travel recommender system approaches** 21](#_Toc41836793)

[**2.4 Major recommender system issues and their common solutions** 25](#_Toc41836794)

[**2.5 Performance assessment metrics: definitions and applications** 28](#_Toc41836795)

[**3. Statement of the research question** 34](#_Toc41836796)

[**4. Methodology** 37](#_Toc41836797)

[**5. Description of the results** 43](#_Toc41836798)

[**5.1 Matrix factorisation models** 43](#_Toc41836799)

[**5.2 K-nearest-neighbours models** 44](#_Toc41836800)

[**5.3 LDA model** 45](#_Toc41836801)

[**5.4 Discussion of the results** 49](#_Toc41836802)

[**6. Conclusion** 52](#_Toc41836803)

[**Reference list** 54](#_Toc41836804)

[**Appendix 1** 63](#_Toc41836805)

[**Appendix 2** 64](#_Toc41836806)

[**Appendix 3** 65](#_Toc41836807)

[**Appendix 4** 66](#_Toc41836808)

[**Appendix 5** 68](#_Toc41836809)

[**Appendix 6** 69](#_Toc41836810)

[**Appendix 7** 70](#_Toc41836811)

[**Appendix 8** 71](#_Toc41836812)

[**Appendix 9** 72](#_Toc41836813)

[**Appendix 10** 73](#_Toc41836814)

# **Abstract**

Recommender systems have become an indispensable part of everyone’s daily online experience, providing users with the personalised suggestions of what items to purchase next or which films to choose for the evening. Thus, recommender systems have massively boosted user engagement and overall quality of service for online businesses across a wide variety of application domains. However, only a few domains have received all of the initial attention from academia and businesses alike, while the majority of other ones, with the tourism domain being the chief one among them, have remained comparatively underexplored. In particular, the machine learning algorithms based on latent factors have been widely recognized for their efficiency in making accurate film recommendations from the extremely sparse data of film ratings as well as for their effectiveness in discovering coherent patterns within the semantic data of electronic documents. Taking into account the results from the film and e-document domains, the present study adopts these latent factor models to be applied on the previously untested and highly sparse TripAdvisor data of user ratings and reviews for the London’s tourist attractions, with the goal of enabling more accurate recommendations in the tourism domain. This is achieved by means of developing and launching a fully functional prototype of a hybrid filtering recommender system on the core basis of the top-performing matrix factorisation algorithm of the FunkSVD (collaborative filtering) as well as on the supplementary basis of a topic modelling technique of the LDA (content-based filtering). In order to identify the highest performing collaborative filtering model, this research compares three of the most computationally accessible matrix factorisation algorithms against each other as well as against the common benchmark methods of k-nearest neighbours, ultimately revealing that the FunkSVD model produces the most accurate rating predictions according to the MAE metric. The LDA model, in its turn, discovers the optimal number of 6 general types of attractions according to the interests of London travellers, as opposed to the default 15 categories from TripAdvisor. Finally, the hybrid recommender system for tourists is developed to, firstly, alleviate the cold start problem by prefiltering the London attractions according to the user’s choice of one of the 6 general types, and secondly, rank those attractions based on the user’s input ratings.

*Keywords:* travel recommender system, recommendation system, TripAdvisor, tourist attractions, hybrid filtering, latent factor models, matrix factorisation, latent Dirichlet allocation.

# **1. Introduction**

Historical background. Ever since the mid-1970s, when the concept of a recommender (or recommendation) system was given its first definition in scientific literature, the respective research field has been quickly taking shape as a combination of several closely-related academic disciplines, such as: applied mathematics, computer science, behavioural economics theory and, not the least important, business studies, which have been key in developing the practical solutions and applying the technical research findings in order to eventually yield commercial value (Sharma & Singh, 2016). The first actual implementation of a recommender system in a business environment was achieved at the Xerox Corporation’s Palo Alto Research Centre in 1992. According to the reports, this novel recommender system called Tapestry was able to much more efficiently manage a user’s stream of incoming electronic mail documents, news wire stories and articles, based on which documents other users had already marked as interesting (Goldberg, Nichols, Oki, & Terry, 1992). As a result, over the next few years the realisation of the enormous potential of the recommender system research for commercial usage has quickly attracted many more investments from large business corporations looking to raise efficiency of internal operations and communication, which enabled the evermore accelerated research developments to take place. Thus, for the widely recognised reason of being able to massively enhance the online experience of consumers, nowadays recommender systems play a crucial part in the daily functioning of the biggest internet platforms, such as Netflix, Amazon or TripAdvisor, by presenting users with personalised suggestions of movies for the evening, relevant shopping items or much sought-after travel destinations (Sharma & Singh, 2016). Along with the continuous increase in computational power and the developments of new machine learning algorithms as well as the digitalisation of many traditional business models, novel ways of improving recommendations are still being actively researched.

Introduction of key terms. The notion of a recommender (or recommendation) system in its general meaning can be defined as a cutting-edge software tool that takes user data as input and produces suggestions of items which are most likely to be of interest to that particular user (Adomavicius & Tuzhilin, 2005). More specifically, a recommender system is, in essence, an information filtering algorithm that often functions on the basis of several either statistical or machine learning predictive models and produces personalised item recommendations for users, by predicting how likely a given item will be preferred by a given user based on the various kinds of available data about the online behaviour of users and the descriptions of items. Moreover, the ultimate products of any recommender system are the personalised item recommendations that are commonly arranged in the form of a limited list of items, which are ranked according to their relevance to a given user and which are very often provided as an output in an online service application that the user interacts with through his or her web browser (Adomavicius & Tuzhilin, 2005).

Research gap and questions. Firstly, the most potent and robust recommender systems, which are the ones exploiting hybrid filtering, have been frequently developed on the basis of the most accessible and the most abundant online data of film ratings, product reviews and e-documents’ texts. Thus, the current study attempts to exploit the less convenient data of user ratings and reviews from TripAdvisor, that way contributing to the much less popular application domain of hybrid recommender system research, namely – the tourism domain. Secondly, as the user-item matrix data of travel reviews is expected to be highly sparse, the present research chooses to adapt the specific algorithms of the latent factor models, which have been widely celebrated for being able to efficiently reduce the dimensionality of sparse data, producing the most accurate results among the other computationally comparable machine learning algorithms. Thirdly, the formidable variety of the matrix factorisation algorithms, although proven highly efficient on the cases of extremely sparse data from other application domains of item recommendations, such as film recommendations, have not yet been tested on the similarly sparse TripAdvisor data of user ratings for the tourist landmarks and experiences. Fourthly, since most studies of recommender systems based on the TripAdvisor data have been heavily reliant on the multi-criteria ratings of hotels and restaurants, significantly fewer research papers have been published exploring how the semantic analysis of textual reviews can be adapted for the purposes of improving the robustness of a hybrid recommendation system in the tourism domain, where such user reviews are copious. Thus, the central aim of this research is to construct a hybrid filtering recommender system for travellers, based on the latent factor models and exploiting the TripAdvisor data for tourist attractions from the single most popular tourist destination in Europe – the city of London, UK, by means of answering the following main and supplementary research questions:

– What is the highest performing collaborative filtering algorithm among the matrix factorisation models that is able to produce the most accurate recommendations, based on the TripAdvisor data of user ratings for the London’s tourist attractions?

– Does the text recognition technique of the latent Dirichlet allocation reveal coherent categories of user preferences to be employed for the content-based filtering of new users, based on the TripAdvisor data of user reviews for the London’s tourist attractions?

Central goal of the research. The present research thesis is directed towards adopting the highly efficient and widespread latent factor models from the prior recommender system research studies based on the data of film and e-documents domains to be applied on the previously untested TripAdvisor data of tourist attractions, with the ultimate goal of combining the results of the latent factor models into a hybrid filtering recommender system for personalised recommendations of tourist attractions.

Purpose of the research. This research aims to incorporate aspects of descriptive and exploratory research purpose types. The proposed research sees it necessary to implement the descriptive purpose aspect at the preliminary stage, in relation to the collection and descriptive analysis of the sampled user data, as well as the exploratory purpose aspect as the core focus of the research, in terms of identifying the novel configuration of a hybrid recommendation system for travellers as well as of training the machine learning models in the subsequent trial-and-error process accompanied by a series of adjustments toward the most accurate versions of the chosen prediction algorithms.

Objectives of the research thesis are identified as the following:

a) First and foremost, the scientific contribution to the extensive research area of hybrid recommender systems for travellers, which consists in demonstrating the unique application benefits of latent factor models on the previously untested tourist attractions data from the TripAdvisor web platform.

b) And secondly, the development and the eventual launch of a fully functional consumer prototype of a hybrid recommender system that can be tested by real online users, which may later further developed into a standalone online service application providing personalised recommendations on tourist attractions, which can be potentially adapted to be implemented into a larger recommendation system architecture of online platforms that provide travel-related services, such as the TripAdvisor platform.

Tasks that are set for the completion of the present research objectives are as follows:

a) Firstly, to identify the most commonly used types of TripAdvisor data from the range of research publications on the subject of travel recommender systems, assembled from the existing trustworthy research papers that have been included into the scientific citation databases of the Web of Science and Scopus platforms, in order to justify the choice of adapting the previously untested TripAdvisor data of tourist attractions for the goals of the present research paper;

b) Secondly, to web-scrape the user ratings and reviews data from the TripAdvisor platform specifically on the landmarks and experiences of London, UK, with the primary help of the Python parser libraries called Selenium and BeautifulSoup, as well as to describe and visualise the results of the respective data analysis, mainly in order to assess the degree of the data sparsity problem and make appropriate adjustments;

c) Thirdly, to train the previously reviewed range of the most popular and computationally accessible matrix factorisation algorithms of the FunkSVD, the SVD++ and the NMF as well as the common benchmark methods of user- and item-based k-nearest neighbours on the basis of the user-item interaction matrix that contains overall ratings for tourist attractions, in order to evaluate and compare the quality of their rating predictions on the unknown test users, based on the commonly used accuracy metric of MAE, that way justifying the ultimate choice of a single collaborative filtering algorithm that will be able to produce the most accurate recommendations for the system’s new users;

d) Fourthly, to test whether the application of the topic recognition algorithm of the Latent Dirichlet Allocation on the array of the users’ textual reviews for the collected sample of London’s tourist attractions reveals a more general, but more relevant list of attraction categories that is consistent with the interests of London travellers, as compared to the default list of 15 attraction types provided on the TripAdvisor platform;

e) Fifthly, to enhance the travel recommender system’s performance by executing the following tasks: solving the cold start problem for the actual new users by pre-filtering them according to the newly discovered list of attraction types; as well as ensuring the diversity of user recommendations by penalising the recommendation algorithm for suggesting the most popular attractions;

f) And finally, to develop a fully-fledged practical solution for the recommendations of London’s tourist attractions in the form of an interactive web-based service application, on the core basis of the single top-performing collaborative algorithm, which showed the highest accuracy of rating predictions according to the MAE metric, as well as on the supplementary basis of the content-based filtering of attractions according to the user’s preference of a certain one of the attraction categories, revealed by the LDA algorithm.

Structure of the thesis. The present research paper consists of the total number of 57pages (without appendices) and comprises a reference list of exactly 74citations of the highly referenced foreign research articles and conference reports, with the majority of them having been taken from the trustworthy databases of scientific research included on the Web of Science and Scopus platforms. The brief structural outline of the present research paper has been compiled of the following elements:

a) First and foremost, the academic literature review for the purposes of identifying the theoretical foundations and the scientific field contribution of the prior research as well as revealing the specific gap that the present research intends to contribute in filling;

b) Secondly, the statement of research questions and respective hypotheses and the justification of the choice of approaches by which the present thesis aims to arrive at the satisfying answers;

c) Thirdly, the methodology of the research which specifies and argues for the appropriability of the exact methods used at the data mining, data analysis and the machine learning stages;

d) Fourthly, the chronological record of the intermediate machine learning model outcomes and the respective justified adjustments as well as the description of the ultimate research results;

e) And finally, the concluding part of the paper that focuses on critically discussing the applicability and quality of the achieved results as well as the potential research possibilities of exploring further on in the set direction.

Professional relevance and significance of the research object. Since recommender systems are currently embedded into the majority of travel related online platforms, improvements in the performance of their algorithms can provide more accurate and diverse user recommendations and, thus, greatly benefit every party involved from consumers and companies that host these online platforms to the travel businesses that advertise and try to improve their product and service offers on such platforms. Moreover, since tourists can often feel incompetent to exploit the advanced search options or even completely lost when inundated with the abundance of information present online, the online service provider that solves the problem of information overload and recommends only the most relevant options of items is surely bound to drastically improve the engagement of its users, that way capturing and retaining much bigger numbers of active users, which inevitably indicates leveraging higher profits no matter whether the platform earns from direct item sales, like Amazon, from user subscriptions, like Netflix or from advertising of third party products, like YouTube. However, even though this research is focused on data in the travel domain, the recommender algorithm it aims to arrive at can potentially be used in other areas where user feedback is provided in the form of textual reviews and ratings.

Scope of the thesis. First and foremost, in order to make the collaborative recommender system algorithm more robust and avoid extreme data sparsity, the present research thesis employs the purposeful method of data sampling and ensures the collection of a representative data sample, by means of scraping all of the user ratings and reviews that have been posted in the course of the past year from February 2019 to January 2020 under every single one of approximately one thousand tourist attractions. Such sampling approach is the most convenient as the default sorting of TripAdvisor reviews is set according to their recency (and not popularity or rating level); as well as completely sufficient as it provides a representative sample of the most recent reviews of the platform’s active users, devoid of any season- or demographic-specific data. Furthermore, in sight of the extremely demanding computational conditions of processing millions of reviews, it is again reasonable to narrow down the focus to the single most popular city destination of Europe, such as London, UK, and scrape the data on its tourist attractions to create a dataset of the total magnitude of tens of thousands of user ratings and reviews.

Limitations of the research. The core limitation of any scientific research that employs machine learning methods is the computational complexity of the algorithms. Even though prediction algorithms that show accurate results might exist in theory, their convenient implementation is impossible due to the inadequacy of practical application cases in the computer science research as well as, much more importantly, due to the temporal and computational constraints of the present bachelor’s thesis. Secondly, since the data on user ratings will be narrowed down to the travel destinations of only a single city, the findings that will be uncovered as the result of applying the text recognition algorithm of the LDA on the user reviews for the London’s attractions will most probably turn out location-specific and, hence, not be applicable to other destinations. Finally, the limited scale and research framework of this bachelor’s thesis recognizably limits the practical applicability of the results, by only allowing to run and test models on the collected data sample, for which reason the launch of the recommender system online for actual users to try out might yield different accuracy results.

# **2. Theoretical foundation**

## **2.1 Definition of main terms and concepts**

The general meaning of the term “recommendation” in the present context of the recommender system research can be defined as a prediction of a given person’s preferences for a specific range of items, based on the preference history of that same user or of other users (Adomavicius & Tuzhilin, 2005). Furthermore, the most basic definition of the term “algorithm” can be formulated as a specific set of instructions that is often expressed in a computer (or programming) language to be executed by a certain computer programme. Thus, the concept of a recommender (or recommendation) system, which is central to this research thesis, can be, in essence, defined as an algorithm that often incorporates statistical as well as machine learning models in order to provide users with a recommendation list of items, which have been predicted to be the most likely to match the interests and other characteristics of these users (Lu et al., 2012).

One of the first ever descriptions of the mechanism that underlies modern recommender systems in academic literature dates back to the year of 1979. Rich (1979), in her paper on the modelling of “user stereotypes”, proposed her own version of a recommender system called Grundy, a virtual librarian that probed users with questions about their personalities and interests, that way associating users with various “stereotypes”, or collections of frequently encountered characteristics of particular types of users, and, thus, being able to recommend new books to users. In the case when the user rejected the recommendation, the system would start clarifying which of the stereotypes it based its recommendation on were unprecise about the user and, after the appropriate corrections were applied, once again attempted to make further recommendations. This cycle continued to repeat for as long as the user requested the system to further adjust the evaluations of his interests and to try recommending other books that they would be interested in reading (Rich, 1979).

Similarly, the key concept at the root of many modern recommendation systems is identified as: firstly, predicting the item ratings that users would very likely give to the yet unrated items; and then, presenting those users with tailored recommendation sets of the top relevant items, which they would be very likely to appreciate the most. Hence, the general task of any recommender system can be described as the creation of a predictive algorithm that separately or both at once considers the attributes of an item space as well as the attributes of a user space (those spaces being also known as profiles) in order to evaluate how well a specific item matches a particular user. User profiles are usually comprised of user IDs, users’ demographic information and a variety of ratings the user has assigned to different items, whereas the profile characteristics of items vary drastically, depending on the domain that the particular recommender system is going to be implemented in, for example, restaurants can be characterised by their location, average bill, overall rating, dominant type of cuisine, availability of al fresco dining and so on (Adomavicius & Tuzhilin, 2005).

Furthermore, the way the most primitive recommender algorithm determines that certain items are worth being recommended to a given user is directly connected to the concept of similarity. However, the core problem of the similarity-based recommendation systems is the choice of how exactly to define and to measure the similarity between the profiles of items or between those of users. In the instance when item ratings of users are explicitly available, a given pair of users can be identified as being similar if they tend to assign similar ratings to the items that both of them have rated. In this case user similarity can be measured using a correlation metric, such as the Pearson correlation coefficient. On the other hand, in the situation when there is no rating information available, a pair of users can be considered as being similar when they both viewed, liked or bought many of the same items. Such information can be either inferred from the structural properties of the users’ input data or gathered implicitly by tracking the online activity of those users as they visit the observed web sites. In addition, other external information such as the meta data of users’ attributes, social network tags and items’ content can be utilised to enhance the quality of a similarity estimate between given pairs of users or pairs of items (Lu et al., 2012).

Lastly, another basic concept that is crucial to understanding the general principles by which any recommender system operates is user interface, or UI for short. The widespread computer science term of user interface denotes the way in which a user is able to interact with any given computer system with that system also being able to communicate back, in particular, by means of input devices and software applications (Lu et al., 2012). Although one of the very first practical implementations of a recommender system – a virtual librarian named Grundy, which was described previously – had an almost completely transparent user interface, revealing to a user all of its assumptions about them as well as the whole of its further decision process of choosing the right book to recommend, nowadays hardly any recommender system is made to be transparent (Rich, 1979). Such a black box approach in regard to user interface, which virtually all modern recommendation systems have evolved to adopt, is mainly dictated by the sheer complexity of the decision-making process of today’s systems, which very rarely lends itself to any kind of user interpretation. However, the transparency of user interface was also reduced in order to conceal any possible traces of illegal user data manipulations as well as any unethical assumptions made about a given user, due to which facts users have actually been enjoying the recommendations slightly less ever since (Sinha & Swearingen, 2002).

## **2.2 Recommender system approaches: core algorithms and respective applications**

In a classical fashion, the majority of recommender systems can be separated into the three categories based on their approach towards matching of the new user’s input data to an existing database in order to produce relevant recommendations:

a) Content-based filtering approach that is based on the varying degrees of similarity between different items;

b) Collaborative filtering approach that is frequently based on the so-called “neighbourhoods” of similar users;

c) Hybrid filtering approach which aspires to combine in the most balanced and efficient way both of the above approaches (Lu et al., 2015).

To start with, the way a content-based recommender system functions is: firstly, by calculating the similarity between all of the items in a database, for instance, movies, news articles or travel destinations; and then, as a given user updates the system with their item ratings, present that user with a recommendation of items, which were previously measured to be similar to the ones that user has rated the highest. Thus, with content-based filtering the ultimate user recommendation depends exclusively on that single user’s profile as well as obviously on the database of item profiles. The origins of the content-based filtering recommenders can be traced back to the first inklings of research in the areas of information retrieval and information filtering techniques, which became to be most extensively applied in the field of Natural Language Processing. For this reason the content-based filtering approach is commonly employed by the recommendation algorithms that are focused on measuring the similarity between chunks of textual data from item contents. Thus, item profiles are often described in terms of keywords, and the similarity between them is calculated via the special measure of *term frequency and inverse document frequency* metric, commonly abbreviated as *TF-IDF*, which can be produced in the following way:

where: – the total number of the keyword’s appearances in the document ;

– the most frequently encountered keyword in the document;

– the total number of documents;

– the total number of the documents within which the keyword is present (Lops

et al., 2011).

In simple terms, the goal behind this basic metric is to identify keywords in a document that are the most relevant for the document’s topic, by means of recording the keywords that occur most frequently within that specific document (term frequency) and that are, at the same time, among the least frequently occurring in the general corpus of all other documents (inverse document frequency). This way, the weight of each document is computed as the product of the TF and IDF values (Adomavicius & Tuzhilin, 2005). As a result, based on the weights of documents, the degree of closeness between every single document in the overall corpus can be easily identified either with the help of various distance metrics, for instance the Euclidean distance or the Manhattan distance, or similarity measures, by far the most common of which is the cosine vector similarity (Lops et al., 2011).

However, the overwhelming majority of modern recommender systems do not exclusively rely on such obsolete heuristic methods as basic similarity measures, instead they employ machine learning algorithms that create predictive models by training them on the input data of users. Thus, content-based recommenders are generally divided into the two major subgroups of classifier-based and nearest neighbour-based models (Portugal et al., 2018). The former ones probabilistically assess all the items that have not been rated by a user according to the binary categories of either “recommend” or “do not recommend” based on the user profiles of item ratings, for example, a wide variety of the decision tree models and Bayesian network methods; while the latter models separate all of the items into clusters based on how similar they are, in order to recommend those items that share the same cluster with the ones a given user has rated the highest, for instance, applying the k nearest neighbour classifier technique to the cluster centres produced in the result of the k-means clustering method (Lops et al., 2011).

One of the main problems inherent in the content-based recommendation systems is the extreme homogeneity of recommended items as well as the narrowness of the overall recommendation range. Users are continuously presented with items too similar to those they liked while being completely unable to discover brand new items, which have slightly different profiles of characteristics but may still be of much interest to them. Moreover, the goal of making highly accurate recommendations of items that are too much alike appears to be a much worse method, then trying to diversify the range of recommended items, for instance, suggesting a myriad of similar news articles that, in fact, cover exactly the same event (Adomavicius & Tuzhilin, 2005).

Speaking of the collaborative filtering recommender systems, the key task of such an approach is to predict the relevance (or the rating scores) of items for a particular user based on the similar profiles of other users present in the database, that way quite literally making all the users collaborate in producing a good recommendation for any single user. The respective family of recommendation methods are commonly further separated into the following two major classes: the memory-based and the model-based ones. The simple core difference between these two types of collaborative algorithms is that the memory-based ones make calculations over and store in computer memory the entire user database in order to produce statistical predictions of item ratings for the new users, while the model-based ones employ various machine learning models in order to, firstly, train them on an available compiled dataset and, then, use them to make predictions for new users (Breese et al., 1998). Each of the two types of collaborative filtering algorithms is associated with its major respective group of specific methods which are, namely, the user neighbourhood methods and the latent factor models respectively. Firstly, the former are focused on measuring similarity between either users or items. However, even though the neighbourhood methods might exploit the attributes of items, unlike the content-based algorithms they make a recommendation based not solely on item similarity, but rather on the similarity between users, that way being able to identify like-minded individuals and recommend to each of them those items which they do not share both, but only one of them has rated. Latent factor models, in their own turn, infer the presence of implicit descriptive factors that can be revealed from certain arbitrary, yet computer discernible patterns in the user-item data (Koren, Bell & Volinsky, 2009).

Although the collaborative filtering approach can be applied by means of many of the same most basic classification and regression methods that were mentioned previously for the content-based approach, such as the various distance, correlation and similarity measures, the probabilistic classifiers (decision tree, random forest and Bayesian models), regression models (linear, polynomial or logistic) and clustering methods, the present paper is going to focus on one of the most promising types of machine learning models for collaborative filtering called the latent factor models, for the main reason that these types of algorithms have proven to deal with huge and sparse datasets more efficiently than the majority of other standalone techniques (Koren et al., 2009).

Latent factor models have been most extensively studied in the research domains of: the Natural Language Processing techniques, with the specific examples including, but not limited to the probabilistic Latent Semantic Analysis (pLSA) being used for information retrieval (Hofmann, 2004) and the Latent Dirichlet Allocation (LDA) employed in text classification (Blei et al., 2003); as well as in the context of the major recommender system algorithms and dimensionality reduction techniques, some examples of which are the matrix factorisation-like methods of singular value decomposition (SVD) being applied for the dimensionality reduction of a user-item ratings matrix (Billsus & Pazzani, 1998).

However, it can be easily argued that one of the most widely known of the latent factor models for collaborative filtering are the matrix factorisation models. These methods have dramatically gained in popularity as a result of the Netflix Prize open competition in the year of 2009, when an SVD algorithm, which later became widely known after its creator as the FunkSVD, was specifically tuned and applied by Simon Funk to produce very surprising improvements in the accuracy of movie recommendations (Piatetsky-Shapiro, 2007). The basic methodology behind the matrix factorisation techniques is based on, firstly, representing both items and users as vectors of latent factors and, then, recommending to users only those items that produce the highest dot product between those factor vectors. Due to the fact that those matrices of user-item interactions often suffered from the data sparsity problem and, as a result, most machine learning models tended to drastically overfit, for the matrix factorisation a regularisation parameter could be naturally implemented into the process of minimising the following squared prediction error, in order to penalise for the extremely high factor vector norms for items and users:

where: – the actual rating of the user *u* for the item *i*;

– the unknown factor vector for each item *i* that is being predicted;

– the unknown factor vector for each user *u* that is being predicted;

– the dot product, i.e. the predicted rating of the user *u* for the item *i*;

– denotes the norm;

– represents the set of user-item pairs where ratings are known;

– the regularisation term (Koren et al., 2009).

There exists a wide range of variations of the matrix factorisation methods that continue to still be experimented with in the domain of recommender system research. The selected range of the three of the most accurate and frequently cited singular matrix factorisation methods, which operate on the basis of noncategorical variables and which without exceptions have all been almost exclusively tested on the most widespread and abundant content of film databases, has been assembled to be the following: the SVD, the SVD++ as well as the NMF algorithms, with every one of them being optimized by means of the regularized SGD learning method (Cacheda el al., 2011; Koren et al., 2009).

First and foremost, the well-established matrix factorisation-like model called the Singular Value Decomposition (SVD), which has proved to be the most accessible out of the range of factorisation methods and at the same time highly efficient with sparse data among other collaborative filtering methods (Cacheda et al., 2011; Gorell, 2016). The basic version of SVD does not deal too well with the data sparsity problem of a user-item matrix, and, when still applied to the very meagre amount of known values, it understandably tends to overfit and fails to make sound predictions for the new users. Initially, most of the research efforts were directed to adapting various data imputation techniques as a way to increase the density of interaction matrices. However, this approach most of the time proves to be extremely costly as well as too complex and case specific (Ranjbar et al., 2015). For this reason a more general and straightforward technique was proposed to yield the double benefits of, firstly, the implementation of the abovementioned regularisation term and, secondly, of including additional bias parameters for each user and each item, which are actually meant to account for the larger part of variations in ratings that are caused by the independent effects associated with particular users and items (Koren et al., 2009; Paterek, 2007). Thus, the ratings prediction formula is modified to include the biases and set to be calculated according to the following four components:

;

where: – the global average rating across all known data points;

– the user bias reflecting the general tendencies of how the user *u* rates items;

– the item bias reflecting the general tendencies of how the item *i* is rated by users;

– the dot product of factor vectors accounting for the user-item interaction (Koren

et al., 2009).

As a side note, in the cases of considering the addition of a new user or a new item, all of the associated biases and factors are, at first, assumed to be zero. As a result, the problem of minimising the squared error turns to be the following:

where: – the actual rating of the user *u* for the item *i*;

– the predicted rating of the user *u* for the item *i*;

– the user bias reflecting the general tendencies of how the user *u* rates items;

– the item bias reflecting the general tendencies of how the item *i* is rated by users;

– the unknown factor vector for each item *i* that is being predicted;

– the unknown factor vector for each user *u* that is being predicted;

– denotes the norm;

– represents the set of actual ratings in the training dataset;

– the regularisation term (Koren et al., 2009).

Furthermore, the unique contribution of the already mentioned Simon Funk’s SVD (FunkSVD) algorithm was that it employed the regularisation term as well as a learning algorithm of Stochastic Gradient Descent (SGD), which became widely known for the accessibility of its implementation on the large-scale datasets, especially the ones that are heavily afflicted by the data sparsity problem, as well as for its fairly short running time of the core computations and the subsequent evaluation process, which is due to the fact that SGD stores in memory only the most recently computed rating prediction errors (Lin, 2007). Without going into too much application details, the SGD optimisation algorithm is employed to circle through all of the available data points and make predictions of user ratings, each time readjusting the user and item factor vectors based on the rating predictions’ error, until eventually arriving at the optimal pairs of user and item biases as well as of user and item factor vectors, as can be specified in the following mathematical fashion:

;

;

;

;

;

where: – the actual rating of the user *u* for the item *i*;

– the predicted rating of the user *u* for the item *i*;

– the unknown factor vector for each item *i* that is being predicted;

– the unknown factor vector for each user *u* that is being predicted;

– the rating prediction error of the user *u* for the item *i*;

– the regularisation term;

– the learning rate (Koren et al., 2009).

Alternatively, soon after the Netflix Prize open competition, where the FunkSVD attracted much attention to the SGD learning algorithm, another optimisation algorithm was popularised by the name of Alternating Least Squares (ALS), which, in essence, is performed by fixing all the factor vectors for users, in order to compute the ones for items, and then switching around to predict those for users through minimising the sum of the squared residuals (Rendle, 2012). The two common use cases for this approach have been observed to be the following: firstly, it can be very efficiently used in the scenario of when the recommender’s system is computationally capable enough to sustain the parallelisation of its predictive algorithm’s execution; and also, especially in the case of the recommender algorithm being trained on the more abundant implicitly collected data of users, as it would be impractical to apply ALS on the too sparse explicit data (Hu et al., 2008).

Second of all, the enhanced version of the FunkSVD algorithm called the SVD++ was also proposed in order to take into account the implicit ratings of users as well, which can be defined as the fact that a given user has rated a certain item, regardless of what that actual rating value was. Due to the incorporation of implicit ratings, the SVD++ method, when applied on the regularised squared error with the SGD optimisation, often shows an improvement in the accuracy of rating predictions over the FunkSVD algorithm, albeit a small one (Cacheda et al., 2011). As for the calculation of the predicted rating, the new set of factors reflecting the implicit ratings are incorporated as follows:

;

where: – the global average rating across all known data points;

– the user bias reflecting the general tendencies of how the user *u* rates items;

– the item bias reflecting the general tendencies of how the item *i* is rated by users;

– the unknown factor vector for each item *i* that is being predicted;

– the unknown factor vector for each user *u* that is being predicted;

– the total number of implicit ratings of user *u*;

– the new set of factor terms that captures implicit ratings (Koren, 2008).

Last, but not least of the three selected matrix factorisation algorithms, the Non-Negative Matrix Factorisation (NMF) algorithm simply introduces a non-negativity constraint to the basic SVD algorithm for both the user and item parameters, which are being calibrated as the training model continues to learn, for the main reason of ensure that the predictions of a data feature values are a more appropriate representation of a user’s interests. In order to actualise this constraint the algorithm basically rescales the learning rate to exclude all of the negative components from predicted factor vectors of users and items, that way leaving only the non-negative ones to compute the next iteration of a predicted rating and of its respective error (Luo et al., 2014). Thus, the stages of the SGD learning algorithm are slightly modified in the following way to calculate only the non-negative user and item factors of and respectively, requiring the starting factor values to be positive as well:

;

;

where: – the factor *f* of the vector which is predicted for each user *u*;

– the factor *f* of the vector which is predicted for each item *i*;

– the new regularisation parameter for user *u*;

– the new regularisation parameter for item *i* (Luo et al., 2014).

Hybrid recommendation systems, as the name suggests, combine the two approaches described before to minimise the drawbacks of each one taken individually. Adomavicius and Tuzhilin (2005) identify four types of hybrid systems. The first type is creating one content-based and one collaborative filtering recommender and combining their results either via voting or linear combination of their outputs. The second type combines collaborative filtering with some content-based system features. Some of the examples include assigning a content-based profile to each user and then finding similar user profiles instead of similar sets of rated items. The third method incorporates some content-based model characteristics into collaborative filtering. One of the approaches described is similar to matrix factorisation, but instead of reducing dimensionality of a simple user-item rating matrix it utilises user content profiles as described in the second approach. Finally, there are systems that combine both content-based and collaborative filtering approach in equal proportion (Adomavicius & Tuzhilin, 2005).

As Portugal et al. (2018) showcase in their systematic review of the various recommender system methodologies, in the course of the past ten years the collaborative filtering (66 papers) algorithms have gained more attention from academics than the content-based (45 papers) and hybrid (18 papers) ones. Moreover, neighbourhood and model-based collaborative filtering approaches have each of them generated vastly more attention as compared to the years before the 2012; while the hybrid recommender methodology remained to be the least researched out of the three classical approaches, which is, for the large part, due to the fact that it does not have any singular filtering methods directly associated with it, but rather requires a more consuming effort of assembling a whole recommender system out of any combination of the different types of filtering methods.

Burke (2002) was the first one to propose a solid classification of such ways of combining the contributions made by each of the two recommender approaches, otherwise known as hybridisation methods. The first and consequently the most popular one of these methods is the weighted one, which weighs the employed recommender algorithms according to their predictive accuracy, by means of assigning them with a set of votes to denote the extent of their contribution to the ultimate user recommendation. Furthermore, the other two methods that usually act as alternatives to the weighted one and that, unlike the weighted method, can be appropriately employed not only when combining techniques of the same relative value are called switching and mixed hybridisation methods; with the first one consisting in switching between recommender techniques depending on the present-moment system tasks and the latter one being applied whenever a side-by-side recommendation output from both techniques is more favourable. Furthermore, feature combination hybrid is a method of implementing both collaborative and content-based features for the purposes of producing a single recommendation. Quite unlike feature combination, feature augmentation does not converge the different techniques, but sequences them in a way that the recommendation output of one technique acts as feature input to the other. Similarly, the cascade method unravels in a staged process, however, the one in which recommendation output from the first technique is used only for further refinement by the next one. Lastly, an even more unique case of creating a confluence of the two different filtering approaches has been proposed as the meta-level method, which is similar to the feature augmentation, however the next technique in a sequence takes as its input not the previous one’s feature, but instead incorporates the previous recommender technique as a whole; for instance, several content-based models are employed to predict restaurant preferences for every separate user, and then these basic models, which, in essence, are just vectors of estimates, are fed into a collaborative filtering model to make predictions by comparing across users (Burke, 2002).

Due to the reason that the TripAdvisor platform incorporates features of a social network to enhance the quality of its travel recommendations and to promote independent communities of active users, it is necessary to point out the specifics of recommender systems that operate on the basis of social trust networks. First of all, the term social trust can be defined as a connection between a pair of users based on either explicit feedback (subscription, voting, commenting) or implicit feedback (frequency of interactions: page visits, message exchanges) (Yang et al., 2014). In parallel to the two types of feedback, there is a respective distinction between two types of alternative data sources that are assumed to enhance recommendation quality: rich side information on items and users and interaction-associated information. The former is mainly concerned with recording social trust relationships between users by exploring a subscription-based social network or by analysing various user-contributed data, like numerical ratings and textual reviews, photo and video material as well as social tags and geotags; while the latter most commonly covers information on time, location and user mood status recorded at particular instances of user-item interactions (Shi et al., 2014).

Moreover, there are again two general classes of social recommendation approaches: matrix factorisation based one, which combines user-user social trust data with user-item feedback history, and nearest neighbour based one, which first traverses the network of users’ direct and indirect friends to provide an additional advantage of social neighbourhood. When it comes to specific models that are commonly trained on such diverse and abundant data, previous research has conveniently identified what state-of-the-art algorithms had been used to implement side information in memory-based (cosine vector similarity, k-nearest neighbours), model-based (Bayesian network model, matrix factorisation model) and graph-based (random walk) collaborative filtering approaches; as well as to consider the interaction-based information by means of tensor factorisation, factorisation machines and graph-based approaches (Shi et al., 2014). It is not surprising that when it comes to efficiently storing in memory and manipulating huge arrays of data the matrix factorisation algorithms prove to be superior and, in particular, manage to excel at both item rating prediction and item list recommendation tasks (Yang et al. 2014).

Not least important is to point out the primary challenges that are associated with the current social recommender system tasks: firstly, trust- and distrust-based social recommendation of potential friends, products and other content; secondly, group recommendation for multiple people looking to choose a single activity, destination, etc; and thirdly, long tail recommendation, which refers to recommending items with low popularity – crucial for an effective recommender system (Shi et al., 2014).

Finally, Pantano, Priporas and Stylos (2016) emphasise that, in the process of making the choice of a specific recommendation system algorithm for the computing of item rating predictions, the researchers should take into account the specifics of the domain’s informational context (what the available data sources are; how the information on users and items is organised there), the type of data that was made available in this particular domain (if numerical, strings, mixed, etc.), as well as the maximum bearable mark of the computational costs (incorporating the speed of a programme execution).

## **2.3 Overview of popular travel recommender system approaches**

The central aim of the majority of tourism-related recommendation systems is to provide users with suggestions of relevant travel destinations and tourist attractions that are otherwise commonly known as the Points of Interest (PoIs). Although there exist other more sophisticated systems that are offering users travel routes recommendations and even personalised trip planning services, in line with the objectives of the present research paper, only the algorithms that specialise in recommending PoIs are going to be considered (Gavalas et al., 2014).

The broad variety of popular recommender system frameworks, which are employed for the application in the travel domain, can be most conveniently represented as the following categories according to both the type of approach that they employ and the type of data that they are based on (Roopesh & Tulasi, 2018):

a) Context aware systems that rely on constantly gathering contextual information from a person’s device, web browser or social media that way enabling a live update of the user’s data on his current location, time and day of the week, current season and weather conditions, etc. The combination of such diverse data of a person’s context provide the basis for presenting the user with, for instance: the recommendations for a number of tourist attractions, based on their working hours, shortest user travel paths and sentiment scores from social media (Meehan et al., 2013); or, based on the current weather data and the person’s travel history in the form of geographically tagged photos, the recommendations of a range of similar-looking attractions in a different city that were shared by other users on the photo sharing web sites (Xu, 2014). The core difficulties with context-aware systems are the high intensity of ongoing per user computations as well as the high complexity of the server-side software architecture enabling the continuous, repeated parsing of massive volumes of online data;

b) Social network-based recommenders, which function on the basis of a users’ social profile information from Twitter, Facebook, Instagram and Flickr among many others, are exploiting the existing social trust relationships between online users in order to make appropriate recommendations. For instance, suggesting to a user a list of attractions based on his as well as on their friends’ check-in data shared on their Facebook pages (Kesorn et al., 2017); or applying the data of users’ social trust relations in order to predict tourism-related customer purchase behaviour, in regard to the specific tour packages and hotel bookings (Esmaeili et al., 2020). The principal challenge with this approach is the high level of data sparsity, which can be slightly brought down by allowing the scope of the recommendation base to be extended to not as relevant travel preferences of the user’s friends of friends;

c) Hybrid filtering approach is also very frequently employed for developing recommender systems in the travel domain, as the collaborative and content-based filtering techniques prove to be inadequate on their own and underperform primarily due to the high levels of data sparsity on the travel-related platforms. For instance, one research has designed a hybrid system by combining multiple similarity measures, namely, the Tanimoto coefficient and the Euclidian distance measure for the collaborative and content-based constituent filtering techniques respectively (Kbaier et al., 2017). In another study a clustering algorithm as well as the associative classification method were sequenced together in order to group users based on demographic data and make predictions of POI ratings for these groups (Lucas et al., 2013);

d) Demographic filtering approach makes the use of the demographic user characteristics, such as age, gender, travel region, general travel style, purpose of the trip, travel companions etc., and is extensively applied as a means of overcoming the cold start problem, which is characterised by the absence of any previously rated items for a new user (Wang et al., 2012). Such application is justified by the fact that different demographic groups enjoy different aspects of their travel experiences, thus, naturally forming unique travel styles per certain group in accordance with these preferences (Fuchs & Zanker, 2012). Due to the fact that this approach is not entirely self-sufficient, in practical application research it is most often developed as a supporting part of either a hybrid (Kbaier et al., 2017; Renjith & Anjali, 2014) or a context-aware travel recommender system (Cheng et al., 2011).

When it comes to the research that has been conducted specifically on the data from the TripAdvisor platform, there has already been a diverse range of research papers published, studying and mining the useful patterns in the users’ trip behaviour data as well as applying that travel-related data for the purposes of developing a travel recommender system.

To begin with, a number of published papers has been dedicated to exploring the structured customer feedback in the form the multi-criteria ratings of hotel reviews, which cover all of the essential aspects of the users’ hotel experiences, from the satisfaction with the specific experiences of the first front desk interaction at check-in as well as of the room quality and business services of Wi-Fi access to the overall hotel qualities of the location, cleanliness and value for money (Fuchs & Zanker, 2012). The recognised recommendation potential of these multi-criteria hotel ratings from TripAdvisor has proven to be of deep interest to multiple research teams. Perhaps the groundworks for the exploration of the multi-criteria tourist ratings were laid down in the paper by Fuchs and Zanker (2012), who applied a number of the multiple linear regression models to examine the influence of different criterion ratings on the users’ overall hotel satisfaction rating across the 4 tourist market segments, which were grouped according to the personal user information explicitly specified as part of the TripAdvisor hotel reviews. As a side note however, only recently several researchers have teamed to attempt exploiting the multi-criteria user ratings, specifically of the Canary Islands hotels and of US spa hotels, for the purposes of themselves creating the SOM clusters of users with different overall satisfaction levels as well as for ultimately applying the CART models for the prediction of the users’ travel behaviour concerning the choice of hotels (Ahani et al., 2019a; Ahani et al., 2019b). In continuation, two years later an extended research team of Jannach, Zanker and Fuchs (2014) managed to produce high accuracy recommendations, by means of employing their previous findings of segment-specific hotel satisfaction factors as well as in the result of successfully experimenting with the single-rating matrix factorisation-based algorithm called SVD. Furthermore, Zheng (2017) has proposed a novel multi-criteria recommendation technique of “Criteria Chains”, which assumes a dependent relationship between multi-criteria ratings and, with the help of the context-aware biased matrix factorisation algorithm, evaluates the user rating for each criterion in the context of the previously predicted ones.

Another notable, however much less common approach towards TripAdvisor data exploration has to do with the analysis and processing of textual user reviews. On the one hand, the extensively researched area of sentiment analysis, also known as opinion mining, has not been seeing too much of general algorithmic improvement in the recent years. For instance, the novel combination of the Fuzzy Domain Ontology with the commonly employed method of the Support Vector Machine has been proposed for the goals of text classification in the paper by Ali, Kwak and Kim (2016), however it has not firmly succeeded in improving the performance of a simple Support Vector Machine model and resulted in a 10 per cent drop in the recall measure, despite a similar degree of improvement in the precision and accuracy metrics. On the other hand, the research has been shifting focus towards the other Natural Language Processing techniques of semantic analysis, such as the association rule-based and the topic model-based ones, the latter of which have recently gained much popularity in the tourism domain and started to be extensively featured for the purposes of data mining and pre-processing of textual reviews. For instance, by far the most common of such topic segmentation methods has been the Latent Dirichlet Allocation model (LDA), which uses an unsupervised learning algorithm in order to identify and label the most ubiquitous underlying topics present in the huge volumes of unstructured textual data (Blei et al., 2003). LDA is highly efficient as it can be adapted to be applied in the cases of extremely sparse disarrayed datasets of textual reviews, which is perfect for the use on TripAdvisor data. In the specific research examples this model has been applied with the goal of effectively extracting from the TripAdvisor hotel reviews the most frequent word dimensions (also known as topics) that signify the degree of consumer satisfaction, in order to identify the most important hotel features according to customers (Guo et al., 2017).

Other instances of already explored and collectively unrelated approaches towards the study of the TripAdvisor data within the context of the recommender system research include the following:

a) Applying the probabilistic classifier techniques, such as the Naive Bayes and Support Vector Machines, to the TripAdvisor attraction ratings in order to make user rating predictions with the additional help of the demographic data specified on user profile pages. Ultimately, the performance results did not show significant improvements after the application of demographic filtering, which was mostly due to the inherently high sparsity of the travellers’ profile data (Wang et al., 2012);

b) Creating a novel and highly effective hybridisation method, which was directly inspired by the weighted and mixed ones, in order to combine the TripAdvisor item recommendations from some of the top performing location-based context-aware recommender systems (Logesh et al., 2019);

c) Applying the deep learning techniques of constructing artificial neural networks to the TripAdvisor photo databases in order to recommend POIs, based on how likely a given user is to be the author of a given POI’s photos that were taken and shared by other users (Díez et al., 2020).

## **2.4 Major recommender system issues and their common solutions**

Although recommender systems can often face a whole range of different issues, such as: the problem of providing users with accurate, but not diverse enough recommendations or vice versa, the problems of scalability and latency, the problems of shilling attacks and user privacy; most of these problems are most likely to be either encountered and dealt with post factum, that is, well after the launch and first test trials of a specific system, or not at all (Khusro et al., 2016). Interestingly enough, there are only a handful of major problems that every single effective recommender system has to eventually overcome, namely, the two most prominent of them are: the cold start problem as well as the general issue of high data sparsity.

Speaking of the former one first, the cold start problem occurs upon the system’s online launch and in all types of recommender system algorithms, especially in the collaborative filtering approach, and denotes the situation of the inability of the system to make relevant item predictions for a new user for the reason of only having received the user’s scant input preferences and no other history of ratings. The instances of when the problem of a cold start is very likely to occur are often divided into the three following categories: (a) recommending items for new users with little or no preference history record, (b) continuing to produce relevant recommendations for the existing users, while at the same time updating the database with new items, and lastly, (c) recommending newly added items to the newly connected users (Lika et al., 2014).

This research area has already seen a significant number of various experimental methods that have been suggested for the goal of avoiding, or at least softening, the impact from the cold start problem. Understandably, most of the respective research was conducted with the focus of solving the more pressing new-user instance of the cold start problem, rather than the new-item one. The specific researched solutions often fall into one of the following groups:

a) Introducing the preparatory stage of the initialisation of new users in the form of a brief interview process that is controlled by something called a bootstrapping algorithm, which is developed to adapt to the user’s choices in order to elicit the most informative responses; for instance, offering users to rate a short list of items that are representative of different groups of user preferences, that way quickly and accurately identifying the preference type of each new user (Golbandi et al., 2011; Zhou et al., 2011);

b) Establishing a range of user categories and simply making the new users decide on their own with which one of these categories they choose to associate themselves. Although this technique may seem to be the easiest in placing a new user profile in the context of the existing database, without having to make the user answer too many questions or to rate any items, this approach often cannot produce effective results and should only be applied in specific domains where it is not expected of users to be associating themselves with more than a single category at a time (Al Mamunur et al., 2002);

c) Employing the context-aware recommender systems that are most commonly based on the social trust networks data of individual users to exploit the established relationships between users and/or items; for instance, collecting data from the platforms that allow the annotation of items by the assignment of social tags for the purposes of quicker topic identification and more effective opinion sharing (Zhang et al., 2010);

d) Deriving from data patterns a range of association rules for user preferences in order to further extrapolate the new user profiles from the starting input data; for instance, guessing the new users’ additional topics of interest by traversing the topic associations of other user profiles (Shaw et al., 2010);

e) Developing standalone predictive algorithms, which most commonly exploit supplementary information of item contents and/or user demographical characteristics, as an addition to a recommender system in order to generate more accurate recommendations for new users; for instance, by way of constructing a regression model for pairs of user and item features for each one of the new users in order to predict their item ratings (Park & Chu, 2009).

However, the most prominent recommender methodology conclusion that is almost unanimously shared by these research papers is the clear superiority of the hybrid system approach in dealing with the cold start (Lika et al., 2014). In this view, hybridisation is often used to adopt a particular content analysis methodology to item descriptions, in order to recommend relevant items based on little initial input data, when the new user’s ratings have not yet been accumulated, that way balancing the drawbacks of collaborative filtering with the advantages of the content-based filtering and vice versa.

As for the other issue of high data sparsity, the collaborative filtering algorithms are once again the most exposed to having this problem. In collaborative recommender systems user profiles are most often represented as vectors of users’ item ratings, which form a user-item (or consumer-product) interaction matrix. Due to the simple fact that an overbearing majority of users at any point in time has rated only a limited number of items, the user-item matrix is always subject to having zero (or missing) values for those items that the users have not yet rated. This natural occurrence, commonly referred to as data sparsity, and was observed to pose an acute problem of drastically reducing the accuracy of the system’s recommendations, the reason for that being that sparsity of data not only weakens the correlation between any given pair of potentially similar users, but also makes a strong correlation between users an essentially unreliable measure. From this perspective, the problem of a cold start can be viewed as a specific case of the sparsity problem, when virtually all elements of a new row or a column in the user-item interaction matrix contain missing values. As a result, specifically in the cases of the technique of collaborative filtering being used without any assistance from other methods, extremely high levels of data sparsity tend to devoid the collaborative approach of any positive impact on the quality of recommendations (Chen et al., 2011).

The research field has seen a number of very diverse solutions to the data sparsity problem, most of which, unsurprisingly, bear a positive impact on and a strong resemblance to the cold start solutions. In spite of the wide range of researched solutions, a few groups of the most commonly used sparsity-alleviating techniques have still been identified as the following:

a) Reducing the dimensionality of the user-item interaction data in order to generate a much denser, more concise interaction matrix of only the top most prolific users with the largest numbers of item ratings. There are many different techniques that can be applied to achieve the reduction of data dimensions: starting from the simple statistical methods, such as creating clusters of either items or users to base the predictions on, and increasing the complexity further to discover the more eccentric techniques, such as the probabilistic topic evaluation model of the Latent Dirichlet allocation (LDA) and the information retrieval technique of the Latent semantic indexing (LSI) in the domain of natural language processing (Grčar et al., 2006; Blei et al., 2003). However, the most widely recognised and, arguably, the most common dimension reduction technique is that of matrix factorisation, which has proven to be extremely efficient in handling large databases without any significant data losses. On the one hand, the possible downside of such a simplification approach can be a noticeable loss in the recommendation accuracy, on the other hand, depending on a chosen technique and a way of its integration, the result may happen to be of increased recommender system’s performance (Bobadilla, Ortega, Hernando & Gutiérrez, 2013).

b) Representing the user-item interaction matrix as a graph of global similarity between users, where nodes refer to users and edges denote the degree of similarity between a given pair of users, in order to predict the potential interest in an item (or even the item rating) of a particular user, based on the length of a path directed from that user to the user who has already rated the item in question. Most commonly such approach requires a preliminary stage of creating for each user a bipartite graph, which establishes connections between a given user and the items that he or she has rated. The issues of this approach are often the low interpretability of user similarity measures as well as the off-the-scale computational intensity given a large enough dataset (Chen et al., 2011).

c) Transitioning onto an item-based collaborative filtering algorithm, in order to move away from seeking for a given user’s similar neighbours and instead focus on the item-item similarity according to a given user’s ratings. Such alternative approach to the user-based collaborative filtering enables the algorithm to rely on the preferences of a specific user who is making a recommendation query, instead of relying on the sparse item ratings of other users. One of the most resource efficient and most common among such methods are the ones of cosine-based and correlation-based similarity, similarity measures of which are then often used as weights for determining a weighted sum of k nearest items (Sarwar et al., 2001).

## **2.5 Performance assessment metrics: definitions and applications**

Ever since the dawn of the recommender system research, the assessment of the recommendation quality as well as of the system’s overall performance have quickly become a vital part of the field. Thus, over the years the recommender systems’ performance evaluation metrics have been developed into the following four broad classes of: (a) the accuracy metrics for the ratings prediction, such as the Mean Absolute Error (MAE), the Root of Mean Square Error (RMSE) and the Coverage measure; (b) the relevance metrics for the set of recommendations, such as the Precision, Recall and Receiver Operating Characteristic (ROC); (c) the recommendation ranking metrics, such as the Half Life Utility (HLU) and the Discounted Cumulative Gain (DCG); and last, but not least, (d) the recommendation variety metrics: such as the diversity and the novelty of the recommended items (Hernández del Olmo & Gaudioso, 2008).

Traditionally, as a means of later assessing any of the quality aspects of a recommender system’s performance in the context of a research study, a user dataset is divided in advance into the following two sets: a training set, which is meant for the purpose of training the machine learning model, and a test set, which is meant for the purpose of evaluating the performance of the already trained model, with the dataset’s division conducted along the line of the most common optimal relation of 80 to 20 per cent respectively (Cacheda et al., 2011).

Accuracy prediction metrics are certainly the most frequently featured in the recommendation system research literature. In accordance with the presently obsolete basic supposition that a successful recommender system is the one that most accurately predicts user preferences, many of the previously published research papers had the intention of constructing algorithms that would provide more and more accurate user recommendations and of evaluating such models accordingly (Shani & Gunawardana, 2011).

The core performance metrics, which are always used to measure the accuracy of predicted ratings in the context of a user study, include the two closely related metrics of: the *Root Mean Squared Error* (or *RMSE*, for short) and its common alternative of the *Mean Average Error* (or *MAE*, for short):

where: R – the user-item interaction matrix;

||R|| – denotes the matrix size (i.e. the number of ratings);

– the user’s actual item ratings;

– the user’s predicted item ratings (Yang et al., 2014).

Both of these metrics solely depend on the magnitude of prediction errors. However, the key difference when comparing the two is that, unlike the MAE metric, the RMSE one tends to be more heavily penalising the models with only a few instances of large prediction errors and instead preferring the models with a uniform level of prediction errors. In sight of this fact, it is also important to add that in the case when the test set has an unbalanced distribution of item ratings, these metrics are very likely to get skewed by the ratings prediction errors of the most frequently rated items. For this reason it is advisable to calculate RMSE and MAE for every item separately and then compute an *Average RMSE* and *Average MAE* over all items. Similarly, if, in the case of an unbalanced user distribution, the goal is to determine what level of recommendation accuracy a randomly chosen user is likely to receive, the Average RMSE and Average MAE metrics should be calculated over all users (Hernández del Olmo & Gaudioso, 2008).

The most common general task of a given recommender system is to provide any user with a top-k list of a certain fixed *k* number ofitem recommendations that were predicted to be the most reflective of the users’ tastes. Thus, in order to measure the relevance of such a list of recommended items for every single user, other special metrics are applied instead, for instance, the most widespread ones are those of *Recall* and *Precision*. Recall (also known as *sensitivity*, *hit rate* or the *true positive rate*) is a metric that represents the fraction of the total amount of relevant items that were actually recommended to the user *u* andis obtained by dividing the number of relevant elements *N (k, u)* present in the top-k list by the total number of relevant items *N(u)* (relevance is established, for example, in the case if a rating exceeds a certain cut-off value) (Yang et al., 2014):

where: True Positives – the number of relevant items that were recommended;

False Negatives – the number of irrelevant items that were not recommended

(Sokolova et al., 2006).

Precision is a metric that represents the fraction of the total amount of recommended items that were actually relevant for the user *u* and is calculated by the same metric of the number of relevant elements *N (k, u)* from the top-k list in its numerator and the total number of recommendations *k* of the top-k list as its denominator (Yang et al., 2014):

where: True Positives – the number of relevant items that were recommended;

False Positives – the number of relevant items that by mistake were not recommended

(Sokolova et al., 2006).

In simpler terms, both of these metrics evaluate a recommender algorithm’s ability to identify all of the user-relevant items that are present in the test dataset. While it is exactly true for the recall metric, the precision metric is also measuring the system’s capacity to draw relevant items in relation to the system’s errors of recommending useless items (Shani & Gunawardana, 2011).

For the reason of producing a more balanced evaluation of a model’s performance based on the abovementioned metrics of recall and precision, a separate set of indicators called *F measures* has been derived in order to simultaneously track the behaviour of both of these metrics. The most extensively applied one of the F measures has been the *F1 score*, which assigns equal weights to both the precision and recall and can be easily computed as a harmonic mean of these two metrics:

where: Precision – the metric showing the fraction of the total number of recommended

items that were actually relevant for the user;

Recall – the metric showing the fraction of the total amount of relevant items that were

actually recommended to the user (Sokolova et al., 2006).

Conveniently enough, the F1 measure also represents the value of the area under the precision-recall curve, which is helpful when trying to assess the quality of algorithm’s performance over a range of recommendation list lengths, in order to determine the most favourable one (Sokolova et al., 2006).

However, there exists another popular alternative of evaluating the accuracy of a recommendation list, namely, the Receiver Operating Characteristic analysis (or ROC, for short) and its respective curve, which measures the recall (known as the true positive rate) against the *false alarm ratio* (also known as the *fall-out* or the *false positive rate*):

where: False Positives – the number of relevant items that by error were not recommended;

True Negatives – the number of irrelevant items that were mistakenly recommended

(Fawcett, 2006).

The focus of the ROC curves is to reflect the proportion of unwanted items that have still been put on the list of user recommendations. Thus, the central goal of the ROC analysis and the ROC optimisation is to return every relevant item without returning any of the irrelevant ones (Fawcett, 2006).

The choice between the two aforementioned options of the recommendation list accuracy metrics should primarily be decided upon the domain of the recommender’s implementation as well as on the goals of the system’s application. The general shared agreement among researchers is that in the simple case of wishing to recommend to users as much relevant items as possible the precision-recall curve should be sufficient for such a task. However, especially if the system gets implemented in a business setting, for instance in the domain of e-commerce, and holds as its core aim the maximisation of the number of new purchases and the minimisation of the marketing costs of maintaining the recommender algorithm, then the ROC curves, which are known for their widespread application in the cost/benefit decision analysis, should certainly be chosen over the precision-recall ones for their crucial ability of tracking the system’s mistakes of recommending useless items (Shani & Gunawardana, 2011).

In the common instance of assessing the precision-recall or ROC curves for several test users when every user will be presented with a fixed number of *k* recommendations, the appropriate strategy for the recommendation set relevance evaluation is to calculate the precision and recall metrics at each number *k* of recommendation list lengths for each user, and then compute the average precision and recall at each number k of recommendation list lengths. The analogous approach can be taken in regard to the construction of an averaged ROC curve, every operating point of which will correspond to a different number of recommendations provided to users (Shani & Gunawardana, 2011).

As the length of the recommendation list is increased, every next recommended item tends to lose relevance to the user more quickly. For this reason, in cases when the *k* number of recommended items is quite large, it is advisable to introduce measures that reflect the quality of the recommendation list ranking. The two of the most frequently employed such ranking metrics are the *Half Life* *Utility* (HLU) and the *Normalised Discounted Cumulative Gain* (NDCG).

The Half Life metric assumes that a user’s interest towards items decays at an exponential rate as he moves from the topmost item down the list of recommendations, which is to say that there is an exactly 50 per cent chance reduction for every next item of a given user actually proceeding on to that following item:

where: – the true rating of user *u* for the item *i*;

d – the neutrality indicator, frequently chosen to be 0;

– the 1-based rank at which the item *i* appears;

– the rank of the item on the recommendation list, such that there is a 50 per cent

chance the user will review that item (Herlocker et al., 2004).

The Normalised Discounted Cumulative Gain, on the other hand, assumes that the relevance gain reduces logarithmically as a user moves down the list of recommendations. The main difficulty in computing NDCG, though, is the requirement to know the true user ratings for every single item on his recommendation list:

where: ,..., – the ranked list of recommended items;

– the true rating of user *u* for the item *p* that was ranked in position *i*;

– the maximum attainable gain value for user *u* which is obtained with the

optimal re-order of the k items in ,...,(Baltrunas et al., 2010).

Over the years the fundamental definition of a successful, useful recommendation has gained in complexity, shifting from the core focus on accuracy to concepts like diversity, novelty and even serendipity (the degree of surprise or unexpectedness) of recommendations (Ge et al., 2010). According to the commonly accepted concept definitions, the novelty aspect of a recommendation can either reflect the degree of how generally unpopular, yet still relevant a recommendation is for a user or, in calculable terms, the degree of difference between the recommended items. However, as items with low popularity scores are quite unlikely to be familiar to the potential users, most typically the novelty of a given item is calculated as the inverse of its popularity rating, which is often simply measured as the total amount of ratings that that particular item has received. The diversity aspect of a recommendation, in its turn, can be defined as the degree of differentiation of the recommended items by their belonging to a certain item type or by differing across other characteristics. As it is impossible to state with absolute certainty the exact range of a few topics the user is most interested in at any specific time, making sure that the recommendation list is comprised of items that collectively cover a wide spectrum of the user preferences, greatly increases the system’s chances of matching the user’s current needs. This can be achieved by optimising the diversity of the recommendation list, which can be measured in terms of either item features or item content features, such as for instance: the general item types, subtypes, topics, etc. (Kaminskas & Bridge, 2016). The single most common metric that is frequently used to assess the degree of recommendation diversity is called the Coverage measure and can be calculated as a simple fraction in the following way:

where: *k* – the number representing the extent of top-k lists of user recommendations;

– the total number of distinct items in top-k places of all user recommendation lists;

– the total number of items that are available for being potentially recommended.

Thus, the fact that Coverage might end up as just a small fraction often means that only the most popular items actually end up on the user recommendation lists, which coincidentally also causes over-the-top levels of accuracy, rendering the respective metrics completely unreliable. For this reason a solid recommendation quality can only be achieved at high levels of both accuracy and coverage. Lastly, the ubiquitous use of the Coverage diversity measure, alongside the common metrics of recommendation accuracy, indicates its utmost importance in achieving a balanced performance evaluation of any given recommender system (Lu et al., 2012).

# **3. Statement of the research question**

The two research questions, of the core and the supplementary focus, that set the direction for the efforts of the current thesis are stated the following way:

– What is the highest performing collaborative filtering algorithm among the matrix factorisation models that is able to produce the most accurate recommendations, based on the TripAdvisor data of user ratings for the London’s tourist attractions?

– Does the text recognition technique of the latent Dirichlet allocation reveal coherent categories of user preferences to be employed for the content-based filtering of new users, based on the TripAdvisor data of user reviews for the London’s tourist attractions?

First and foremost, in order to approach answering such a research question, the present research thesis sees it necessary to narrow down the focus and to only consider the specific case of the TripAdvisor data on any single city that is currently ranked as one of the most popular ones among tourists. Hence, the case has been made for the city of London, UK, which, according to the Mastercard’s annual report on the Global Destination Cities Index, has been forecasted to overtake Paris by the overnight international visitors in 2019 and once again rank in the 2nd place, inferior only to the Bangkok’s figures (Hamel & Robino, 2019). In addition, the current research has settled on developing a system, based on the single-rating user reviews of tourist attractions, that would simply recommend to the UK travellers the Places of Interest (otherwise known as PoIs), namely, the landmarks and experiences available in London. Such decision has been informed by the fact that, as this paper previously reviewed, there have been extensive research performed on many of the other types and thematic categories of the TripAdvisor data, for instance: the multi-criteria user ratings and textual reviews for the various hotels (Chang et al., 2017; Ahani et al., 2019b); the demographic information of users, explicitly stated in their TripAdvisor profiles (Wang et al., 2012); as well as the users’ photos of PoIs attached to their reviews (Diaz et al., 2020).

Furthermore, in view of the previously reviewed base of academic literature, covering some of the more recent and promising developments in the research field of recommender system methodologies, as well as taking into account the matrix configuration and the inherent sparseness of the travel-related data that has been planned to be effectively collected from TripAdvisor within the present computational and temporal constraints, the thesis is able to justify the key choice of measuring and comparing the performance quality of, specifically, the range of latent factor models for the purpose of answering the abovementioned research question. As it have already been discussed in the literature review, the factorisation models have been shown to be some of the more computationally accessible and efficient recommendation algorithms when applied to the huge and sparse datasets of user-item interaction matrices, which makes them absolutely perfect for the use in the proposed case of the TripAdvisor review data on the London PoIs.

When it comes to the choice of the performance evaluation metrics, first and foremost, from the two reviewed accuracy metrics for ratings predictions this research has settled on the reasonable choice of the MAE metric, as it has a more uniform error calculation approach, treating singular instances of large prediction errors much more leniently in comparison to its closest counterpart of the RMSE metric (Hernández del Olmo & Gaudioso, 2008).

Thus, the respective research hypotheses have been specified to be the following:

– The application of the collaborative filtering method of the SVD on the numerical ratings will lead to the best results in the core accuracy metric of MAE as compared against the other factorisation methods of the SVD++ and the Non-Negative Matrix Factorisation as well as against the other common collaborative filtering methods of the user-based and item-based KNN, as it was proved to perform better than other CF algorithms on sparse rating datasets in other domains such as film recommendations (Billsus & Pazzani, 1998; Cacheda et al., 2011; Koren et al., 2009; Luo et al., 2014);

– The application of the topic modelling technique of the Latent Dirichlet Allocation on the textual reviews will reveal a more concise spectrum of the types of attractions enjoyed by users as compared to the 15 types that are specified as the default ones on the TripAdvisor platform without much loss in the coherence of the uncovered topics, as it was proved efficient in uncovering latent topic dimensions (Blei et al., 2003).

Consequently, in line with the research question analysis and the testing of its hypotheses, the main objective of the present thesis is to explore and demonstrate the benefits of the latent factor models for producing user recommendations in the tourism domain, by constructing a fully functional recommender system on the core basis of a top-performing collaborative filtering factorisation algorithm.

Finally, as a means of achieving this objective in a step-by-step process, the list of specific tasks has been proposed to be the following:

a) Firstly, to web scrape user ratings and reviews data from the TripAdvisor platform, specifically, on the landmarks and experiences of London, UK, with the primary help of the Python library requests and parser library BeautifulSoup, as well as to describe and visualise the results of the respective data analysis, mainly in order to assess the degree of the data sparsity problem and make appropriate adjustments;

b) Secondly, to apply the previously reviewed range of collaborative filtering algorithms on the user-item interaction matrix of single numerical ratings as well as to evaluate and compare the performance of the resulting predictive algorithms on the basis of the commonly used accuracy metric of MAE, that way justifying the ultimate choice of a single (or a range) of specific recommender algorithm(s) that will have produced the most accurate ratings predictions for the unknown test users;

c) Thirdly, to apply the topic recognition algorithm of the Latent Dirichlet Allocation on the array of the users’ textual reviews for all of the London’s tourist attractions, in order to discover a more relevant and precise list of attraction types according to the UK travellers, than the general one provided on the TripAdvisor platform;

d) Fourthly, to enhance the travel recommender system’s performance by executing the following tasks: solving the cold start problem for the actual new users by pre-filtering them according to the newly discovered list of attraction types; as well as ensuring the diversity of user recommendations by penalising the recommendation algorithm for suggesting the most popular attractions;

e) And lastly, to develop a fully-fledged practical solution for travel recommendations in the form of an interactive online service application, on the core basis of a single recommendation algorithm which not only showed the highest accuracy of ratings predictions, but also ranked first according to the relevancy metrics of users’ recommendation lists.

# **4. Methodology**

In line with the single core objective of the current thesis of exploring and demonstrating the exceptional benefits of the latent factor models for producing user recommendations in the tourism domain of study, the exact factorisation methods, which will be employed for the purpose of conducting the present study, have previously been reviewed on the basis of their successful applications in the other domains of research, especially, for film recommendations. Specifically, the present thesis has chosen a total of three factorisation models, which are to be considered as the potential core filtering algorithm of the future recommender system, namely, the SVD, SVD++ and the Non-Negative Matrix Factorisation methods. The three primary reasons that were considered in making this choice were the facts that: firstly, these are some of the few models that are suitable to be applied in the present case of extremely sparse tourism-related data which can only be effectively represented as a user-item interaction matrix; secondly, these three models out of the rest of the more complex factorisation techniques were tested to be the only ones accessible enough in terms of the computational and temporal constraints of the present thesis; and thirdly but not least importantly, the three of these models were assumed to be sufficient as they have been the most common ones to be featured in the previous recommender studies in other domains, especially of film recommendations, where they have been extensively tested with much success against other collaborative filtering algorithms (Billsus & Pazzani, 1998; Cacheda et al., 2011; Koren et al., 2009; Luo et al., 2014).

In addition, in order to showcase and solidify the superiority of the top-performing latent factor model within this study as well, other classical collaborative filtering methods of comparable prediction accuracy and computational complexity have been chosen to be tested and evaluated on the same data alongside the latent factor models. Specifically, these other methods have been selected to be the two closely related ones of the user-based and item-based k nearest neighbours’ algorithms, which are often compared with and were even proposed to be merged with the abovementioned most accessible matrix factorisation methods as a factorisation of a neighbourhood model (Koren, 2010).

When it comes to the choice of the performance evaluation metrics, first and foremost, from the two reviewed accuracy metrics for ratings predictions this research has settled on the reasonable choice of the MAE metric, as it has a more uniform error calculation approach, treating singular instances of large prediction errors much more leniently in comparison to its closest counterpart of the RMSE metric (Hernández del Olmo & Gaudioso, 2008).

Thus, the hypotheses for each of the research questions have been specified to be the following:

– The application of the collaborative filtering method of the FunkSVD on the numerical ratings will lead to the best results in the core accuracy metric of MAE as compared to the other factorisation methods of the SVD++ and the Non-Negative Matrix Factorisation as well as to the other common collaborative filtering methods of the user-based and item-based k nearest neighbours’ algorithms;

– The application of the topic modelling technique of the Latent Dirichlet Allocation on the textual reviews will reveal a more concise spectrum of the types of attractions enjoyed by users as compared to the 15 types that are specified as the default ones on the TripAdvisor platform without much loss in the coherence of the uncovered topics.

To start with the data collection process, first and foremost, the mining of the TripAdvisor data had to be done completely by hand for the primary reason that the access to the TripAdvisor API, independent of the type of purpose (whether for consumer analysis, academic research or business application), would have had to be requested and paid for. Thus, the data collection process consisted in parsing the user review data to comprise a sample of, for the most part, 150 reviews per each of the 975 unique landmarks and experiences available for the city of London, UK on the TripAdvisor platform, which has amounted to a little over 42 per cent of the total population figure of 2318 respective tourist attractions. It is important to note that the representativeness of the sample was ensured by the simple fact that the user reviews on TripAdvisor are sorted from the most to the least recent ones by default, which allowed to collect a representative dataset of user reviews from all the various types of travellers who posted over the span of the whole past year. That way the collection of the unwanted and skewed sample of season-specific and traveller-type-specific review data was successfully avoided. The only data collection tool, employed to partially automate the process, was a straightforward web-scraping programme, which was written in the Python programming language with the only help of the BeautifulSoup library package as well as the Selenium web-testing framework. As a result, the two datasets of user reviews, with the one containing the numerical ratings of users and the other comprising the respective textual reviews, were represented as the two matrices, which reflected every single interaction (or an absence of such) between the total figures of the 47908 unique users and the 975 unique attractions.

Now speaking in detail about the core process of developing the chosen predictive algorithms, first and foremost, the sample dataset that contains the users’ numerical single-ratings for London’s landmarks and experiences was separated into the training and test sets according to the most optimal relation of 80 to 20 per cent respectively (Cacheda et al., 2011). For the purposes of training as well as analysing the performance of the total of six pre-selected machine learning models, the whole of the employed toolset is comprised of the two sets of add-on Python packages of SciPy Toolkits, with the one being the sklearn library, while the other is collectively entitled as Surprise, which is a rough abbreviation of the Simple Python Recommendation System Engine. The two libraries of Surprise and sklearn allow the choice of incorporating all of the necessary model features of the previously reviewed matrix factorisation and k nearest neighbour algorithms respectively. Subsequently the present section of the thesis presents all of the selected models, accompanied by a brief specification list of their exact model features and modes of application, the purpose and the advantages of which have already been established and discussed in detail in the literature review section of the present thesis.

To begin with, the FunkSVD method is described to function on the basis of the following bullet list of setting modes and model features:

a) Accounts for the independent user and item biases;

b) Employs the primary mode of the L1 regularisation method;

c) Achieves the optimised minimisation of the predictions’ squared error with the help of the SGD learning algorithm;

d) Predicts the deviation from the mean rating value and then further exploits that same mean rating value in the prediction of the user’s actual rating;

e) Sets a rating value to the training dataset’s global mean rating in the cases when a given user or item from the test set is not present in the training set;

f) Updates both user and item matrix values simultaneously (Koren et al., 2009).

Secondly, the model of SVD++ is a direct extension of the FunkSVD algorithm, for which reason it enumerates the same points that have just been listed above with one key addition to the calculation of predicted ratings. Specifically, the implicit ratings of users are accounted for as well, by defining the users’ factor vectors through the additional term, which assumes the value of 1 if a given user has rated a certain item regardless of the actual rating value and, on the opposite, turns to a value of 0 if a given user has not rated a certain item (Koren, 2008).

Thirdly, the model of NMF, which is an alternative to the same FunkSVD algorithm, also employs the optimisation procedure of the SGD learning method to minimise the regularised squared prediction error. However, the integral difference of the NMF model is that it restricts the factorised matrices of users and items, which form the basis for rating predictions, to only include the non-negative values, filtering out the negative ones altogether (Luo et al., 2014).

And lastly, the two k nearest neighbours’ techniques, namely, the item-based and regular as well as weighted user-based ones, were trained on the same data based on the measure of cosine distance, rather than on its comparable alternative of Euclidian distance measure, as its use would be more favourable in the present case of the highly sparse and multi-dimensional configuration of the sample dataset (Cacheda et al., 2011). Starting with the regular user-based k nearest neighbours, since it is impossible to test the algorithm by asking users to rate new recommended items and calculate the precision and recall metrics, only users with 4 and more ratings were left in the data set, which amounted to a total of 2256 users. Then, for each user a pseudo-random sample of 3 out of all the rated items was created, while the other ratings were set to 0. Similar users were calculated based on the three remaining ratings. Subsequently, the predicted ratings that were initially set to 0 were calculated as an average of the ratings on a specific attraction which was posted by the 10 most similar users. If not even a single one of the most similar users have rated that particular attraction, the predicted rating was calculated as an average rating across the most similar users. The weighted user-based nearest neighbour differs from the regular one in the fact that each rating posted by a similar user was multiplied by the inverse distance between them and the target user and, then, in order to get the average target rating, the sum of weighted ratings was divided by the total weight.

Now speaking in short on the execution of the fifth and last machine learning algorithm of the item-based k nearest neighbour that was tested within this thesis, it used a transposed rating matrix where rows represent items and columns represent users. For each user only their rated items were selected, their ratings set to 0 and similar items found. From the rating vectors of the similar items, the selected user’s ratings were extracted and the average of them used as a predicted rating. Had the user not rated any of the similar items, their rating of the item in question was predicted as its average rating with the user’s input still set to 0.

Finally, as the second research hypothesis is concerned with showcasing the tourism domain-related benefits of the latent factor-based semantic analysis technique of text classification known as the Latent Dirichlet Allocation (LDA), the methodology of the Python’s statistical natural language processing tools, as provided by the two Python libraries of the Gensim as well as of the Natural Language Toolkit, was adapted for the goals of the current thesis. Speaking in detail, the LDA approach is divided into the two consecutive stages of: firstly, pre-processing of the textual review data and, secondly, the actual algorithm execution (Jelodar et al., 2019).

To start with the pre-processing stage, first and foremost, the reviews on each unique tourist sight were concatenated into one single text, also formally called a document, amounting to a total figure of 975 texts – the same as the total number of unique London’s attractions within the data sample. As the next data cleaning stage a series of actions was performed on that semantic data, namely: all of the stop words (or terms) were removed; also, every single word was stemmed to its basic form by removing all the prefixes and suffixes of words; as well as the stems that appeared less than 10 times within the whole single text were removed. Lastly, every document was represented as a so-called Bag of Words (BoW), which is the name for the representation of singular stemmed words as tuples of the word ids and the number of times a specific word was encountered within a specific document (Guo et al., 2016).

The core stage of the execution of the LDA algorithm unravelled in the following series of consecutive steps (Blei et al., 2003):

a) Every term of every document is randomly attributed to one of the *k* number of topics.

b) The algorithm assumes that the topics are attributed inappropriately for every currently performed-on document, while the topic assignment for rest of the document base is correct.

c) The fraction of terms in a document that have been collected under a certain topic is computed.

d) The fraction of a specific words’ assignments for a specific topic is computed for the whole of the document base.

e) The two proportions, specified at the third and fourth stages, are multiplied and their product is assumed to denote the probabilities of certain terms being assigned to certain topics.

f) This 5-stage process is cycled through again and again until the point when a repeated steady pattern of topic assignments is established.

To find the optimal number of topics to train the model on, coherence score was used as it had been proved to have the highest correlation with coherence ratings given by humans (Röder et al., 2015).

Ultimately, the goal is to propose the LDA topic modelling algorithm as the supplementary latent factor-based method that solves the cold start problem as well as enables the diversity of recommendations. To this end, the number of reviews are normalised and then multiplied by ratings to produce popularity scores the following way:

Then, in order to compile a non-trivial recommendation list out of the least reviewed of the London’s sights, and yet the ones with the highest ratings among them, the normalised values are inversed:

Thus, when a user picks their preferred categories or, in other words, the newly discovered topics, the recommendations will be sorted by descending popularity (or inversed popularity if they choose to see the least known yet highly rated of the London’s sights). Furthermore, if a user selects not one but several categories, in order to first present them with the attractions where all of the chosen topics are most equally present and have the highest probability scores, the following formula will be applied to sort the sights in the descending order of its output:

;

where: *min probability* – is each attraction’s smallest probability score across the selected topics;

*max probability* – the highest probability across the selected topics.

To produce the final coefficient value that will be used to sort the recommended attraction list, the topic distribution coefficientwill be multiplied either by popularity of inversed popularity based on the user’s choice.

# **5. Description of the results**

## **5.1 Matrix factorisation models**

First of all, the matrix factorisation algorithms were trained and tested to reveal the optimal setting of the models’ parameters with the single goal of yielding the most accurate results of the models’ performance, according to the major accuracy metric of MAE.

To begin with, the first one out of the number of matrix factorisation models to be trained and assessed was the FunkSVD algorithm. This unsupervised learning model, once trained, had to be optimised across, firstly, the number of factors and the value of the regularisation term as well as then, with those two parameters being fixed, across the range of different learning rates. The bullet point summary of the key findings for the evaluation process of the FunkSVD model is provided below (see appendix 1):

a) The higher the value of the regularisation term (with any number of features from 2 to 50), the higher the MAE value;

b) The smallest MAE value of 0.698 was yielded with 7 factors and regularisation parameter of 0.005 when learning rate was fixed at 0.001;

c) The best learning rate with the optimal number of 7 factors and the regularisation parameter of 0.005 was revealed at the training mark of 50 epochs and was equal to 0.005. The MAE at this optimal learning rate was shown to be 0.688, which is the best result across all iterations of parameter tuning.

Secondly, the next one out of the number of matrix factorisation models to be trained and assessed was the SVD++ algorithm, which was optimised through the regularised SGD just as the previous model of the FunkSVD. Moreover, similar to the FunkSVD, this unsupervised learning model, once trained, had to be optimised across, firstly, the number of factors and the value of the regularisation term as well as then, with those two parameters being fixed, across the range of different learning rates. The bullet point summary of the key results for the evaluation process of the SVD++ model is provided below (see appendix 2):

a) The best MAE value of 0.699 was recorded by a model configuration of 3 factors with the regularisation parameter of 0.005 and the learning rate of 0.001;

b) Thus, SVD++ algorithm has shown just a slightly worse performance than the FunkSVD method with the difference in the MAE value amounting to 0.11.

Lastly, the final one out of the number of matrix factorisation models to be trained and assessed was the NMF algorithm. Unlike the previous two methods, this unsupervised learning model only allows for non-negative factor values, which most probably in a major way contributed to its much lower overall performance, according to the MAE accuracy metric, than that of other evaluated matrix factorisation models. The bullet point summary of the key results for the evaluation process of the NMF model is provided below (see appendix 3):

a) The NMF algorithm has shown an overall higher values of MAE across all the variations of the number of factors than that of the FunkSVD and the SVD++ algorithms, its lowest MAE value being 0.862;

b) When trained with the parameters that yielded the best performance of the FunkSVD algorithm (7 factors with the regularisation and learning rate of 0.005), NMF model produced MAE value of 1.432.

## **5.2 K-nearest-neighbours models**

The first KNN model that implemented was user-based filtering based on 10 neighbours. With the base sample (i.e. the sample of ratings posted by each user that is used to compute 10 similar users) of 3, which required to leave in the dataset only the users with 4 or more ratings (8807 users), the MAE amounted to 0.692. When only users with 3 or more ratings and samples of size 2 were used (3910 users), the MAE amounted to 0.730 and almost the same result of 0.725 was yielded when similarity was calculated only by one rating for each user (8807 users with 2 or more ratings). These results are summarised in the Table 1. Thus, the lowest MAE produced by the user-based KNN is 0.692 (Table 1).

Table 1

Performance results of the regular user-based KNN model

|  |  |  |
| --- | --- | --- |
| Base sample of ratings | Number of users | MAE |
| 1 | 8807 | 0.725 |
| 2 | 3910 | 0.730 |
| 3 | 2256 | 0.692 |

After the weights were introduced to the algorithm in an attempt to improve its performance, the MAE values increased for each of the three groups of base samples (Table 2). When similarity of users was calculated based on one rating, the MAE amounted to 1.125. The base samples of 2 and 3 ratings showed better performance, yielding MAE of 0.749 and 0.782. Thus, with its best rating prediction on average deviating from the actual ratings by 0.749, the weighted user-based algorithm based on cosine distance proved inferior to the regular user-based KNN on the given dataset.

Table 2

Performance results of the weighted user-based KNN model

|  |  |  |
| --- | --- | --- |
| Base sample of ratings | Number of users | MAE |
| 1 | 8807 | 1.125 |
| 2 | 3910 | 0.749 |
| 3 | 2256 | 0.782 |

Since each item was rated at least 4 times, item-based KNN, unlike the user-based one, did not require a base sample of ratings to calculate MAE and, therefore, it was possible to test it on the whole dataset. For each item the similar ones were computed based on at least three ratings, with the rating posted by the user who is getting recommendations set to 0. When the user-item matrix was transposed and the algorithm performed, the MAE value amounted to 0.699. Even though it is 0.007 points less than the best performing user-based KNN, this algorithm has the advantage of using the whole dataset including the users who rated only one item, and the MAE produced is a stable value since its calculation did not employ a random sample that leads to different MAE values based on what items were randomly selected to be used to compute the distance at each iteration.

## **5.3 LDA model**

To derive the best value for the main LDA hyperparameter, the number of topics, coherence scores were calculated for models with 2 to 20 topics. Even though it is possible to produce coherent word-to-topic allocations for more than 20 topics, the primary purpose of this analysis was to derive a number of groups of attractions that would be presented to a new user upon their first login to the system. Thus, in order to avoid overwhelming them with a long list of attraction groups, word allocations into topics above 20 were not tested. The number of training iterations for the algorithm was set to 10 during the coherence score computation due to the limited computational resources and the fact that generally higher numbers of iterations tend to increase model performance instead of decreasing it.

The best coherence score of 0.535 was shown by the model with 16 topics, another model scoring above 0.53 but slightly less being the one with 17 topics (Figure 1). However, since all other models whose number of topics was above 10 did not perform that well, the third best performing model was chosen. The selected model divided documents into 6 topics and produces a coherence score of 0.519. The reason for choosing it over those with 16 and 17 topics was to avoid inundating users with a large number of highly specific location groups. Moreover, as TripAdvisor offers 15 categories that the attractions fall into and the goal of applying LDA in this paper was to produce a smaller number of topics without losing their representativeness, 6 topics were considered the optimal choice.

A close up of a mans face

Description automatically generated

Figure 1. Coherence scores at different number of topics

After selecting the most appropriate and high performing number of topics, various numbers of training iterations (i.e. passes) from 10 to 50 were tested and the one producing the highest coherence score selected. Increasing the number of passes to 50 improved the score by 0.039 points and, therefore, it was used to train the model (Figure 2).

A screenshot of a cell phone

Description automatically generated

Figure 2. Coherence scores at different number of passes

The word allocations produced by the algorithm are presented in Table 3. The words are listed in the descending order of the weights of their contribution to the topics. The initial model output contained stems that were then manually completed to represent actual words occurring most frequently in each of the documents allocated to a particular topic. The topics were indexed from 0 to 5 and the names were assigned to them by the authors of this thesis based on their perception of the words in the model output.

Table 3

Allocation of nouns according to different topics

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| TOPIC 0  Landmarks | TOPIC 1  Art | TOPIC 2  Tours | TOPIC 3  Food | TOPIC 4  Nature | TOPIC 5  Performing arts |
| church  square  statue  memorial  palace  gin  Westminster  monument  guard  architecture | museum  exhibit  tour  galleries  art  information  guide  display  collect  fascinating | tour  bridge  guide  Thames  tower  river  fan  ticket  stadium  informative | market  restaurant  stall  eat  bar  store  service  beer  buy  train | garden  cafe  animals  children  kid  relax  green  play  Greenwich  canal | theatre  seat  play  perform  venue  music  bar  ticket  product  stage |

Next, a model containing only adjectives and adverbs as well as their comparative and superlative forms was trained in order to derive travel styles that would describe the users. Based on the coherence scores for the numbers of topics from 2 to 20, the most highly interpretable model was the one with 4 topics. However, the scores produced by each of these models were significantly lower than those of the models trained at the previous step, with the lowest and highest coherence scores amounting to 0.267 and 0.378 respectively. The model did not produce an interpretable result (Table 4).

Table 4

Allocation of adjectives and adverbs according to different topics

|  |  |  |  |
| --- | --- | --- | --- |
| TOPIC 0 | TOPIC 1 | TOPIC 2 | TOPIC 3 |
| local  green  central  quiet  atmospheric  clean  cafe  south  fresh  nearby | informative  knowledgeable  helpful  brilliant  top  expensive  enjoyable  interactive  own  able | historical  British  famous  modern  memorial  original  national  queen  royal | brilliant  comfortable  theatrical  helpful  musical  west  funny  fabulous  every  intimate |

Based on the output of the two models, the one trained without excluding any parts of speech was selected to be used in the recommender system. The distribution of the topics produced by the LDA model among the categories of attractions presented on TripAdvisor is shown in Appendix 5. The categories offered on TripAdvisor with the numbers of sights that belong to them in the collected dataset are listed in Appendix 6.

Most of the sights classified as Art by the model fall into the TripAdvisor Museums and Sights & Landmarks categories. Attractions marked as Tours are represented in 12 out of all 15 of the TripAdvisor categories, with the majority of them falling into the Sights & Landmarks group. Landmarks are also represented mostly among Sights & Landmarks while Food attractions cover 12 categories with the top four most populated ones being Shopping, Sights & Landmarks, Other and Food & Drink. Sights marked as Nature are mostly categorised by TripAdvisor as Nature & Parks and Sights & Landmarks. Finally, the majority of attractions grouped into Performing arts fall into the Concerts & Shows category (see appendix 6).

These distributions were derived by assigning each attraction with one topic that had the highest probability as given in the model output. However, some attractions are characterised by high probability of two or more different topics and thus can be categorised into different attraction groups (see appendix 7). This feature will be used in the recommender system to provide users who select more than one attraction group at the login screen with recommendations that fit all or at least two of the chosen categories at the same time, while those who select just one attraction type will be presented with the locations that have the highest probability of falling into the chosen topic. These lists will be ordered based on the popularity scores as presented in the methodology.

## **5.4 Discussion of the results**

The interpretation of the results relating to the research questions and the hypotheses is the following. First, FunkSVD was expected to show better performance on the TripAdvisor rating dataset than SVD++ and NMF. While the best performing instances of the latter models produced MAE values of 0.699 and 0.862 respectively, the lowest MAE produced by FunkSVD amounted to 0.688, which allows to confirm the first part of the first hypothesis.

Secondly, FunkSVD was expected to perform better than the user- and item-based k-nearest neighbours, which are the common benchmark algorithms for the comparison of collaborative filtering methods. Three different KNN models were tested and their MAE values were calculated. Weighted user-based KNN showed the worst performance results with its lowest MAE being 0.749. Item-based KNN produced the mean error of 0.699, falling behind the regular user-based KNN by 0.007 points. However, even the best result of 0.692 showed by the user-based KNN model with the base sample of 3 ratings is still slightly worse than the error of 0.688 yielded by FunkSVD. Thus, the second part of the second hypothesis is confirmed, proving FunkSVD to be the best performing model on the TripAdvisor attraction ratings dataset out of the selected six recommender algorithms.

Finally, the Latent Dirichlet Allocation for topic modelling allowed to uncover a total of 6 coherent latent topics in the reviews that supplemented the rating prediction. The final range of six topics is the following: “Landmarks”, “Art”, “Tours”, “Nature”, “Food” and “Performing arts”. While half of these attraction categories, such as “Art”, “Landmarks” and “Performing arts” are categorised by TripAdvisor into the similar groups of “Museums”, “Sights & Landmarks” and “Concerts & Shows” respectively, the other three types, in their turn, are distributed between up to 12 different categories from the TripAdvisor website, specifically: with “Tours” being the most diverse category and in small portions including the “Museums”, “Fun & Games” and “Transportation”; with “Food” being comprised primarily from “Food & Drinks” and “Shopping” categories; and with “Nature” largely consisting of the “Nature & Parks”, “Fun & Games” as well as “Zoos & Aquariums”. The highest coherence score of 0.535 was shown by the 16-topic model which can be supported by the fact that TripAdvisor also divides the sights into 15 categories. However, since the goal of this study was to produce a smaller number of groups that would be presented to a new user at the login screen with the minimal loss in the algorithm performance, 6 topics were considered the optimal choice. Thus, it can be concluded that the application of LDA helped produce a smaller number of attraction categories while the loss in the coherence score amounted to only 0.016 points.

To sum up, the highest performing collaborative filtering algorithm that was able to produce the most accurate predictions of user ratings on the TripAdvisor dataset of attractions in London, UK is FunkSVD matrix factorisation model. Incorporation of the semantic analysis performed by the LDA algorithm allowed to group the attractions into a smaller number of categories than that presented on the platform. This classification will allow to mitigate the problem of the cold start by providing new users with the choice of their most preferred attraction types and then presenting them either by their descending popularity or by the inverse of popularity based on the user’s preference.

Speaking in detail about the prototype of the application interface that allows any given user to interact with the recommendation system, first of all, on the starting screen, a given user is presented with the range of six options for different types of attractions according to the results of the LDA model, namely: “Landmarks”, “Art”, “Food”, “Nature”, “Tours” and “Performing Arts” (see appendix 7). The user can either select just one category and be presented with a list of attractions that score the highest in that particular topic, or choose a combination of several categories up to all six of them, in which case an output will be produced in the form of a list of tourist sights sorted from those where topic probabilities are the most evenly distributed across the chosen categories to those where one of the topics prevails. In addition, the user is also provided with the tick option of sorting the chosen type of tourist attractions according to the inverse of their popularity, as opposed to the standard sorting by decreasing popularity, which will rank and display the least reviewed but highly rated item recommendations to be the first. Secondly, after the preliminary stage of pre-filtering the attractions according to the type that the new user has chosen (content-based filtering), the new user receives a list of recommended items, any of which he/she can rate by clicking on the icon to the right of every attraction, that way creating their profile and writing their ratings into the system’s database (see appendix 8). And finally, after the core stage of employing the FunkSVD model to predict the new user’s ratings on the initial basis of the total of at least one attraction rating (collaborative filtering), the user is transferred to the third and final screen, which displays the list of recommended attractions ranked according to the highest predicted ratings, which can be appropriately rated by the user in the similar way as on the second screen (see appendix 9). The user can also navigate to the tab that contains the items which he/she has previously rated, being displayed with their respective actual rating values (see appendix 10).

For clarity, it is important to note that, unlike the LDA method which has been performed offline and only one single time, the core prediction algorithm of the FunkSVD trained model has to be constantly computing the rating predictions for every new user in the real time, based on the newly added data of the new user’s first rated attraction(s) and then each time a user adds a new rating.

# **6. Conclusion**

This research thesis has set out to explore the unique application benefits of the machine learning models based on the latent factors for the development of a recommender system in the tourism domain. As the promising results of such factorisation algorithms had been extensively demonstrated on the film-related data, the present research was dedicated to adapting those models to a much less explored domain of travel recommendations. Thus, in order to achieve this objective, the TripAdvisor platform, which is rightfully considered to be the most visited travel-related online resource in the world, currently enjoying more than 490 million active users every month, was chosen as the only source of data for the construction of a sample dataset.

In the process of conducting a comparison study of the target models of matrix factorisation against each other as well as against the similarly efficient collaborative filtering algorithms of the nearest neighbours, the research’s core hypothesis, which postulated the ultimate superiority of the famous FunkSVD factorisation method in accurately predicting the single numerical user ratings for the London’s tourist attractions, has been successfully proven. In addition, the LDA technique has discovered a more concise and representative range of generalised topics for the types of London’s landmarks and experiences as compared to the TripAdvisor’s respective default categories. Moreover, this newly discovered classification has been adapted to pre-filter the system’s new users according to their preferences in regard to the different types of attractions. Thus, the machine learning model of the LDA has yielded some very fruitful results, which have been successfully incorporated into the recommender system to solve the cold start problem.

When critically assessing the research’s employed methodology and the received results, it should be pointed out that the study managed to conduct the comparison between a very limited range of only six collaborative filtering algorithms on the present-case travel data, while essentially only relying on a single point of comparison in the form of the MAE accuracy metric. Furthermore, the proof of the usefulness and applicability of the results of the LDA algorithm admittedly hinges on the fact of the model’s inherently high efficiency on any type of semantic data, which, however, does not discredit the algorithm’s findings as it is meant for the generalised predictions of a text’s main topics, the process of which can only be sufficiently optimised, but not evaluated according to the accuracy of its results.

The major limiting factors of the performed research study have been the computational constraints of the available resources, which have noticeably restricted the volume of user-item data that could be parsed from the TripAdvisor platform and which have also cancelled out some of the types of model configurations that could not be properly assessed within this study. Thus, it is left for the future research to attempt increasing not so much the size of a sampled dataset as the degree of density of the user-item matrix, for instance, by means of scraping the review data of the most prolific users via the TripAdvisor API; as well as to try testing other more computationally demanding configurations of the matrix factorisation algorithms, such as, optimising the factorisation models by employing the Alternating Least Squares (ALS) learning method. Another significant limiting factor was the inability to test the proposed system online to collect the data necessary to assess the real performance of the model through the precision and recall metrics. Even though the real world testing was simulated through eliminating parts of the collected ratings and calculating distances based on the remaining ones while predicting the excluded ratings, calculating precision and recall that way would have led to unreliable results as it would have required only considering users who rated at least 8 or 10 items which there were very few of.

Some of the future improvements of the presently developed recommender system prototype are proposed to be the following. Firstly, the modification of the present recommender system into a hybrid one by means of, for instance, incorporating the LDA results even further into the recommendation algorithm, which could consist in providing users with the recommendations of those items that have similar topic distributions to the ones they have already rated highly (content-based filtering), in addition to the core collaborative filtering algorithm of the matrix factorisation-like model of the FunkSVD. Secondly, to further hybridise the model, introducing content filtering based on the data on the attractions and adding a set of optimised weights to each model’s output might increase recommendation accuracy and diversity. Thirdly, it would also be interesting to extend the system’s capabilities to include the option of optimally mapping a potential tour across the city of London based on the user’s personalised recommendation list of tourist landmarks and experiences. Lastly and perhaps more importantly, it is paramount for the future development of the prototype to launch the travel recommender system in the current state online for the actual new users to test, as this will enable the application of the recommendation to set relevance measures, such as the precision and recall, allowing the subsequent enhancement of the predictive algorithm.

# **Reference list**

Adomavicius, G., & Tuzhilin, A. (2005). Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering, 17*(6), 734–749. doi: 10.1109/TKDE.2005.99

Ahani, A., Nilashi, M., Othman, I., Sanzogni, L., & Weaven, S. (2019a). Market segmentation and travel choice prediction in Spa hotels through TripAdvisor’s online reviews. International Journal of Hospitality Management. *International Journal of Hospitality Management, 80*, 52–77. doi: 10.1016/j.ijhm.2019.01.003

Ahani, A., Nilashi, M., Yadegaridehkordi, E., Sanzogni, L., Tarik, A., Knox, K., Sarminah, S., & Othman, I. (2019b). Revealing customers’ satisfaction and preferences through online review analysis: The case of Canary Islands hotels. *Journal of Retailing and Consumer Services, 51*, 341-343. doi: 10.1016/j.jretconser.2019.06.014

Al Mamunur, R., Istvan, A., Cosley, D., Lam, S. K., McNee, S. M., Konstan, J. A., & Riedl, J. (2002). Getting to Know You: Learning New User Preferences in Recommender Systems. *Proceedings of the 7th International Conference on Intelligent User Interfaces*, 1–9. doi: 10.1145/502716.502737

Ali, F., Kwak, K., & Kim, Y. (2016). Opinion mining based on fuzzy domain ontology and Support Vector Machine: A proposal to automate online review classification. *Applied Soft Computing, 47*, 235-250. doi: 10.1016/j.asoc.2016.06.003

Baltrunas, L., Makcinskas, T., & Ricci, F. (2010). Group recommendations with rank aggregation and collaborative filtering. *Proceedings of the 4th ACM conference on Recommender systems*, 119–126. doi: 10.1145/1864708.1864733

Billsus, D., & Pazzani, M. (1998). Learning Collaborative Information Filters. *Proceedings of the Fifteenth International Conference on Machine Learning*, 46–54. Retrieved from: https://www.aaai.org/Papers/Workshops/1998/WS-98-08/WS98-08-005.pdf

Blei, D. M., Ng, A. Y., & Jordan M. I. (2003). Latent Dirichlet Allocation. *Journal of Machine Learning Research, 3*(5), 993-1022. doi: 10.1162/jmlr.2003.3.4-5.993

Bobadilla, J., Ortega, F., Hernando, A., & Gutiérrez, A. (2013). Recommender systems survey. *Knowledge-Based Systems, 46*, 109-132. doi: 10.1016/j.knosys.2013.03.012

Bottou, L. (2010). Large-Scale Machine Learning with Stochastic Gradient Descent. In Lechevallier Y., & Saporta, G. (Eds). *Proceedings of COMPSTAT'2010* (pp. 177-186). Springer-Verlag Berlin Heidelberg. doi: 10.1007/978-3-7908-2604-3\_16

Breese, J., Heckerman, D., & Kadie, C. (1998). Empirical Analysis of Predictive Algorithm for Collaborative Filtering. Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence, 43-52. Retrieved from: https://arxiv.org/ftp/arxiv/papers/1301/1301.7363.pdf

Burke, R. (2002). Hybrid Recommender Systems: Survey and Experiments. *User Modeling and User-Adapted Interaction, 12*(4),331–370. doi: 10.1023/A:1021240730564

Cacheda, F., Carneiro, V., Fernández, D., & Formoso, V. (2011). Comparison of collaborative filtering algorithms: Limitations of current techniques and proposals for scalable, high-performance recommender systems. *ACM Transactions on the Web, 5*(1), 1-33. doi: 10.1145/1921591.1921593

Chang, Y., Ku, J., & Chen, C. (2017). Social media analytics: Extracting and visualizing Hilton hotel ratings and reviews from TripAdvisor. *International Journal of Information Management, 48*, 263-279. doi: 10.1016/j.ijinfomgt.2017.11.001

Chen, Y., Wu, C., Xie, M., & Guo, X. (2011). Solving the Sparsity Problem in Recommender Systems Using Association Retrieval. *Journal of Computers 6*(9), 1896–1902. doi: 10.4304/jcp.6.9.1896-1902

Cheng, A., Chen, Y., Huang, Y., Hsu, W., & Liao, H. (2011). Personalized travel recommendation by mining people attributes from community-contributed photos. *Proceedings of the 19th ACM International Conference on Multimedia*, 83–92. doi: 10.1145/2072298.2072311

Díez, J., Pérez-Núñez, P., Luaces, O., Remeseiro, B., & Bahamonde, A. (2020). Towards Explainable Personalized Recommendations by Learning from Users’ Photos. *Information Sciences, 520*, 416-430. doi: 10.1016/j.ins.2020.02.018

Esmaeili, L., Mardani, S., Golpayegani, S., & Madar, Z. (2020). A Novel Tourism Recommender System in the Context of Social Commerce. *Expert Systems with Applications, 149*, 1-11. doi: 149. 10.1016/j.eswa.2020.113301

Fawcett, T. (2006). An introduction to ROC analysis. *Pattern Recognition Letters, 27*(8), 861-874. doi: 10.1016/j.patrec.2005.10.010

Fuchs, M., & Zanker, M. (2012). Multi-criteria Ratings for Recommender Systems: An Empirical Analysis in the Tourism Domain. In C. Huemer & P. Lops (Eds.), *E-Commerce and Web Technologies* (Vol. 123, pp. 100–111). Springer Berlin Heidelberg. doi: 10.1007/978-3-642-32273-0\_9

Gavalas, D., Konstantopoulos, C., Mastakas, K., & Pantziou, G. (2014). Mobile recommender systems in tourism. *Journal of Network and Computer Applications, 39*, 319–333. doi: 10.1016/j.jnca.2013.04.006

Ge, M., Delgado, C., & Jannach, D. (2010). Beyond accuracy: Evaluating recommender systems by coverage and serendipity. *Proceedings of the 4th ACM Conference on Recommender Systems*, 257-260. doi: 10.1145/1864708.1864761

Golbandi, N., Koren, Y., & Lempel, R. (2011). Adaptive bootstrapping of recommender systems using decision trees. *Proceedings of the 4th International Conference on Web Search and Web Data Mining*, 595-604. doi: 10.1145/1935826.1935910

Goldberg, D., Nichols, D., Oki, B. M., & Terry, D. (1992). Using collaborative filtering to weave an information tapestry. *Communications of the ACM*, *35*(12), 61–70. doi: 10.1145/138859.138867

Gorrell, G. (2006). Generalized Hebbian Algorithm for Incremental Singular Value Decomposition in Natural Language Processing. *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics*, 430-451. Retrieved from: https://www.aclweb.org/anthology/E06-1013.pdf

Grčar, M., Mladenić, D., Fortuna, B., & Grobelnik, M. (2006). In Nasraoui O., Zaïane O., Spiliopoulou M., Mobasher B., Masand B., & Yu P.S. (Eds). Data Sparsity Issues in the Collaborative Filtering Framework. Advances in Web Mining and Web Usage Analysis (LNCS Vol. 4198, pp. 58-76). Springer, Berlin, Heidelberg. doi: 10.1007/11891321\_4

Guo, Y., Barnes, S., & Jia, Q. (2016). Mining Meaning from Online Ratings and Reviews: Tourist Satisfaction Analysis Using Latent Dirichlet Allocation. *Tourism Management. 59*, 467-483. doi: 10.1016/j.tourman.2016.09.009

Hamel, S., & Robino, D. (2019). Global Destination Cities Index Report 2019. Retrieved from: https://newsroom.mastercard.com/wp-content/uploads/2019/09/GDCI-Global-Report-FINAL-1.pdf

Herlocker, J., Konstan, J., Terveen, L., & Riedl, J. (2004). Evaluating collaborative filtering recommender systems*. ACM Transactions on Information Systems, 22*(1), 5-53. doi: 10.1145/963770.963772

Hernández del Olmo, F., & Gaudioso, E. (2008). Evaluation of recommender systems: A new approach. *Expert Systems with Applications, 35*(3), 790-804. doi: 10.1016/j.eswa.2007.07.047

Hofmann, T. (2004). Latent semantic models for collaborative filtering. *ACM Transactions on Information Systems, 22*(1), 89–115. doi: 10.1145/963770.963774

Hu, Y., Koren, Y., & Volinsky, C. (2008). Collaborative Filtering for Implicit Feedback Datasets. *Proceedings of the 8th IEEE International Conference on Data Mining*, 263-272. doi: 10.1109/ICDM.2008.22

Jannach, D., Zanker, M., & Fuchs, M. (2014). Leveraging multi-criteria customer feedback for satisfaction analysis and improved recommendations. *Information Technology & Tourism, 14*(2), 119-149. doi: 14. 119-149. 10.1007/s40558-014-0010-z

Jelodar, H., Wang, Y., & Yuan, C. (2019). Latent Dirichlet allocation (LDA) and topic modeling: models, applications, a survey. *Multimedia Tools and Applications, 78*, 15169–15211. doi: 10.1007/s11042-018-6894-4

Kaminskas M., & Bridge, D. (2016). Diversity, Serendipity, Novelty, and Coverage: A Survey and Empirical Analysis of Beyond-Accuracy Objectives in Recommender Systems. *ACM Transactions on Interactive Intelligent Systems, 7*(1), 2-42. doi: 10.1145/2926720

Kbaier, M., Masri, H., & Krichen, S. (2017). A Personalized Hybrid Tourism Recommender System. *2017 IEEE/ACS 14th International Conference on Computer Systems and Applications (AICCSA)*, 244–250. doi: 10.1109/AICCSA.2017.12

Kesorn, K., Juraphanthong, W., & Salaiwarakul, A. (2017). Personalized Attraction Recommendation System for Tourists through Check-in Data. *IEEE Access, 5*, 26703–26721. doi: 10.1109/ACCESS.2017.2778293.

Khusro, S., Ali, Z., & Ullah, I. (2016). Recommender Systems: Issues, Challenges, and Research Opportunities. In K.J. Kim & N. Joukov (Eds.), *Information Science and Applications (ICISA) 2016* (LNEE Vol. 376, pp. 1179–1189). Springer Singapore. doi: 10.1007/978-981-10-0557-2\_112

Koren, Y. (2008). Factorization meets the neighborhood: A multifaceted collaborative filtering model. *Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, 426-434. doi: 10.1145/1401890.1401944

Koren, Y. (2010). Factor in the Neighbors: Scalable and Accurate Collaborative Filtering. *ACM Transactions on Knowledge Discovery from Data, 4*(1). doi: 10.1145/1644873

Koren, Y., Bell, R., & Volinsky, C. (2009). Matrix Factorization Techniques for Recommender Systems. *Computer*, *42*(8), 30–37. doi: 10.1109/MC.2009.263

Lika B., Kolomvatsos K., & Hadjiefthymiades S. (2014). Facing the cold start problem in recommender systems. *Expert Systems with Applications, 41*(4), 2065–2073. doi: 10.1016/j.eswa.2013.09.005

Lin, C. (2007). Projected Gradient Methods for Nonnegative Matrix Factorization. *Neural Computation, 19*(10), 2756-2779. doi: 10.1162/neco.2007.19.10.2756

Logesh, R., Subramaniyaswamy, V., Vijayakumar, V., & Li, X. (2019). Efficient User Profiling Based Intelligent Travel Recommender System for Individual and Group of Users. *Mobile Networks and Applications*, *24*(3), 1018–1033. doi: 10.1007/s11036-018-1059-2

Lops, P., de Gemmis, M., Semeraro, G. (2011). Content-based Recommender Systems: State of the Art and Trends. In Ricci F., Rokach L., Shapira B., Kantor P. (Eds). *Recommender Systems Handbook* (Ch. 3, pp. 73-105). Springer, Boston, MA. doi: 10.1007/978-0-387-85820-3\_3

Lu, J., Wu, D., Mao, M., Wang, W., & Zhang, G. (2015). Recommender system application developments: A survey. *Decision Support Systems, 74*, 12–32. doi: 10.1016/j.dss.2015.03.008

Lu, L., Medo, M., Yeung, C., Zhang, Y., Zhang, Z., & Zhou, T. (2012). Recommender systems. *Physics Reports, 519*(1), 1-49. doi: 10.1016/j.physrep.2012.02.006

Lucas, J., Luz, N., Moreno, M., Anacleto, R., Figueiredo, A., & Martins, C. (2013). A hybrid recommendation approach for a tourism system. *Expert Systems with Applications, 40*(9), 3532–3550. doi: 10.1016/j.eswa.2012.12.061

Luo, X., Zhou, M., Xia, Y., & Zhu, Q. (2014). An Efficient Non-Negative Matrix-Factorization-Based Approach to Collaborative Filtering for Recommender Systems. *IEEE Transactions on Industrial Informatics, 10*(2), 1273-1284. doi: 10.1109/TII.2014.2308433

Meehan, K., Lunney, T., Curran, K., & Mccaughey, A. (2013). Context-Aware Intelligent Recommendation System for Tourism. *Proceedings of the 11th IEEE International Conference on Pervasive Computing and Communications*, 328-331. doi: 10.1109/PerComW.2013.6529508

Pantano, E., Priporas, C., & Stylos, N. (2017). ‘You will like it!’ using open data to predict tourists' response to a tourist attraction. *Tourism Management, 60*. 430-438. doi: 10.1016/j.tourman.2016.12.020

Parikh, V., Keskar, M., Dharia, D., & Gotmare, P. (2018). A Tourist Place Recommendation and Recognition System. *2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT)*, 218–222. doi: 10.1109/ICICCT.2018.8473077

Park, S. T., & Chu, W. (2009). Pairwise preference regression for cold-start recommendation. *Proceedings of the 3rd ACM Conference on Recommender Systems*, 1-8. doi: 10.1145/1639714.1639720

Paterek, A. (2007). Improving Regularized Singular Value Decomposition for Collaborative Filtering. Proceedings of the ACM Press KDD Cup and Workshop, 5(8), 39-42. Retrieved from: http://arek-paterek.com/ap\_kdd\_poster.pdf

Piatetsky-Shapiro, G. (2007). Interview with Simon Funk. *ACM SIGKDD Explorations Newsletter, 9*, 38-40. doi: 10.1145/1294301.1294311

Portugal, I., Alencar, P., Cowan, D. (2018). The use of machine learning algorithms in recommender systems: A systemic review. *Expert Systems with Applications, 97,* 205–227. doi: 10.1016/j.eswa.2017.12.020

Ranjbar, M., Moradi, P., Azami, M., & Jalili, M. (2015). An imputation-based matrix factorization method for improving accuracy of collaborative filtering systems. *Engineering Applications of Artificial Intelligence, 46*, 58-66. doi: 10.1016/j.engappai.2015.08.010

Rendle, S. (2012). Factorization Machines with libFM. *ACM Transactions on Intelligent Systems and Technology, 3*(3), 1-22. doi: 10.1145/2168752.2168771

Renjith, S., & Anjali, C. (2014). A Personalized Mobile Travel Recommender System Using Hybrid Algorithm. *2014 First International Conference on Computational Systems and Communications (ICCSC)*, 12–17. doi: 10.1109/COMPSC.2014.7032612

Rich, E. (1979). User Modeling via Stereotypes. *Cognitive Science, 3*(4), 329–354. doi: 10.1207/s15516709cog0304\_3

Röder, M., Both, A., & Hinneburg, A. (2015). Exploring the Space of Topic Coherence Measures. *Proceedings of the Eighth ACM International Conference on Web Search and Data Mining - WSDM ’15*, 399–408. doi: 10.1145/2684822.2685324

Roopesh, L., & Tulasi, B. (2018). A Survey of Travel Recommender Systems. *International Journal of Computer Sciences and Engineering, 6*(9), 1–7. doi: 10.13140/RG.2.2.34775.32168

Salakhutdinov, R., & Mnih, A. (2007). Probabilistic Matrix Factorization. *Proceedings of the 20th International Conference on Neural Information Processing Systems*, 1257–1264. Retrieved from: https://papers.nips.cc/paper/3208-probabilistic-matrix-factorization.pdf

Sarwar, B., Karypis, G., Konstan, J., & Reidl, J. (2001) Item-based collaborative filtering recommendation algorithms. *Proceedings of the 10th International Conference on World Wide Web*, 285-295. doi: 10.1145/371920.372071

Shani, G., & Gunawardana, A. (2011). Evaluating Recommendation Systems. In Ricci F., Rokach L., Shapira B., & Kantor P. (Eds). *Recommender Systems Handbook* (Ch. 8, pp. 257-297). Springer Boston MA. doi: 10.1007/978-0-387-85820-3\_8

Sharma, R., & Singh, R. (2016). Evolution of Recommender Systems from Ancient Times to Modern Era: A Survey. *Indian Journal of Science and Technology, 9*(20), 1–12. doi: 10.17485/ijst/2016/v9i20/88005

Shaw, G., Xu, Y., & Geva, S. (2010). Using Association Rules to Solve the Cold-Start Problem in Recommender Systems. In M.J. Zaki, J.X. Yu, B. Ravindran, & V. Pudi (Eds). *Advances in Knowledge Discovery and Data Mining* (LNCS Vol. 6118, pp. 340–347). Springer Berlin Heidelberg. doi: 10.1007/978-3-642-13657-3\_37

Shi, Y., Larson, M., & Hanjalic, A. (2014). Collaborative Filtering beyond the User-Item Matrix: A Survey of the State of the Art and Future Challenges. *ACM Computing Surveys, 47*(1), 1–45. doi: 10.1145/2556270

Sinha, R., & Swearingen, K. (2002). The Role of Transparency in Recommender Systems. Proceedings of the CHI EA ‘02 Conference on Human Factors in Computing Systems, 830-831. doi: 10.1145/506443.506619

Sokolova, M., & Japkowicz, N., & Szpakowicz, S. (2006). Beyond Accuracy, F-Score and ROC: A Family of Discriminant Measures for Performance Evaluation. In Sattar A., Kang B. (Eds.). *AI 2006: Advances in Artificial Intelligence* (LNCS Vol. 4304, pp. 1015-1021). Springer, Berlin, Heidelberg. doi: 10.1007/11941439\_114

Wang, Y., Chan, S., & Ngai, G. (2012). Applicability of Demographic Recommender System to Tourist Attractions: A Case Study on Trip Advisor. *Proceedings of the 2012 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology*, 97–101. doi: 10.1109/WI-IAT.2012.133

Xu, Z. (2014). Trip similarity computation for context-aware travel recommendation exploiting geotagged photos. *Proceedings of the 2014 IEEE 30th International Conference on Data Engineering Workshops*, 330-334. doi: 10.1109/ICDEW.2014.6818350

Yang, X., Guo, Y., Liu, Y., & Steck, H. (2014). A Survey of Collaborative Filtering Based Social Recommender Systems. *Computer Communications*, *41*, 1–10. doi: 10.1016/j.comcom.2013.06.009

Zhang, Z. K., Liu, C., Zhang, Y., & Zhou, Z. (2010). Solving the cold-start problem in recommender systems with social tags. *Europhysics Letters, 92*(2), 1–6. doi: 10.1209/0295-5075/92/28002

Zhou, K., Yang, S., & Zha, H. (2011). Functional Matrix Factorizations for Cold-Start Recommendation. *Proceedings of the 34th International ACM SIGIR Conference on Research and development in Information Retrieval*, 315-324. doi: 10.1145/2009916.2009961

# **Appendix 1**

**Evaluation of the ratings prediction accuracy of the FunkSVD algorithm**

**Table 1: Comparison of MAE at n-factors**

|  |  |  |  |
| --- | --- | --- | --- |
| Iteration | Number of factors | Regularisation parameter | MAE |
| 0 | 7 | 0.005 | 0.69767 |
| 1 | 10 | 0.005 | 0.6977 |
| 2 | 5 | 0.005 | 0.698 |
| 3 | 4 | 0.005 | 0.698 |
| 4 | 6 | 0.005 | 0.698 |
| 5 | 32 | 0.005 | 0.698 |
| 6 | 9 | 0.005 | 0.698 |
| 7 | 23 | 0.005 | 0.698 |
| 8 | 30 | 0.005 | 0.698 |
| 9 | 3 | 0.005 | 0.698 |
| 10 | 11 | 0.005 | 0.698 |
| 11 | 12 | 0.005 | 0.698 |
| 12 | 8 | 0.005 | 0.698 |
| 13 | 13 | 0.005 | 0.698 |
| 14 | 16 | 0.005 | 0.698 |
| 15 | 18 | 0.005 | 0.698 |

**Table 2: Comparison of MAE at different learning rates (a 7-factor model with the regularization parameter of 0.05)**

|  |  |
| --- | --- |
| Learning rate | MAE |
| 0.005 | 0.688 |
| 0.010 | 0.697 |
| 0.015 | 0.704 |
| 0.020 | 0.706 |
| 0.025 | 0.713 |
| 0.030 | 0.714 |
| 0.035 | 0.712 |
| 0.040 | 0.714 |
| 0.045 | 0.718 |
| 0.050 | 0.718 |
| 0.055 | 0.718 |
| 0.060 | 0.722 |
| 0.065 | 0.725 |
| 0.070 | 0.724 |
| 0.075 | 0.722 |
| 0.080 | 0.722 |
| 0.085 | 0.725 |
| 0.090 | 0.723 |

# **Appendix 2**

**Evaluation of the ratings prediction accuracy of the SVD++ algorithm**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Iteration | Number of factors | Regularisation  parameter | Learning rate | MAE |
| 0 | 3 | 0.005 | 0.001 | 0.699 |
| 1 | 3 | 0.005 | 0.001 | 0.700 |
| 2 | 4 | 0.010 | 0.001 | 0.700 |
| 3 | 3 | 0.020 | 0.001 | 0.700 |
| 4 | 4 | 0.015 | 0.001 | 0.700 |
| 5 | 3 | 0.010 | 0.001 | 0.700 |
| 6 | 4 | 0.025 | 0.001 | 0.701 |
| 7 | 4 | 0.020 | 0.001 | 0.701 |
| 8 | 3 | 0.015 | 0.001 | 0.701 |
| 9 | 3 | 0.040 | 0.001 | 0.701 |
| 10 | 22 | 0.045 | 0.050 | 0.701 |
| 11 | 3 | 0.025 | 0.001 | 0.701 |
| 12 | 4 | 0.035 | 0.001 | 0.701 |
| 13 | 3 | 0.030 | 0.001 | 0.701 |
| 14 | 4 | 0.005 | 0.001 | 0.701 |
| 15 | 4 | 0.040 | 0.001 | 0.701 |
| 16 | 5 | 0.035 | 0.001 | 0.701 |
| 17 | 3 | 0.035 | 0.001 | 0.701 |
| 18 | 5 | 0.005 | 0.001 | 0.701 |
| 19 | 3 | 0.045 | 0.001 | 0.701 |
| 20 | 5 | 0.025 | 0.001 | 0.701 |

# **Appendix 3**

**Evaluation of the ratings prediction accuracy of the NMF algorithm**

|  |  |  |
| --- | --- | --- |
| Iteration | Number of factors | MAE |
| 0 | 2 | 0.862 |
| 1 | 3 | 0.869 |
| 2 | 4 | 0.916 |
| 3 | 5 | 0.934 |
| 4 | 36 | 0.941 |
| 5 | 6 | 0.950 |
| 6 | 28 | 0.957 |
| 7 | 7 | 0.963 |
| 8 | 39 | 0.965 |
| 9 | 8 | 0.974 |
| 10 | 31 | 0.979 |
| 11 | 9 | 0.990 |
| 12 | 33 | 0.991 |
| 13 | 29 | 0.995 |
| 14 | 38 | 0.997 |
| 15 | 10 | 0.999 |

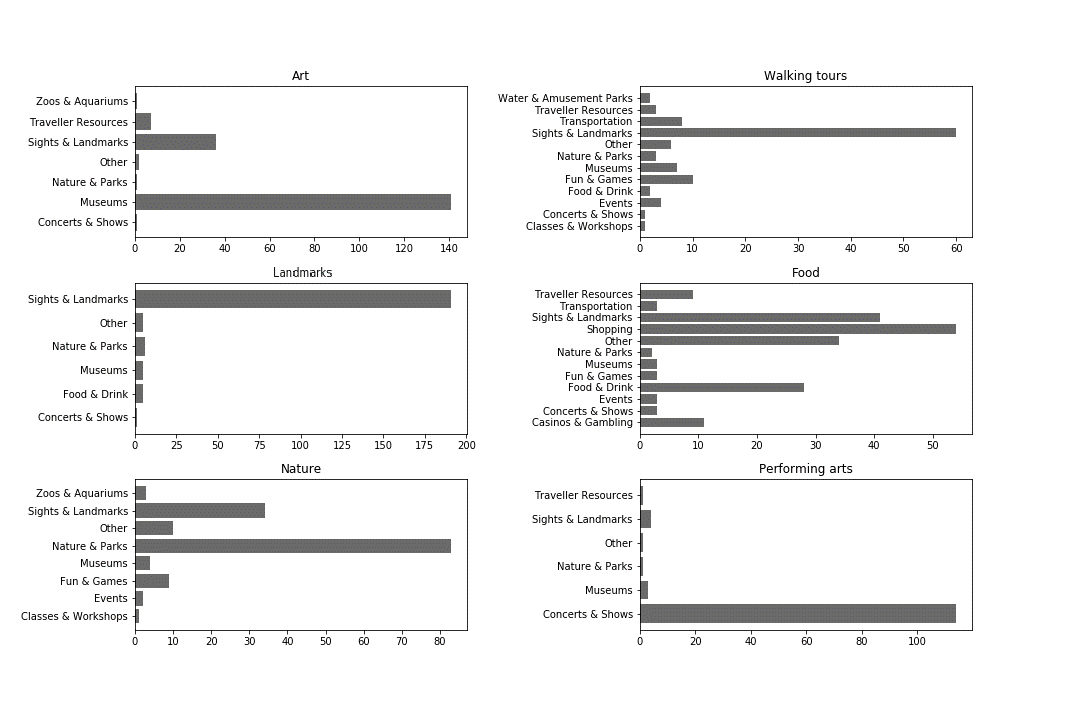
# **Appendix 4**

**Distribution of the LDA topics across TripAdvisor categories**

|  |  |  |
| --- | --- | --- |
| Topic | Category | Number of attractions |
| Art | Concerts & Shows | 1 |
| Art | Museums | 141 |
| Art | Nature & Parks | 1 |
| Art | Other | 2 |
| Art | Sights & Landmarks | 36 |
| Art | Traveller Resources | 7 |
| Art | Zoos & Aquariums | 1 |
| Food | Casinos & Gambling | 11 |
| Food | Concerts & Shows | 3 |
| Food | Events | 3 |
| Food | Food & Drink | 28 |
| Food | Fun & Games | 3 |
| Food | Museums | 3 |
| Food | Nature & Parks | 2 |
| Food | Other | 34 |
| Food | Shopping | 54 |
| Food | Sights & Landmarks | 41 |
| Food | Transportation | 3 |
| Food | Traveller Resources | 9 |
| Nature | Classes & Workshops | 1 |
| Nature | Events | 2 |
| Nature | Fun & Games | 9 |
| Nature | Museums | 4 |
| Nature | Nature & Parks | 83 |
| Nature | Other | 10 |
| Nature | Sights & Landmarks | 34 |
| Nature | Zoos & Aquariums | 3 |
| Performing arts | Concerts & Shows | 114 |
| Performing arts | Museums | 3 |
| Performing arts | Nature & Parks | 1 |
| Performing arts | Other | 1 |
| Performing arts | Sights & Landmarks | 4 |
| Performing arts | Traveller Resources | 1 |
| Landmarks | Concerts & Shows | 1 |
| Landmarks | Food & Drink | 5 |
| Landmarks | Museums | 5 |
| Landmarks | Nature & Parks | 6 |
| Landmarks | Other | 5 |
| Landmarks | Sights & Landmarks | 191 |
| Tours | Classes & Workshops | 1 |
| Tours | Concerts & Shows | 1 |
| Tours | Events | 4 |
| Tours | Food & Drink | 2 |
| Tours | Fun & Games | 10 |
| Tours | Museums | 7 |
| Tours | Nature & Parks | 3 |
| Tours | Other | 6 |
| Tours | Sights & Landmarks | 60 |
| Tours | Transportation | 8 |
| Tours | Traveller Resources | 3 |
| Tours | Water & Amusement Parks | 2 |

# **Appendix 5**

**Representation of TripAdvisor categories within the discovered topics**



# **Appendix 6**

**Numbers of tourist attractions according to different categories**

|  |  |
| --- | --- |
| Category | Number of attractions |
| Sights & Landmarks | 366 |
| Museums | 163 |
| Concerts & Shows | 120 |
| Nature & Parks | 96 |
| Other | 58 |
| Shopping | 54 |
| Food & Drink | 35 |
| Fun & Games | 22 |
| Traveller Resources | 20 |
| Transportation | 11 |
| Casinos & Gambling | 11 |
| Events | 9 |
| Zoos & Aquariums | 4 |
| Classes & Workshops | 2 |
| Water & Amusement Parks | 2 |

# **Appendix 7**

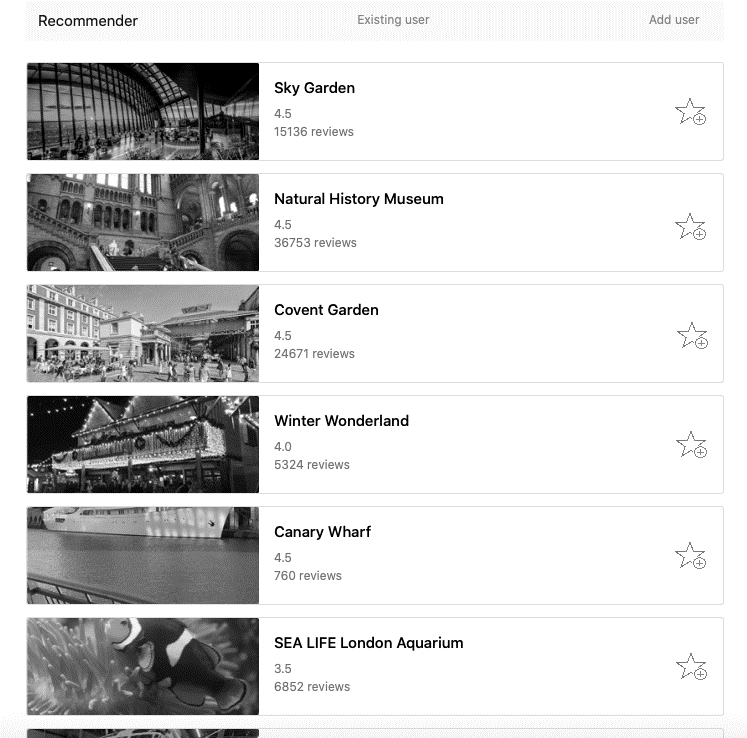
**First screen of the recommender system app**

**A screenshot of a cell phone

Description automatically generated**

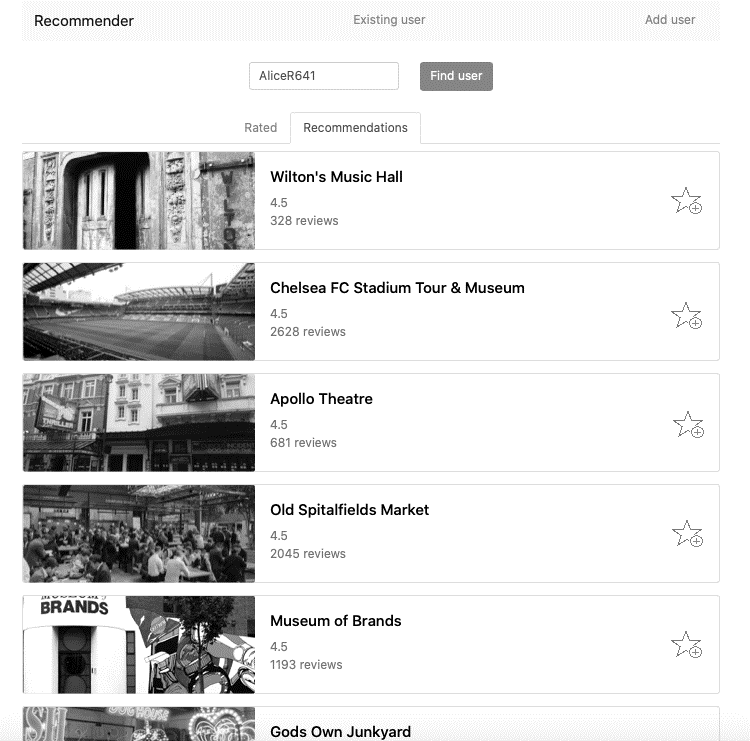
# **Appendix 8**

**Content-based filtering recommender output for a combination of the “Nature” and “Food” categories**

****

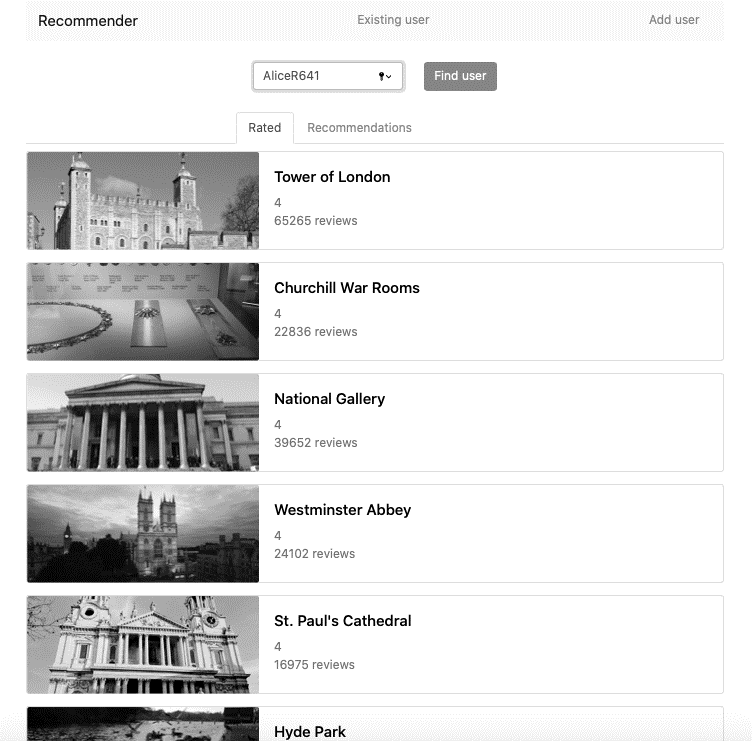
# **Appendix 9**

**FunkSVD recommendation output screen**

****

# **Appendix 10**

**Screen showing items rated by a particular existing user named**

****