A Concise Survey on Content Recommendations

Mehdi Srifi[1(✉)], Badr Ait Hammou[1], Ayoub Ait Lahcen[1,2], and Salma Mouline[1]

[1] LRIT, Associated Unit to CNRST (URAC29), Faculty of Sciences, Mohammed V University, Rabat, Morocco
mehdisrifii@gmail.com

[2] LGS, National School of Applied Sciences, Ibn Tofail University, Kenitra, Morocco

Abstract. A recommender system is often perceived as an enigmatic entity that seems to guess our thoughts, and predict our interests. It is defined as a system capable of providing information to users according to their needs. It is enable them to explore data more effectively. There are several recommendation approaches and this domain remains to date an active research area that aims improving the quality of recommended contents. The main goal of this paper is to provide not only a global view of major recommender systems but also comparisons according to different specifications. We categorize and discuss their main features, advantages, limits and usages.

Keywords: Recommender systems · Content recommendation
Collaborative filtering · Survey

1 Introduction

Recommender systems are powerful tools widely deployed to cope with the information overload problem. These systems are used to suggest relevant items to targeted users based on their past preferences [1].

Currently, the effectiveness of recommender systems has been demonstrated by their use in several domains, such as E-commerce [2], E-learning [3], News [5], Search engines [6], Web pages [7], and so on.

In the literature, several methods have been proposed for building recommender systems, which are based on either the content-based or collaborative filtering approach [8]. However, in order to improve the performance of recommender systems, these two approaches can be combined to define the so-called hybrid recommendation approach. The implementation of the hybrid approach requires a lot of effort in parameterization [9]. In recent years, several recommendation approaches based user reviews have been developed [10], which aim to solve the sparsity and cold start problems by incorporating textual information generated by users (i.e. reviews).

Y. Tabii et al. (Eds.): BDCA 2018, CCIS 872, pp. 393–405, 2018.
https://doi.org/10.1007/978-3-319-96292-4_31

The rest of paper is organized as follows: Sect. 2 presents the backgrounds. Section 3 describes the different recommendation approaches based on the traditional sources of information: ratings, item data, demographic-data and knowledge-data. Section 4 describes the content recommendation approaches. Section 5 presents the evaluation metrics. Finally, Sect. 6 concludes the paper.

2 Backgrounds

In order to recommend interesting items to targeted users, recommender systems collect and process the useful information about the users and items [11].

2.1 Item Profiles

In the personalized recommendation, the item profile is intimately linked to the recommendation technique used, that is to say according to whether or not the content of the item is taken into account in the recommendation process [1,11,12]:

- In the case of a technique that does not take into account the content of the item, the latter can be represented by a simple identifier to distinguish it in a unique way.

- In the opposite case, the latter can be described according to three representations: structured, unstructured or semi-structured, for these last two representations, a step of pre-processing of text, which is the indexing, becomes necessary, in order to transform this text into a structured representation.

2.2 User Profiles

The main purpose of the personalized recommendation is to provide the user with items that meet his needs [11]. To do this, the recommendation system exploits the user's interactions with the e-service, in order to build him a specific profile, modeling his preferences [13–15].

Explicit Feedback. In this method, the user is involved in the process of collecting data about him. The recommender system prompts the user to fill out forms, or to note items, in order to directly specify his preferences to the system. The information provided by the user can take several forms, namely [11]:

Numeric : defined on a scale generally from 1 to 5.

Binary : the user must specify if the item is "good" or "bad".

Ordinal : the user chooses from among a list of terms the one describing the best his feeling with respect to the item in question.

Descriptive : Also called reviews, they represent the textual comments left by users on items. Their exploitation can make it possible to know the preferences of a user in a more refined way. There are many types of review elements [10], such as the contextual information, the multi-faceted nature of opinions, comparative opinions, discussed topics, and reviewers' emotions. Furthermore, several methods for their extraction are described in [10].

Implicit Feedback. In this method, the user is not involved in the process of collecting data about him [13]. This type of method uses the appropriate analysis of the user's history, thus informing about the frequency of consultation of the item, based on the number of visits or only the number of clicks on the corresponding page at item [15]. Other criteria can also be taken into account, including the time spent on the page in question, the list of favorite sites of the user, its downloads, its backups of pages, etc.

Hybrid Feedback. In this method, a combination of the two feedbacks (implicit and explicit) is made [16], in order to be able to fill the gaps of each of them, in terms of lack of information about the user. To do this, it is possible to use the implicit data as check on explicit data provided by the user, in order to understand well his behavior towards the system.

3 Standard Recommendation Approaches

There is a wide variety of recommendation approaches presented in the literature [8]. In this section, we present the most used approaches, with their advantages and limitations [17].

Content-Based Recommendation Approach. The content-based approach directs the user into his decision-making process by suggesting him, items that are close to the content of items he has appreciated in the past [19]. Indeed, it consists of matching the attributes of a given item with the attributes of the user profile (the ideal item). To do this, this approach is based on the representation of items by a profile in the form of a vector of terms obtained from either the item's textual description, keywords, or meta-data. A weighting strategy, such as the Term Frequency/Inverse Document Frequency (TF-IDF) measure, can be used to determine each term's representativeness [18]:

$$W_{i,d} = TF_{i,j} \times IDF_i = \frac{f_{i,d}}{\sum(f_{i,d})} \times \log(\frac{N}{n_i}) \quad (1)$$

where N is the number of documents, n_i is how many times term i is appears in the documents, and $f_{i,d}$ is the number of times term i is appears in the document d.

The content-based approach then tries to recommend the most similar items to the user profile (ideal item) by using for example, the Cosine similarity measure described as follows:

$$sim(item_1, item_2) = \frac{\overrightarrow{item_1}.\overrightarrow{item_2}}{|\overrightarrow{item_1}| * |\overrightarrow{item_2}|} \quad (2)$$

There are other methods derived from the machine learning domain, such as the Bayesian classifier, neural networks, decision trees [18]. These methods can also be used to measure the similarity between profiles of items and users [18,20].

The content-based approach has advantages, each user in such an approach is independent of others, only his behavior affects his profile [19]. Moreover, this approach is able to recommend newly items introduced in the system, even before they are evaluated by users (item cold-start problem) [21]. However, this approach has limitations, namely, the complexity of the representation of the items [11], which must be described in a manner that is both automatic and well structured. Another problem is the limitation of the user to recommendation of similar items to those appreciated in the past [13], which prevents him from discovering new items that may interest him (serendipity). In addition, for a new user, who has not yet sufficiently interacted with the e-service, the system can not develop him its own profile (user cold-start problem) [22].

Collaborative Filtering Approach. The collaborative filtering approach attempts to orient the user in his process of choice by recommending him items that other users with similar tastes have appreciated in the past [23]. The main goal of collaborative filtering systems is thus to guess the user-item connections of the rating matrix [15]. Two main axes stand out in the literature [8]. The first axis is relative to the memory-based approaches that act only on user-item rating matrix, and usually use similarity metrics to obtain the distance between users, or items [24]. The second axis concerns the model-based approaches, which use the machine learning methods, to generate the recommendations. The most used models are Bayesian classifiers, neural networks, matrix factorization, genetic algorithms, among others [8,16,25]. The model-based approaches yield better results, but their implementation cost is higher than that of memory-based approaches [21].

• Item-based collaborative filtering approach: The item-based approach aims to search for items that are neighbors, those who have been appreciated by the same users [21]. To do this, the k-nearest neighbor algorithm (K-NN) can be used to determine the k items closest to the target item, for which the Cosine similarity [16], can be applied to identify the similarity, between two items i and j.

$$sim(i,j) = \frac{\sum_{u \in U_{i,j}} r_{u,i} \times r_{u,j}}{\sqrt{\sum_{u \in U_{i,j}} r_{u,i}^2} \cdot \sqrt{\sum_{u \in U_{i,j}} r_{u,j}^2}} \tag{3}$$

Where $r_{u,i}$ and $r_{u,j}$ are the user's notes u for item i and j respectively. After that, the prediction of the note that the user u will assign to item i is calculated as follows:

$$P_{u,i} = \frac{\sum_{i \in I_u} sim(i,j) r_{u,j}}{\sum_{i \in I_u} |sim(i,j)|} \tag{4}$$

Items with the highest predicted ratings are then recommended to the user.

• User-based collaborative filtering approach: The principle of this technique is that users who have shared the same interest in the past are likely to share in a similar way their future affinities [22]. The k-NN algorithm can be used to select the k-nearest neighbors of the target user, based on the Pearson similarity measure [26], to determine the similarity between two users u and v.

$$sim(u,v) = \frac{\sum_{i \in I_{u,v}} (r_{u,i} - \bar{r_u}).(r_{v,i} - \bar{r_v})}{\sqrt{\sum_{i \in I_{u,v}} (r_{u,i} - \bar{r_u})^2}.\sqrt{\sum_{i \in I_{u,v}} (r_{v,i} - \bar{r_v})^2}} \tag{5}$$

Where $r_{u,i}$ and $r_{v,i}$ are the users's notes u and v for the item i. $\bar{r_u}$ and $\bar{r_v}_u$ are the averages rating of the user u and v respectively. After that, the user's note prediction u for an item i, is done as follows:

$$P_{u,i} = \bar{r}_u + \frac{\sum_{v \in Neighbor(u)} (r_{v,i} - \bar{r_v}).sim(u,v)}{\sum_{v \in Neighbor(u)} |sim(u,v)|} \tag{6}$$

Items with the highest predicted ratings are then recommended to the user.

In contrast to content-based approaches, in this two collaborative filtering approaches mentioned above, the item can be represented only by a simple identifier [11]. This avoids the system to go through the analysis phase of the contents of the items, which can sometimes lead to bad recommendations [13]. Thus, by using these approaches and thanks to their independence of the content, various types of items can be recommended to the user on the same e-service (diversity) [17]. In addition, this kind of approaches makes possible the effect of surprise to the user, by offering him items totally different from items previously appreciated [21]. However, these approaches have limitations [25], namely, the need to have a database containing a large number of user interactions with the e-service, in order to be able to generate recommendations. Thus, these approaches are limited to short-lived items such as news, products containing promotions because this type of items appears and disappears before having a sufficient number of ratings by users of the system [21].

• Matrix Factorization: Matrix factorization models aim to put in a latent factorial space of dimension f, the profiles of users and products directly deduced from the rating matrix [27]. Thus, a note $P_{u,i}$ is predicted by performing the dot product between the latent profiles q_i of the item i and the latent profiles p_u of the user u: $P_{u,i} = q_i^T p_u$.

Several matrix factorization techniques exist [18], namely, the SVD (Singular Value Decomposition), PCA (principal Component Analysis) and (NMF) (Non-negative Matrix Factorization) models that are used to identify latent factors from explicit users feedback. Another enhancement to basic SVD model is SVD++ [18]. This asymmetric variation enables adding implicit feedback which in turn allows to improve the precision of the predictions of the SVD.

In recent years, matrix factorization models are becoming more efficient [27], thanks to consideration of various factors such as social links [28], text or time [29], allowing a better tracking of user behavior. Matrix factorization techniques give better precisions in the prediction than the recommendation approaches based on the neighborhood mentioned above [18,28,30]. In addition, they offer an efficient model in terms of memory, thus, easy to learn by the systems [31].

Demographic Recommendation Approach. The principle on which this approach is based, is that users who have common demographic-attributes (gender, age, city, job, etc) will necessarily also have common trends in the future [8,24]. Several works [32–34] have shown that the exploitation of demographic data instead of the user evaluation history, solves the problem of cold start of the user. However, this approach does not always provide users with recommendations that meet their needs in a precis way, because it does not take into account their preferences [21].

Knowledge-Based Recommendation Approach. This technique is based on a set of knowledge that defines the user's preference domain [15]. In the literature, this type of approach is sometimes considered to belong to the same family of content-based approach [35]. The only difference is that in the knowledge-based approach, the user explicitly specifies criteria for the recommendation system, that define conditions on items of interest [18], unlike the content-based recommendation approach that relies only on the user's history. Therefore, the knowledge-based approach takes as input: the user's specifications, item attributes, and the domain of knowledge (domain-specific rules, similarity metrics, utility functions, constraints). The use of this approach becomes useful, in the case of items rarely sold and therefore rarely noted as for example, very expensive products [18]. Recommendation systems based on this approach can be classified into two classes: *Constraint-based recommender systems*, which takes as input, the user-defined constraints on the attributes(eg: min or max limits...) of the item [36]. *Case-based recommender systems*, in which, the recommendation is made by calculating the similarity between the attributes of the items and the cases specified by the user [37].

Hybrid Recommendation Approach. Hybrid approaches are techniques that combine two or more different recommendation techniques [9,15], in order to overcome the limitations posed by each of them. For instance, several works [38–40] have shown that the use of an hybrid recommendation approach can solves the users/items cold-start problem encountered when using an individual recommendation approach. However, the implementation of hybrid approaches requires a lot of effort in parameterization allowing the combination between different approaches [9], so the process of explaining these recommendations to users becomes difficult [41].

4 Content Recommendation Approaches

4.1 Preference-Based Product Ranking

The preference-based product ranking approach, becomes useful when the items are described by a set of attributes, for example, for a movie (Producer, actors, genre) [25]. In this approach, the user's preference can be represented by ($\{V_1, \ldots, V_n\}$, $\{w_1, \ldots, w_n\}$), where V_i is the value function (criterion) that a user specifies for the attribute a_i [25], and w_i is the relative importance (i.e., the weight) of a_i. Then, the utility of each product a_i is calculated, using the multi-attribute utility (MAUT) as follows:

$$U(< a_1, a_2, \ldots, a_n >) = \sum_{i=0}^{n} w_i \times V_i(a_i) \tag{7}$$

Products with large utility values, are classified and then recommended to the user.

Based on the utility of each item characteristic for the user in question, this approach allows to filter items, in a finer and more tailored way, than other classical recommendation approaches [18]. However, the major challenge of this technique is in defining the most appropriate utility function for the user at hand [25].

4.2 Exploiting Terms on Reviews for Recommender Systems

In [42] the authors presented an approach called index-based approach, in which, each user is characterized by the textual content of his reviews. The term-based user profile $\{t_1, \ldots, t_n\}$ is constructed by extracting keywords from user reviews, followed by assignment of a weight $U_{i,j}$ to each extracted term, by using TF-IDF technique. This weight indicates how important each term is to the user. Similarly, each item is represented by a set of terms extracted from the reviews published on this item P_i. During the recommendation process, the user's profile serves as a query to retrieve items that are most similar to the user profile. The index-based approach has been evaluated [42] using a dataset collected from Flixster. The evaluation shows that this approach outperforms the user/item based collaborative filtering approaches, in terms of diversity, coverage, and novelty, but its accuracy is lower than that of user/item based collaborative filtering approaches.

4.3 Exploiting Emotions on Reviews for Recommender Systems

In [43], a new recommendation approach has been proposed, with the aim of improving the results of standard collaborative filtering approaches, by exploiting the emotions left by these users in reviews relating to given items. The principle of this approach is the following: given the user-item rating matrix R and emotion E towards others' reviews, the goal is to deduce the missing values in R.

To do this, the proposed approach (Mirror framework) aims to minimize the following equation [43]:

$$\begin{aligned} &\underset{U,V}{min} ||\widetilde{W} \odot (R - U^T V)||_F^2 + \alpha(||U||_F^2 + ||V||_F^2) \\ &+ \gamma min \sum_{i=1}^{n} \sum_{j=1}^{m} max(0, (u_i^T v_j - \bar{R}_{*j}^{ip})^2 - (u_i^T v_j - \bar{R}_{*j}^{in})^2) \end{aligned} \tag{8}$$

where U denotes the preference latent factors of each user u_i, and V denotes the characteristic latent factors of each item v_j. $\widetilde{W}$ is function that controls the importance of $R_{i,j}$. The term $\alpha(||U||_F^2+||V||_F^2)$ is introduced to avoid over fitting. γ is introduced to control its local contribution of emotion regularization to model emotion on other users' reviews. $\bar{R}_{*j}^{ip}$ and $\bar{R}_{*j}^{ip}$, are denoted as the average rating of positive and negative emotion reviews from u_i to v_j, respectively.

The results of experience and comparison [43] of this approach with standard approaches [44,45], show that when training sets (Ciao, Epinions) are more sparse, this approach allows to provide more precise recommendations than those returned by the standard approches. Thus its performance decreases more slowly, when cold-start users are involved in both training sets.

4.4 Exploiting Contexts on Reviews for Recommender Systems

Starting from the following idea: "the utility of choosing an item may vary according to the context", the authors of [46] have defined the utility of an item for the user, by two factors, namely, the *predictedRating*, calculated using standard item-based collaborative filtering algorithm, and the *contextScore*, measuring the convenience of an item i to the target user u's current context. The context is mined from a textual description of user's current situation and the features that are important to him. The utility score of item i for user u is calculated as:

$$utility(u,i) = \alpha \times predictedRating(u,i) + (1-\alpha) \times contextScore(u,i) \quad (9)$$

where α is a constant, representing the weight of the predicted rating. Products with large utility values, are classified and then recommended to the user.

The results of the tests performed by the authors in [46] on a data set (hotels on TripAdvisor), show that this approach gives better predictions than the standard non-context based rating prediction using the item-based collaborative filtering algorithm. In [47] another approach was developed, which associate the latent factors with the contextual information inferred from reviews, to enhance the standard latent factor model.

4.5 Exploiting Topics on Reviews for Recommender Systems

In [48], the authors proposed an approach in which each user is assigned a profile of preferences grouping the topics (aspects of the item, for example: the location of the hotel, the cleanliness, the view of the room, etc.) mentioned by the user in his reviews, and having a large number of opinions (exceeding a certain threshold ts). More precisely, the profile of the user is represented by $Z_i = \{$z| *count*(z, R_i)>ts$\}$, where *count*(z, R_i) indicates the number of opinions associated with the aspect *z* in the set of reviews R_i written by the user *i*, and *ts* is a threshold defined as zero in their experience. Thus, the relevance of a review $r_{j,A}$ belonging to the set of reviews R_A associated with a product candidate A(j $\in$ 1, ..., | R_A |), is defined by $Z_{i,r_{j,A}}$, which consists of aspects appearing both in the user's profile Z_i and in the review $r_{j,A}$. Finally, the interest of an item for the user is calculated by weighting the average of the already existing ratings of this item by $Z_{i,r_{j,A}}$.

The results of the experiments [48] of this technique on a set of data collected from TripAdvisor, showed that this technique surpasses the non-personalized technique of product classification, with regard to the Mean Absolute Error (MAE) as well as Kendall's tau, which measures the fraction of items with the same order in the classification provided by the system and the one wanted by the user [49].

5 Evaluation Metrics for Recommendation Approaches

There are several criteria for evaluating recommendation approaches, the most important of which are [8,15,16]:

Statistical Accuracy Metrics. Its principle is based on the fact of verifying if the predicted scores for the user with respect to given items are correct [8], to do this two measurements have been reported namely the Mean Absolute Error (MAE) and the Root Mean Square Error (RMSE). Let $p_{u,j}$ a user note prediction u for item i and $n_{u,j}$ the actual note assigned by the user u for the item i:

MAE: measure the difference between predicted and true notes, small values of MAE means that the recommendation system accurately predicts the ratings. It is calculated as follows:

$$MAE = \frac{1}{N} \sum_{u,j} |p_{u,j} - n_{u,j}| \tag{10}$$

RMSE: puts more importance on larger absolute error. The recommendation is more accuracy when the RMSE is smaller. It is calculated via:

$$RMSE = \sqrt{\frac{1}{N} \sum_{u,j} (p_{u,j} - n_{u,j})^2} \tag{11}$$

Decision Support Accuracy Metrics. These measures allow users to find the items that interest them most, among all those available [18]. Several measures exist [16], namely, Weighted errors, Reversal rate, Precision Recall Curve (PRC), Receiver Operating Characteristics (ROC) and Precision, Recall and F-measure. The most used are Precision, Recall and F-measure.

Precision: the precision determines among the set of recommended items those who are the most relevant, its calculated via:

$$Precision = \frac{Correctly\ recommended\ items}{Total\ recommended\ items} \tag{12}$$

Recall: the Recall determines the proportion of recommended items among all relevant items, its calculated as follows:

$$Recall = \frac{Correctly\ recommended\ items}{Total\ useful\ recommended\ items} \tag{13}$$

F-measure: another way exists making the computation much simpler and easier [16], it is the F-measure which groups the two previous metrics into one, it is defined as follows:

$$F-measure = \frac{2PrecisionRecall}{Precision + Recall} \tag{14}$$

Coverage. It consists in determining the proportion of users for whom the recommender system can actually recommend items, as well as the proportion of items that can be recommended by this system [18].

Novelty, Diversity and Serendipity. Anothers measures [8,25] can be taken into consideration as, the novelty criterion which represents a very important aspect in the recommendation process especially if this element has not been seen before. Another important criterion is diversity, the absence of this criterion can generate a feeling of boredom in the user who is sentenced to receive similar items. In addition, the criterion of serendipity, it brings a surprise effect it can recommend users unexpected and surprising items.

6 Conclusion

The recommendation systems present tools for personalization and filtering of the information sought by the user. Several approaches on which these systems are based, exist in the literature, the best known of which are content-based recommendation approaches and collaborative filtering approaches presenting the problem of sparsity and cold start. The hybrid approach remains however an alternative trying to merge the advantages of these methods to fill their weak points. Recently, new approaches have been developed to fill the gaps in standard approaches. These new approaches in turn have some limitations, which presupposes the possibility of intervention by the researchers' community in order to reinforce and develop other approaches likely to adequately meet users' expectations. Thus, the present work can serve as a platform for exploring and developing new methods that can bridge the gaps in the presented approaches.

References

1. Cliquet, G.: Innovation method in the Web 2.0 era. Dissertation, Arts et Métiers ParisTech (2010)
2. Linden, G., Smith, B., York, J.: Amazon.com recommendations: item-to-item collaborative filtering. IEEE Internet Comput. **7**(1), 76–80 (2003)

3. Bobadilla, J.E.S.U.S., Serradilla, F., Hernando, A.: Collaborative filtering adapted to recommender systems of e-learning. Knowl.-Based Syst. **22**(4), 261–265 (2009)
4. Miller, B.N., et al.: MovieLens unplugged: experiences with an occasionally connected recommender system. In: Proceedings of the 8th International Conference on Intelligent User Interfaces. ACM (2003)
5. Billsus, D., et al.: Adaptive interfaces for ubiquitous web access. Commun. ACM **45**(5), 34–38 (2002)
6. Pass, G., Chowdhury, A., Torgeson, C.: A picture of search. In: InfoScale, vol. 152 (2006)
7. McNally, K., et al.: A case study of collaboration and reputation in social web search. ACM Trans. Intell. Syst. Technol. (TIST) **3**(1), 4 (2011)
8. Bobadilla, J., et al.: Recommender systems survey. Knowl.-Based Syst. **46**, 109–132 (2013)
9. Burke, R.: Hybrid recommender systems: survey and experiments. User Model. User-Adap. Interac. **12**(4), 331–370 (2002)
10. Chen, L., Chen, G., Wang, F.: Recommender systems based on user reviews: the state of the art. User Model. User-Adap. Interac. **25**(2), 99–154 (2015)
11. Ben Ticha, S.: Hybrid personalized recommendation. Dissertation, Université de Lorraine (2015)
12. Goldberg, D., et al.: Using collaborative filtering to weave an information tapestry. Commun. ACM **35**(12), 61–70 (1992)
13. Wei, C.-P., Shaw, M.J., Easley, R.F.: Recommendation systems in electronic commerce. In: E-Service: New Directions in Theory and Practice, p. 168 (2002)
14. Burke, R.: Hybrid web recommender systems. In: Brusilovsky, P., Kobsa, A., Nejdl, W. (eds.) The Adaptive Web. LNCS, vol. 4321, pp. 377–408. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-72079-9_12
15. Lemdani, R.: Hybrid adaptation system in recommendation systems. Dissertation, Paris Saclay (2016)
16. Isinkaye, F.O., Folajimi, Y.O., Ojokoh, B.A.: Recommendation systems: principles, methods and evaluation. Egypt. Inf. J. **16**(3), 261–273 (2015)
17. Sharma, M., Mann, S.: A survey of recommender systems: approaches and limitations. Int. J. Innov. Eng. Technol. **2**(2), 8–14 (2013)
18. Ricci, F., Rokach, L., Shapira, B.: Introduction to recommender systems handbook. In: Ricci, F., Rokach, L., Shapira, B., Kantor, P.B. (eds.) Recommender Systems Handbook, pp. 1–35. Springer, Boston, MA (2011). https://doi.org/10.1007/978-0-387-85820-3_1
19. Louëdec, J.: Bandit strategies for recommender systems. Dissertation, University Paul Sabatier-Toulouse III (2016)
20. Schafer, J.B., Konstan, J.A., Riedl, J.: E-commerce recommendation applications. Data Min. Knowl. Discov. **5**(1–2), 115–153 (2001)
21. Quba, R.C.A.: On enhancing recommender systems by utilizing general social networks combined with users goals and contextual awareness. Dissertation, Université Claude Bernard-Lyon I (2015)
22. Adomavicius, G., Tuzhilin, A.: Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. IEEE Trans. Knowl. Data Eng. **17**(6), 734–749 (2005)
23. Lousame, F.P., Sánchez, E.: A taxonomy of collaborative-based recommender systems. In: Castellano, G., Jain, L.C., Fanelli, A.M. (eds.) Web Personalization in Intelligent Environments, pp. 81–117. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-02794-9_5

24. Croft, W.B., Metzler, D., Strohman, T.: Search Engines: Information Retrieval in Practice. Addison-Wesley, Reading (2010)
25. Aggarwal, C.C.: Recommender Systems. Springer, Heidelberg (2016). https://doi.org/10.1007/978-3-319-29659-3
26. Zhang, F., et al.: Fast algorithms to evaluate collaborative filtering recommender systems. Knowl.-Based Syst. **96**, 96–103 (2016)
27. Dias, C.E., Guigue, V., Gallinari, P.: Recommendation and analysis of feelings in a latent textual space. In: CORIA-CIFED (2016)
28. Hammou, B.A., Lahcen, A.A.: FRAIPA: a fast recommendation approach with improved prediction accuracy. Expert Syst. Appl. **87**, 90–97 (2017)
29. Dias, C.-E., Guigue, V., Gallinari, P.: Recommendation and analysis of feelings in a latent textual space, Sorbonne University, UPMC Paris univ 06, UMR 7606, LIP6, F-75005 (2016)
30. Hammou, B.A., Lahcen, A.A., Aboutajdine, D.: A new recommendation algorithm for reducing dimensionality and improving accuracy. In: 2016 IEEE/ACS 13th International Conference of Computer Systems and Applications (AICCSA). IEEE (2016)
31. Koren, Y., Bell, R., Volinsky, C.: Matrix factorization techniques for recommender systems. Computer **42**(8), 43–47 (2009)
32. Safoury, L., Salah, A.: Exploiting user demographic attributes for solving cold-start problem in recommender system. Lect. Notes Softw. Eng. **1**(3), 303 (2013)
33. Wang, Y., Chan, S.C.-F., Ngai, G.: Applicability of demographic recommender system to tourist attractions: a case study on trip advisor. In: Proceedings of the The 2012 IEEE/WIC/ACM International Joint Conferences on Web Intelligence and Intelligent Agent Technology, vol. 03. IEEE Computer Society (2012)
34. Sun, M., Li, C., Zha, H.: Inferring private demographics of new users in recommender systems. In: Proceedings of the 20th ACM International Conference on Modelling, Analysis and Simulation of Wireless and Mobile Systems. ACM (2017)
35. Smyth, B.: Case-based recommendation. In: Brusilovsky, P., Kobsa, A., Nejdl, W. (eds.) The Adaptive Web. LNCS, vol. 4321, pp. 342–376. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-72079-9_11
36. Felfernig, A., Burke, R.: Constraint-based recommender systems: technologies and research issues. In: Proceedings of the 10th International Conference on Electronic Commerce. ACM (2008)
37. Bridge, D., et al.: Case-based recommender systems. Knowl. Eng. Rev. **20**(3), 315–320 (2005)
38. De Pessemier, T., Vanhecke, K., Martens, L.: A scalable, high-performance algorithm for hybrid job recommendations. In: Proceedings of the Recommender Systems Challenge. ACM (2016)
39. Strub, F., Gaudel, R., Mary, J.: Hybrid recommender system based on autoencoders. In: Proceedings of the 1st Workshop on Deep Learning for Recommender Systems. ACM (2016)
40. Braunhofer, M., Codina, V., Ricci, F.: Switching hybrid for cold-starting context-aware recommender systems. In: Proceedings of the 8th ACM Conference on Recommender systems. ACM (2014)
41. Kouki, P., et al.: User preferences for hybrid explanations. In: Proceedings of the Eleventh ACM Conference on Recommender Systems. ACM (2017)
42. Esparza, S.G., O'Mahony, M.P., Smyth, B.: Effective product recommendation using the real-time web. In: Bramer, M., Petridis, M., Hopgood, A. (eds.) Research and Development in Intelligent Systems XXVII, pp. 5–18. Springer, London (2011). https://doi.org/10.1007/978-0-85729-130-1_1

43. Meng, X., et al.: Exploiting emotion on reviews for recommender systems. AAAI (2018)
44. Zhang, S., et al.: Learning from incomplete ratings using non-negative matrix factorization. In: Proceedings of the 2006 SIAM International Conference on Data Mining. Society for Industrial and Applied Mathematics (2006)
45. Raghavan, S., Gunasekar, S., Ghosh, J.: Review quality aware collaborative filtering. In: Proceedings of the Sixth ACM Conference on Recommender Systems. ACM (2012)
46. Hariri, N., et al.: Context-aware recommendation based on review mining. In: Proceedings of the 9th Workshop on Intelligent Techniques for Web Personalization and Recommender Systems (ITWP 2011) (2011)
47. Li, Y., et al.: Contextual recommendation based on text mining. In: Proceedings of the 23rd International Conference on Computational Linguistics: Posters. Association for Computational Linguistics (2010)
48. Musat, C-C., Liang, Y., Faltings, B.: Recommendation using textual opinions. In: IJCAI International Joint Conference on Artificial Intelligence, No. EPFL-CONF-197487 (2013)
49. Kendall, M.G.: A new measure of rank correlation. Biometrika **30**(1/2), 81–93 (1938)