



A content-based recommendation approach based on singular value decomposition

Francesco Colace, Dajana Conte, Massimo De Santo, Marco Lombardi, Domenico Santaniello & Carmine Valentino

To cite this article: Francesco Colace, Dajana Conte, Massimo De Santo, Marco Lombardi, Domenico Santaniello & Carmine Valentino (2022) A content-based recommendation approach based on singular value decomposition, Connection Science, 34:1, 2158-2176, DOI: [10.1080/09540091.2022.2106943](https://doi.org/10.1080/09540091.2022.2106943)

To link to this article: <https://doi.org/10.1080/09540091.2022.2106943>



© 2022 The Author(s). Published by Informa UK Limited, trading as Taylor & Francis Group



Published online: 08 Aug 2022.



Submit your article to this journal [↗](#)



Article views: 733



View related articles [↗](#)



View Crossmark data [↗](#)



Citing articles: 1 View citing articles [↗](#)



A content-based recommendation approach based on singular value decomposition

Francesco Colace^a, Dajana Conte^b, Massimo De Santo^a, Marco Lombardi^a,
Domenico Santaniello ^a and Carmine Valentino^{a,b}

^aDIIN, University of Salerno, Fisciano, Italy; ^bDIPMAT, University of Salerno, Fisciano, Italy

ABSTRACT

In the Internet era, where information and communication technologies (ICT) allow data exchange, new tools able to select the correct data are needed. In this field, Recommender Systems have a prime location. The recommendation methods take many forms based on the information exploited in order to provide rating forecasts. However, all of them can choose the correct information to support users. Indeed, these methods allow for overcoming the information overload problem. This paper introduces the theoretical bases of a novel recommendation method defined Rating Singular Value Decomposition (RSVD). RSVD is a Content-Based method that exploits the Singular Value Decomposition properties in order to calculate rating forecasts. This method aims to elaborate the users and items profile to obtain matrices related to ones obtained in Collaborative Filtering methods that exploit Singular Value Decomposition. The accuracy of RSVD is compared with the accuracy of Collaborative Filtering methods, and a study on the sparsity problem is performed. The results obtained are promising.

ARTICLE HISTORY

Received 6 August 2021
Accepted 21 July 2022

KEYWORDS

Recommender system;
content-based approach;
singular value
decomposition; rating
forecast

1. Introduction

In the Internet era, the collection of available data arises the problem of Big Data (Castiglione et al., 2018), which is characterised by volume, velocity, and variety (Allam & Dhunny, 2019). Indeed, during the nineties, the first e-commerce sites had to face the Big Data problem through a tool that could filter and analyse data in order to provide support to users of the system. In this context, Recommender Systems (Quijano-Sánchez et al., 2020; Y. Zhang & Chen, 2020) were developed and, over the years, have owned great relevance. Over the years, Recommender Systems (RSs) have exploited much different information in order to improve the ability to provide suggestions to users. Based on the information exploited and the development, three main categories of RSs can be identified (Sinha & Dhanalakshmi, 2019a, 2019b). The First Generation RSs consist of the Content-Based, Collaborative Filtering, and Hybrid methods (Bobadilla et al., 2013; Ricci et al., 2015). The Second Generation RSs exploit the knowledge related to user preference (Eirinaki et al., 2018) and the context that allows defining Context-Aware Recommender Systems (CARSS) (Casillo,

Colace, et al., 2021; Renjith et al., 2020). CARs (Liu et al., 2021) were largely exploited in the tourism field because of their ability to exploit contextual information (Adomavicius & Tuzhilin, 2015) in order to improve forecasts related to users' preferences in a specific context (Casillo, Gupta, et al., 2022). The Third Generation RSs exploit Ontologies (Renjith et al., 2020; Sinha & Dhanalakshmi, 2019a, 2019b).

The central aim of RSs consists of providing proper support to users (Portugal et al., 2018). The support can be obtained by calculating forecasts that all recommender systems can get based on the exploited information. The problem associated with forecasts is the lack of sufficient information to support all users on all items. This paper will describe a recommendation approach, defined as Rating Singular Value Decomposition (RSVD), that exploits the properties of Singular Value Decomposition in Content-Based RS instead of Collaborative Filtering. In fact, SVD usually is exploited to reduce the problem dimensionality and can be exploited when known ratings are available. Instead, The proposed approach allows transforming users' and items' profiles according to the factorisation to calculate the ratings that are all unknown. The numerical results measure the accuracy of RSVD on the MovieLens100k dataset and testify to the ability of RSVD to provide forecasts when information is lacking.

This paper is organised as follows. Section 2 focuses on the description of Background and Related Works; Section 3 describes the proposed approach and its theoretical basis; Section 4 presents numerical experiments developed on the datasets MovieLens100k and MovieLens10M, and, finally, Section 5 contains conclusions and future works.

2. Background and related works

To filter information, Recommender Systems exploit three principal tools: the user of the system, the item that the system suggests, and the transaction that consists of an interaction between users and the system (Ricci et al., 2015). The most common form of transaction is the rating: an implicit or explicit evaluation of an item by a user. This evaluation can be expressed in various forms: Numerical Rating (the most used, the user chooses from a range of values based on his appreciation), Ordinary Rating (Strongly agree, Agree, Neutral, Disagree, Strongly disagree), Binary Rating (like, dislike) and Unary Rating (the information associated with the interaction between the user and an item is stored) (Ricci et al., 2015). The rating can be defined as a function, defined rating function or utility function and summarised in Equation (1), that has as domain the Cartesian Product of the users set U and the items set I and as codomain, the real numbers set \mathbb{R} .

$$r : (u, i) \in U \times I \mapsto r_{ui} \in \mathbb{R} \quad (1)$$

Because of the lack of knowledge of all values $r_{ui} \forall (u, i) \in U \times I$, Recommender Systems have to provide a rating forecasts \hat{r}_{ui} that estimates the unknown ratings and the efficacy of RSs is based on the ability to provide reasonable approximations. Based on how these forecasts are made and how information relating to users and items is stored and processed, it is possible to classify RSs.

The main typologies of Recommender Systems are Content-Based, Collaborative Filtering, and Hybrid.

Content-Based RSs generate rating forecasts through the feature vectors of the items and the user profile. The structure of an RS based on high-level content is composed of

three phases. In the Content Analyzer phase, the features of the items are analysed to build a vector that describes them (Lops et al., 2011) and afterward, the Profile Learner phase involves the study of user preferences summarised through a numerical vector which constitutes the relative profile (Lops et al., 2011). At last, in the Filtering Component phase, the user-item affinity is generated through similarity measures between the feature vector of the item and the user profile vector.

D. Wang et al. (2018) develop a Content-Based RS that captures information about the abstract of a scientific article to suggest the most appropriate journals or conferences. After choosing the mode of feature acquisition, the Content-Based approach used general rating predictions through softmax regression, which generalises the logistic regression. Casillo, Conte, et al. (2021) propose a Content-Based RS that exploits users and items features in order to provide rating forecasts and uses contextual information to select the adequately calculated forecasts. Casillo, Conte, et al. (2022) exploit social information in order to create users' profiles and the experts in the cultural heritage field to create items profiles. These profiles allow calculating the user-item affinity. Pérez-Almaguer et al. (2021) provide a Content-Based group recommendation method. In particular, they calculate the items and users' profiles through two different approaches. The affinity between the entities is calculated through cosine similarity if item profiles consist of binary elements. If the item profiles have multivalued values they exploit an analysis on each feature. Then, the group recommendation takes advantage of the introduction of a taxonomy. Finally, some learning methods are exploited into the Content-Based field for various applications in healthcare and finance (Javed et al., 2021).

Collaborative Filtering RSs (Koren et al., 2009; Ricci et al., 2015) is based on the ratings that users give to the system about items they interact with and calculate predictions through the opinion of a community (Geuens et al., 2018), which allow recommending items which similar users have evaluate positively (Aberger, 2014; Anastasiu et al., 2016). Collaborative Filtering methods can be divided into two groups: Memory-Based and Model-Based (Casillo, Colace, et al., 2021). Memory-Based RSs exploit three main strategies. The first one consists in creating Neighbourhoods among users (User-Based) (F. Zhang et al., 2016). The similarity can be calculated through the Cosine Vector or the Pearson Correlation (Desrosiers & Karypis, 2011; Jain et al., 2020). The second strategy, instead, creates groups of items (Item-based) through the Pearson Correlation for the items or with the Adjusted Cosine formula (Desrosiers & Karypis, 2011; Jain et al., 2020). Finally, it is possible to combine the two techniques to obtain the Hybrid User-Item Based CF. Model-Based RSs aim to create a numerical model by analysing the rating matrix (Casillo, Colace, et al., 2021; Ricci et al., 2015). To analyse and elaborate the rating matrix, matrix factorisation methods are exploited. Among all of them, the most common is Principal Component Analysis (PCA) (Bokde et al., 2015; Vozalis & Margaritis, 2007), which aims to reduce the dimension of the problem through the analysis of the correlation matrix of data analysed (Kumaravel & Dutta, 2014), Non-Negative Matrix Factorization (NMF) (Bokde et al., 2015; Luo et al., 2014) that aims to factorise the rating matrix in the product of two matrices composed by non-negative elements (Hernando et al., 2016), Probabilistic Matrix Factorization (PMF) (Bokde et al., 2015; Mnih & Salakhutdinov, 2008), which exploits probabilistic theory to reduce the root mean squared error on the training data, and Singular Value Decomposition (SVD) (Golub & Van Loan, 2013; Gu et al., 2020; Symeonidis & Zioupos, 2016).

He et al. (2018) propose a Memory-Based CF of the Item-Based CF type built through neural networks. In particular, relevance is given to the user's history in the theoretical construction. Mnih and Salakhutdinov (2008) exploit probabilistic matrix factorisation, hence a Model-Based RS. The application of the PMF to the Gradient Descent method provides a stable Recommender System with few ratings from the users. Recommendation methods that fall in Model-Based approaches are Average Rating Filling (ARF) (Anastasiu et al., 2016; van der Vlugt, 2018), Stochastic Gradient Descent (SGD) (Anastasiu et al., 2016; R. Wang et al., 2019; Zhao et al., 2019) and Biased Stochastic Gradient Descent (BSGD) (R. Wang et al., 2019; Zhao et al., 2019). Gu et al. (2020) propose a weighted model based on SVD and gradient descent. In their paper, non-negative weights aim to solve the noise effect and reduce the unreliable rating forecasts. Luo et al. (2019) deeply studied eight models that exploit stochastic gradient descent and the effects of the different modification strategies on SGD. Yu et al. (2021) exploit SVD, SGD and Stacked AutoEncoder (G. Zhang et al., 2020) in order to overcome sparsity problem. Bhavana et al. (2019) propose an approach in which Singular Value Decomposition is developed through Graphics Processing Unit (GPU) and CUDA (Compute Unified Device Architecture) (Nickolls et al., 2008). This approach divides the rating matrix into blocks, and the factorisation development is processed in parallel. Nguyen and Zhu (2013) exploit matrix factorisation and item profiles. In particular, they introduce in the factorisation model items profiles that are composed of binary elements. Moreover, the analysis of common attributes for items and the introduction of the tag analysis concept translated to items improve the recommendation method's performance. Rendle (2010) exploits Support Vector Machine to provide rating forecasts and overcome the specialisation of the input data. The principal aim of the paper consists of improving the recommendation method to manage the sparsity problem. In addition, the provided model is linear.

Other typologies of RSs are Demographic Filtering, Knowledge-Based, and Community-Based. Demographic Filtering RS recommends an item to the user by using the information regarding the region to which he belongs, the language, the gender, and the age (Ricci et al., 2015). Knowledge-Base RS suggests an item to the user by considering how much the features of the suggested item meet the user's specific needs. To determine these needs, it is essential to build an appropriate user profile (Ricci et al., 2015). Community-Based RS recommends an item to the user by using the information acquired from social networks. The means of acquisition are the user tags or the preferences of the friends of the latter (Ricci et al., 2015). The social information can be exploited to create profiles (Goripati & Vatsavayi, 2017) or in a Collaborative Filtering RS (Liang et al., 2019). Hence, the Community-Based RS main feature is the exploitation of social media information.

Table 1 summarises the advantages and disadvantages of RSs typologies described in this Section.

3. RSVD: a content-Based approach based on SVD

This section focuses on a novel Content-Based approach that exploits the properties of Singular Value Decomposition. SVD is usually exploited in Collaborative Filtering methods, reducing the requested memory and obtaining rating forecasts. Instead, the proposed approach, defined RSVD, aims to exploit SVD in a Content-Based recommendation method.

Table 1. Summary table of advantages and disadvantages of recommendation methods.

Method	Technique	Advantages	Disadvantages
Content Based		– Not a large number of users is need to forecasting	– Limited content analysis (Tarus et al., 2018)
		– Requires knowledge of item features only	– Over-specialisation
Collaborative Filtering	Memory Based	– Ease of implementation and data addition	– Cold start (new user) (F. Zhang et al., 2016)
		– It adapts well to related items	– Dependence on known ratings
	Model Based	– Serendipity of results (D. Wang et al., 2018)	– Cold Start (new user/new item/new community) (Karabadjji et al., 2018)
		– Work well with sparse matrices	– Limited scalability for large dataset
Hybrid		– Adequately addresses the scalability problem	– Work poorly with sparse matrices
		– Exploits the advantages of different techniques	– High cost for the model construction
Demographic Filtering		– Interacts well with the issues of the exploited techniques	– Increase of complexity
		– No historical data of user transaction (Renjith et al., 2020)	– Difficulty in collecting all the demographic data (Sinha & Dhanalakshmi, 2019a)
Knowledge Based		– Deal well with cold start problem (Renjith et al., 2020)	– Highly computational intensive approach (Renjith et al., 2020)
Community Based		– Exploits social media information (Sinha & Dhanalakshmi, 2019b)	– High cost for the model construction

To underline this difference, this section is divided into two subsections: the first describes how SVD is usually exploited, and the second presents the theoretical basis of RSVD.

3.1. SVD: basis and usage on collaborative filtering RSs

The Singular Value Decomposition allows to factorise a real rectangular matrix $R \in \mathbb{R}^{m \times n}$, where $m, n \in \mathbb{N}$, into the product of three matrices according to Equation (2).

$$R = UDV^T \quad (2)$$

The matrices $U \in \mathbb{R}^{m \times m}$ and $V \in \mathbb{R}^{n \times n}$ are orthogonal and are defined matrices of left and right singular vectors respectively. The matrix $D = \text{diag}(\sigma_1, \dots, \sigma_p) \in \mathbb{R}^{m \times n}$ $p = \min\{m, n\}$ is defined matrix of singular values and the elements $\sigma_1, \dots, \sigma_p$ satisfy Equation (3).

$$\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_k > \sigma_{k+1} = \dots = \sigma_p = 0 \quad (3)$$

In particular, matrix U columns represent the eigenvectors of the square matrix $RR^T \in \mathbb{R}^{m \times m}$ and, similarly, the columns of V represent the eigenvectors of the matrix $R^T R \in \mathbb{R}^{n \times n}$. The nonzero elements of the diagonal matrix D represent, then, the square roots of the eigenvalues of RR^T and $R^T R$.

The great diffusion of Singular Value Decomposition in the application field is related to the ability to approximate the matrix R so that the error related to approximation is known. This fundamental property is established by the Eckart–Young Theorem 3.1 (Golub & Van Loan, 2013).

Theorem 3.1 (of Eckart–Young): Let the matrix $R \in \mathbb{R}^{m \times n}$ and the matrices U, V, D be expressed by relation (2) and let $r = r(R)$ the rank of matrix R , with $r \leq p = \min\{m, n\}$. The matrices U_k, V_k and D_k are defined from U, V, D as follows:

- $U_k \in \mathbb{R}^{m \times k}$ contains the firsts k columns of the matrix U ;
- $V_k \in \mathbb{R}^{n \times k}$ contains the firsts k columns of the matrix V ;
- the matrix $D_k = \text{diag}(\sigma_1, \dots, \sigma_k) \in \mathbb{R}^{k \times k}$ is a square diagonal matrix that contains the first k rows and columns of the matrix D .

$$R_k = U_k D_k V_k^T \quad (4)$$

Then the matrix R_k defined in (4) is the matrix of rank k that best approximates the matrix R among all matrices of rank k :

$$\min_{r(B)=k} \|R - B\|_F^2 = \|R - R_k\|_F^2 = \sum_{i=k+1}^p \sigma_i^2, \quad (5)$$

$$\min_{r(B)=k} \|R - B\|_2 = \|R - R_k\|_2 = \sigma_{k+1}, \quad (6)$$

where $\|\cdot\|_F$ and $\|\cdot\|_2$ are the Frobenius and the spectral norm, respectively.

The approximation error is estimated by Equations (5) and (6). Theorem 3.1 allows for reducing the computational cost relating to SVD with an estimate of the acceptable error based on Equation (3). The low-rank approximation of matrices is used in many applications such as control theory, signal processing, machine learning, image compression, information retrieval, quantum physics (Conte, 2020; Conte & Lubich, 2010; Koch & Lubich, 2007; Nonnenmacher & Lubich, 2008).

The Singular Value Decomposition is usually exploited in Collaborative Filtering methods, where a rating matrix is provided. This matrix $R \in \mathbb{R}^{m \times n}$ is related to a system composed by m users and n items. The SVD reduces the needed memory to allocate the matrix and provides rating forecasts through some State of the Art techniques such as Stochastic Gradient Descent or Average Rating Filling. The rating matrix can be factorised through Equation (2). Then, the Eckart–Young Theorem can be exploited to reduce the dimension of the matrices U, V and D . To obtain the reduction, the number of latent factors $k \in \mathbb{N}$ is selected, and the matrices U_k, V_k, D_k can be obtained. Then, the matrices P and Q can be calculated according to Equations (7) and (8), respectively.

$$P = U_k \sqrt{D_k} \in \mathbb{R}^{m \times k} \quad (7)$$

$$Q = \sqrt{D_k} V^T \in \mathbb{R}^{k \times n} \quad (8)$$

The matrices P and Q contain the fake numerical features of the users and the items, respectively. For instance, in SGD and BSGD algorithms, they are defined randomly, and the elements of P and Q are corrected through the provided available ratings (van der Vlugt, 2018).

3.2. RSVD: the proposed approach

The paper aims to exploit the SVD properties in a Content-Based RS. In Collaborative Filtering RSs the known rating matrix R is factorised in the product of the matrices P and Q of relations (7) and (8). The proposed approach defined RSVD aims to elaborate the matrices \hat{P} and \hat{Q} , which contain the real profiles of users and items to obtain two matrices \tilde{P} and \tilde{Q} . The latter matrices have the same properties as the matrices P and Q generated in the Collaborative Filtering case.

To understand the properties that must be obtained by elaborating users' and items' real profiles, some theoretical analysis on Singular Value Decomposition will be discussed.

Lemma 3.2: *Let the matrix $R \in \mathbb{R}^{m \times n}$, $k \leq r(R)$, I_k the identity matrix of dimension k and the matrices U_k, V_k, D_k related to Theorem 3.1, then the following properties are satisfied:*

- (1) $U_k^T U_k = I_k$;
- (2) $V_k^T V_k = I_k$;
- (3) $R_k R_k^T = U_k D_k^2 U_k^T \in \mathbb{R}^{m \times m}$;
- (4) $R_k^T R_k = V_k D_k^2 V_k^T \in \mathbb{R}^{n \times n}$.

Proof: (1) From Theorem 3.1 the matrix U_k is defined as the matrix containing the first k left singular vectors.

$$U_k = (u_1, \dots, u_k) \quad \text{with } u_i \in \mathbb{R}^m, i = 1, \dots, k. \quad (9)$$

The vectors $u_i \forall i = 1, \dots, k$ are the columns of the orthogonal matrix U , therefore, by denoting with \langle, \rangle the standard scalar product in \mathbb{R}^m , Equation (10) is verified, where δ is the Kronecker delta.

$$\langle u_i, u_j \rangle = \delta_{ij}, \quad i, j = 1, \dots, k, \quad (10)$$

The matrix $B = U_k^T U_k$ has elements $B_{ij} = \langle u_i, u_j \rangle$, $i, j = 1, \dots, k$, so, from (10), Equation (11) is verified.

$$U_k^T U_k = B = I_k \quad (11)$$

(2) The proof is analogous to point 1.

(3) From (4) Equation (12) is satisfied.

$$R_k R_k^T = (U_k D_k V_k^T) (U_k D_k V_k^T)^T = (U_k D_k V_k^T) (V_k D_k U_k^T) = U_k D_k (V_k^T V_k) D_k U_k^T \quad (12)$$

From property 2 of the Lemma the thesis follows.

(4) The proof is analogous to point 3. ■

Through the properties analysed in the Lemma 3.2 some properties related to matrices P and Q of Equations (7) and (8) are developed in the Lemma 3.3.

Lemma 3.3: *Let the matrix $R \in \mathbb{R}^{m \times n}$ and the matrix $R_k = U_k D_k V_k^T$ obtained according to Theorem 3.1. Let P and Q the matrices defined in Equations (7) and (8). Then the following properties apply:*

- (1) $PP^T = U_k D_k U_k^T \in \mathbb{R}^{m \times m}$;
- (2) $P^T P = D_k \in \mathbb{R}^{k \times k}$;
- (3) $Q^T Q = V_k D_k V_k^T \in \mathbb{R}^{n \times n}$;
- (4) $QQ^T = D_k \in \mathbb{R}^{k \times k}$

Proof: (1) From Equation (7), the matrix $P = U_k \sqrt{D_k}$ satisfies Equation (13).

$$PP^T = (U_k \sqrt{D_k}) (U_k \sqrt{D_k})^T = (U_k \sqrt{D_k}) \left(\sqrt{D_k^T} U_k^T \right). \quad (13)$$

The matrix D_k is square and diagonal, then $D_k = D_k^T$ and Equation (13) can be developed according to Equation (14).

$$PP^T = (U_k \sqrt{D_k}) \left(\sqrt{D_k} U_k^T \right) = U_k D_k U_k^T \quad (14)$$

(2) From $P = U_k \sqrt{D_k}$ we also obtain

$$P^T P = \left(U_k \sqrt{D_k} \right)^T \left(U_k \sqrt{D_k} \right) = \sqrt{D_k^T} (U_k^T U_k) \sqrt{D_k}. \quad (15)$$

Property 1 of Lemma 3.2 allows to derive Equation (16).

$$P^T P = \sqrt{D_k^T} I_k \sqrt{D_k} = \sqrt{D_k^T} \sqrt{D_k}. \quad (16)$$

The matrix D_k is square and diagonal, then $D_k^T = D_k$ and the thesis immediately follows;

- (3) Proof similar to point 1;
- (4) Proof similar to point 2.

■

The Lemma 3.3 allows deducing that the columns of the matrix P constitute a system of orthogonal vectors as the rows of the Q matrix. These orthogonality properties are going to be emulated by the users and items profiles matrices \hat{P} and \hat{Q} through an elaboration that is going to be described in the Theorem 3.4.

Theorem 3.4: Let the input matrices $\hat{P} \in \mathbb{R}^{m \times k}$ and $\hat{Q} \in \mathbb{R}^{k \times n}$ have rank $r(\hat{P}) = r(\hat{Q}) = k$. Denote with μ_i the eigenvalues of matrix $\hat{P}^T \hat{P}$ and with λ_i the eigenvalues of the matrix $\hat{Q} \hat{Q}^T$, $i = 1, \dots, k$. Then

(i) There exist orthogonal matrices $X, Y \in \mathbb{R}^{k \times k}$ such that

$$Y^T \left(\hat{P}^T \hat{P} \right) Y = D_{\hat{P}} = \text{diag} (\mu_1, \dots, \mu_k) \quad (17)$$

$$X^T \left(\hat{Q} \hat{Q}^T \right) X = D_{\hat{Q}} = \text{diag} (\lambda_1, \dots, \lambda_k) \quad (18)$$

with $D_{\hat{P}}, D_{\hat{Q}}$ invertible matrices and $\mu_1 \geq \dots \geq \mu_k$ and $\lambda_1 \geq \dots \geq \lambda_k$.
By further setting

$$\tilde{U}_k = \hat{P} Y \sqrt{D_{\hat{P}}^{-1}}, \quad \tilde{V}_k = \hat{Q}^T X \sqrt{D_{\hat{Q}}^{-1}}, \quad \tilde{D}_k = \sqrt{D_{\hat{P}}} \sqrt{D_{\hat{Q}}} \quad (19)$$

$$\tilde{P} = \tilde{U}_k \sqrt{\tilde{D}_k}, \quad \tilde{Q} = \sqrt{\tilde{D}_k} \tilde{V}_k^T, \quad (20)$$

$$\bar{R} = \tilde{P} \tilde{Q}, \quad (21)$$

- (ii) the matrices defined \tilde{U}_k and \tilde{V}_k in (19) satisfy properties 1 and 2 of Lemma 3.2, i.e. they have orthogonal columns;
- (iii) the matrices in (19) and (20) satisfy properties 1–4 of Lemma 3.3;
- (iv) the matrix \bar{R} in (21) coincides with its reduced factorisation matrix related to Theorem 3.1, defined in (4), i.e. $\bar{R}_k = \bar{R}$.
- (v) the matrix \bar{R} in (21) has singular values $\sigma_1, \dots, \sigma_k$ satisfying;

$$\sigma_i = \sqrt{\mu_i} \sqrt{\lambda_i}, \quad i = 1, \dots, k, \quad (22)$$

Proof: In order to prove (i), we observe that the matrix $\hat{Q}\hat{Q}^T \in \mathbb{R}^{k \times k}$ is a real symmetric matrix, then for the Spectral Theorema it is diagonalisable. In particular it is similar to the diagonal matrix $D_{\hat{Q}} \in \mathbb{R}^{k \times k}$ containing its eigenvalues on the diagonal, through the orthogonal matrix $X \in \mathbb{R}^{k \times k}$ containing the corresponding eigenvectors. Analogously the matrix $\hat{P}^T \hat{P} \in \mathbb{R}^{k \times k}$ is similar to the diagonal matrix $D_{\hat{P}} \in \mathbb{R}^{k \times k}$ containing its eigenvalues on the diagonal, through the orthogonal matrix $Y \in \mathbb{R}^{k \times k}$ containing the corresponding eigenvectors. Furthermore, from the hypothesis that matrices $\hat{P}^T \hat{P}$ e $\hat{Q}\hat{Q}^T$ have maximal rank k , we deduce that the respective eigenvalues are all non-zero

$$\begin{aligned} \lambda_j &\neq 0 \quad \forall j = 1, \dots, k \\ \mu_i &\neq 0 \quad \forall i = 1, \dots, k \end{aligned} \quad (23)$$

In particular, the matrices $D_{\hat{Q}}$ e $D_{\hat{P}}$ are invertible.

In order to prove (ii), by exploiting (17) and (18), we obtain

$$\tilde{U}_k^T \tilde{U}_k = \sqrt{D_{\hat{P}}^{-1}} Y^T \hat{P} \hat{P} Y \sqrt{D_{\hat{P}}^{-1}} = I_k \quad (24)$$

and

$$\tilde{U}_k^T \tilde{U}_k = \sqrt{D_{\hat{P}}^{-1}} Y^T \hat{P} \hat{P} Y \sqrt{D_{\hat{P}}^{-1}} = I_k, \quad (25)$$

where I_k is the identity matrix of dimension k .

As regards (iii), the properties 1–4 of Lemma 3.3 immediately follow from (20), by using (ii).

We finally prove properties (iv) and (v). By substituting (20) in Equation (21) we obtain:

$$\bar{R} = \tilde{P}\tilde{Q} = \tilde{U}_k \tilde{D}_k \tilde{V}_k^T \quad (26)$$

Then, by construction, the obtained matrix \bar{R} coincides with matrix \bar{R}_k associated to relation (4).

Furthermore, again by construction, the singular values $\sigma_1, \dots, \sigma_k$ of \bar{R} coincide with the product of the square roots of the eigenvalues of the matrices $\hat{P}^T \hat{P}$ and $\hat{Q}\hat{Q}^T$:

$$\sigma_i = \left(\hat{D}_k \right)_{ii} = \sqrt{\mu_i} \sqrt{\lambda_i} \quad \forall i = 1, \dots, k \quad (27)$$

■

The Theorem 3.4 constitutes the theoretical basis of the RSVD method and allows generating the matrices \tilde{P} and \tilde{Q} that represent the fake numerical profiles matrices related to

users and items. Moreover, they are obtained from the input matrices \hat{P} and \hat{Q} that contain the user's and items profiles. The RSVD method has been applied in designing an application to support an individual visiting a new location (Casillo et al., 2019; Casillo, Conte, et al., 2022). The application operates in three successive phases:

- the *Fetch Phase* acquires data about the user's position and provides an initial description of the site of interest. It also searches for all resources, services, and events through external sites and requests access to the user's social data for the creation of a profile that describes his interests;
- the *Decode Phase* calculates user/resource affinities and introduces the context variables through the Context Dimension Tree;
- the *Execute Phase* presents the information processed in the previous phases through a contiguous narrative. In addition, the system provides a list of recommended activities to users.

In the design of the described application, the Recommender Systems are inserted in the Decode Phase with the calculation of user/resource affinities. The available data for creating the rating forecasts (in the specific case coinciding with the affinities) are user profiles and information acquired on resources through external sites. According to Casillo, Conte, et al. (2022), the construction of users' profiles takes advantage of the social network information and calculates the friendship degree among users and the level of interest in the selected categories of interest. The items' profiles are built through binary vectors in which each feature describes if the item belongs to the specific category of interest. However, the creation of profiles is strictly related to each study case. Applying the recommendation method in a specific field can involve different methods to create profiles. For instance, the aim to enhance cultural heritage sites can need to exploit experts to profile items (Casillo, De Santo, et al., 2021); instead, the application in the learning paths field can involve the necessity to use a questionnaire to profile users (Carbone et al., 2021). The different techniques exploited in various fields allow determining the input matrices \hat{P} and \hat{Q} .

The elaboration of input matrices \hat{P} and \hat{Q} is described by the Algorithm 1 and summarised in Figure 1. The input matrices are elaborated through the Users Elaboration Submodule and the Items Elaboration Submodule. The first Submodule processes points 2 and 4, which are related to the Algorithm 1. In particular, the Users Elaboration Submodule calculates the eigenvalues and the eigenvectors of the matrix $\hat{P}^T \hat{P}$ and memorises the first

Algorithm 1 RSVD Algorithm

- 1: Input: $\hat{P} \in \mathbb{R}^{m \times k}$, $\hat{Q} \in \mathbb{R}^{k \times n}$;
 - 2: Calculation of the eigenvalues and the eigenvectors of the matrix $\hat{P}^T \hat{P}$ memorised in the matrices Y and $D_{\hat{P}}$, respectively;
 - 3: Calculation of the eigenvalues and the eigenvectors of the matrix $\hat{Q} \hat{Q}^T$ memorised in the matrices X and $D_{\hat{Q}}$, respectively;
 - 4: $\tilde{P} = \hat{P}Y$ according to Equation (19) and (20);
 - 5: $\tilde{Q} = X^T \hat{Q}$ according to Equations (19) and (20);
 - 6: $\tilde{R} = \tilde{P} \tilde{Q}$ according to Equation (21);
 - 7: Regularisation of \tilde{R} according to process described in Barragáns-Martínez et al. (2010);
-

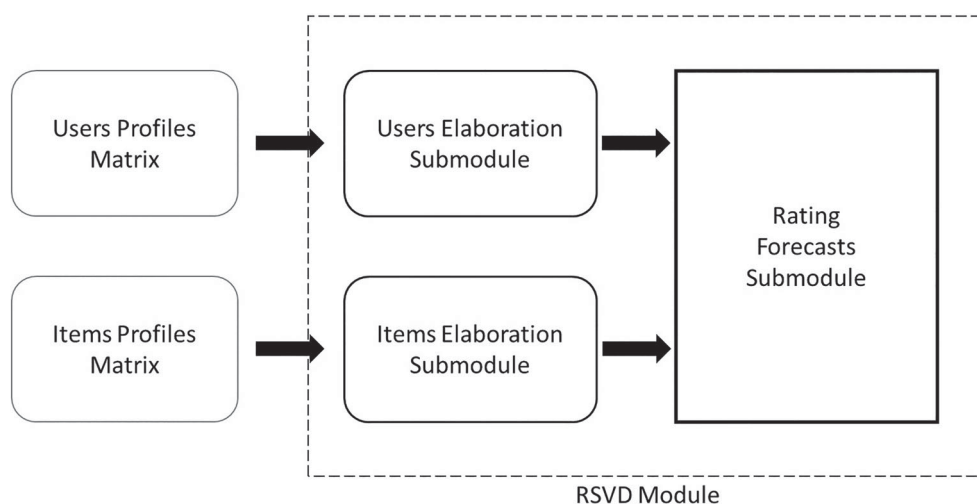


Figure 1. Architecture related to RSVD method.

in the diagonal elements of the matrix $D_{\hat{P}}$ and the seconds in the matrix Y . Then the matrix $\tilde{P} = \hat{P}Y$ is obtained. Instead, the second Submodule processes points 3 and 5 related to the Algorithm 1. In particular, the Items Elaboration Submodule calculates the eigenvalues and the eigenvectors of the matrix $\hat{Q}\hat{Q}^T$ and memorises the firsts in the diagonal elements of the matrix $D_{\hat{Q}}$ and the seconds in the matrix X . Then the matrix $\tilde{Q} = X^T\hat{Q}$ is obtained.

The obtained matrices \tilde{P} and \tilde{Q} satisfy the orthogonality properties as proved in the Theorem 3.4. Finally, the Rating Forecast Submodule calculates the rating forecasts through the matrix's product according to Equation (21). Finally, the rating forecasts matrix \bar{R} is regularised through the procedure described by Barragáns-Martínez et al. (2010) for Collaborative Filtering methods.

4. Numerical results

This Section provides the numerical experiments related to the accuracy measure of the RSVD method. The proposed approach is compared to Average Rating Filling (ARF) (Anastasiu et al., 2016; van der Vlugt, 2018), Stochastic Gradient Descent (SGD) (Anastasiu et al., 2016; R. Wang et al., 2019; Zhao et al., 2019), and Biased Stochastic Gradient Descent (BSGD) (R. Wang et al., 2019; Zhao et al., 2019) methods. Another method compared with RSVD is the process that obtains rating forecasts through the product of the input matrices \hat{P} and \hat{Q} . This method is defined as PQ and aims to give consistency to RSVD and apply the Theorem 3.4.

To compare the accuracy related to the methods, the datasets MovieLens100k and MovieLens10M are exploited, and the accuracy measures are the Mean Absolute Error (MAE) and Root Mean Square Error (RMSE). Let N be the set of pairs (i, j) for which the numerical test is carried out and \hat{r}_{ij} the prediction made for the pair (i, j) . Mean Absolute Error is

defined in Equation (28) and the Root Mean Square Error is defined in Equation (29).

$$\text{MAE} = \sum_{(i,j) \in N} \frac{|\hat{r}_{ij} - r_{ij}|}{|N|} \quad (28)$$

$$\text{RMSE} = \sqrt{\sum_{(i,j) \in N} \frac{(\hat{r}_{ij} - r_{ij})^2}{|N|}} \quad (29)$$

The exploited datasets are MovieLens100k, and MovieLens10M made available by the GroupLens Research Project of the University of Minnesota (Harper & Konstan, 2015). The item profile matrix \hat{Q} was obtained through the 'u.item' file for MovieLens100k and from the "movies.csv" file for the MovieLens10M database. The item matrix element (c, j) is 1 if the movie j th is part of the category c th otherwise it is zero. Movie categories are available in the 'u.genre' file provided by the database. About the user profiles matrix, in the case of MovieLens100k, the matrix \hat{P} was formed through the means of the known ratings by category associated with each user. On the other hand, in the case of the MovieLens 10M database, tag information was exploited for user profiling. Since the available tags are 100,000, users without information were provided an average profile $p = \frac{(1, \dots, 1)}{\|(1, \dots, 1)\|}$ that does not characterise the user but that allows the system to provide the best items available. This choice allows overcoming the problem of the initial cold start of the system that later will have the opportunity to adjust the specific user's preferences. The chosen value k is selected by the number of movie categories provided equally to 18.

The employment of datasets that contain known ratings and tags penalises the proposed approach. However, this application provides an estimation of the proposed approach's accuracy.

The parameters needed for SGD and BSGD to provide reliable rating forecasts are selected experimentally, and the results provided are summarised in Figures 2 and 3. Figure 2 describes the variation of the RMSE related to the parameter α . Figure 3 describes the variation of the RMSE related to the variation of the parameters α and λ .

After the experimental evaluation of the parameters, the values exploited to compare these methods to RSVD are:

- $\alpha = 0.01$ for SGD;
- $\alpha = 0.04$ and $\lambda = 0.005$ for BSGD.

The number of iterations needed for the training phase of SGD and BSGD is 10, 20, and 40. Moreover, due to oscillating results on the value of MAE for the methods SGD and BSGD, the reported results are the median of the value obtained in 11 experiments.

Furthermore, the rating forecasts provided by RSVD, and PQ methods on MovieLens100k and MovieLens10M are regularised on the interval $[1, 5]$.

4.1. Numerical results on movieLens100k

The results provided by the methods on the dataset MovieLens100k are summarised in Table 2. The RSVD method returns a worse accuracy than the comparison methods ARF, SGD, and BSGD, except for the case of the BSGD method applied with 10 iterations.

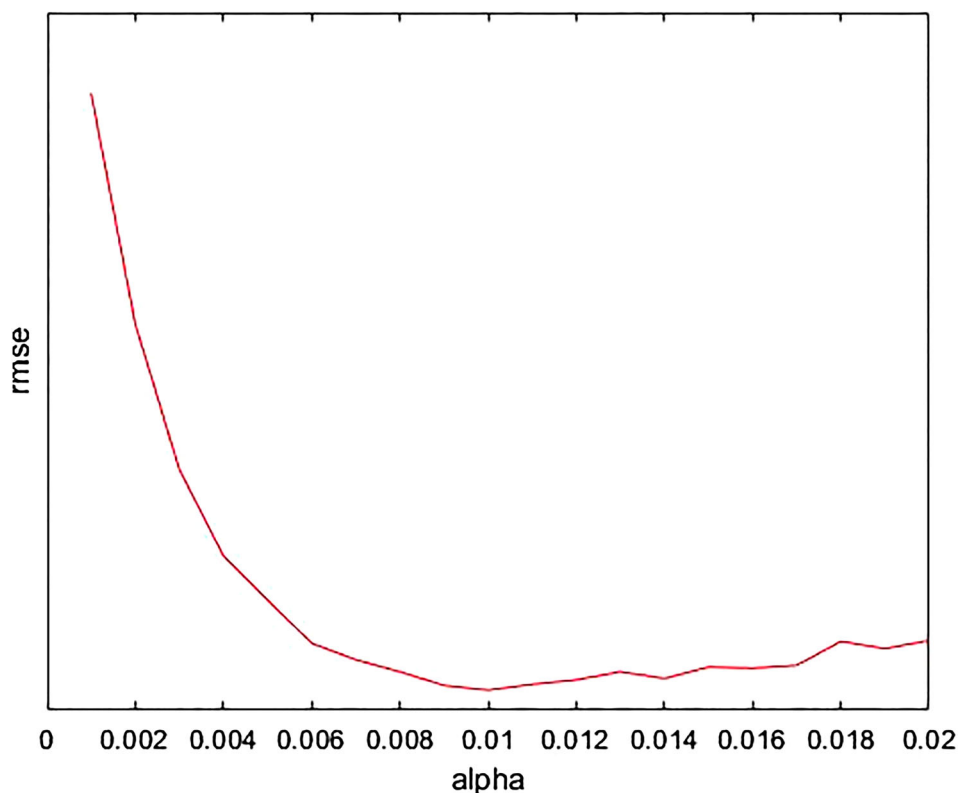


Figure 2. Graphical representation of RMSE error of method SDG when α changes.

These results applied in the field that benefits the comparison methods testify that despite the adaption of the proposed approach on a classical Collaborative Filtering dataset, the provided accuracy is promising. Indeed, the provided results provide evidence to generate reliable rating forecasts despite the need to adapt the Content-Based procedure of RSVD to the presence of available ratings. In addition, the results provided allow evaluating the idea of using available ratings for profile creation in RSVD in case of the impossibility of acquiring other typologies of information. The use of a matrix factorisation technique improves the ability of RSVD to deal with the scalability problem compared to classical Content-Based methods.

The value k , in the RSVD method, represents the number of categories of interest to the user. Therefore it is possible to intuitively build user and item profiles needed for RSVD, without adding too much computational cost to the method.

The next focus is on the differences between the RSVD and PQ procedures. The poor results found for the second method justify the increase in complexity associated with the first, which, as mentioned, is comparable with previously tested methods.

Finally, RSVD is compared with the recommendation method exploited by Casillo et al. (2019). This methodology, defined as PREC, represents a Content-Aware RS in which the affinity user-item calculation can be extrapolated to compare the accuracy provided with the RSVD method. Table 3 shows the accuracy results related to MAE and RMSE provided by RSVD and PREC in MovieLens100k dataset.

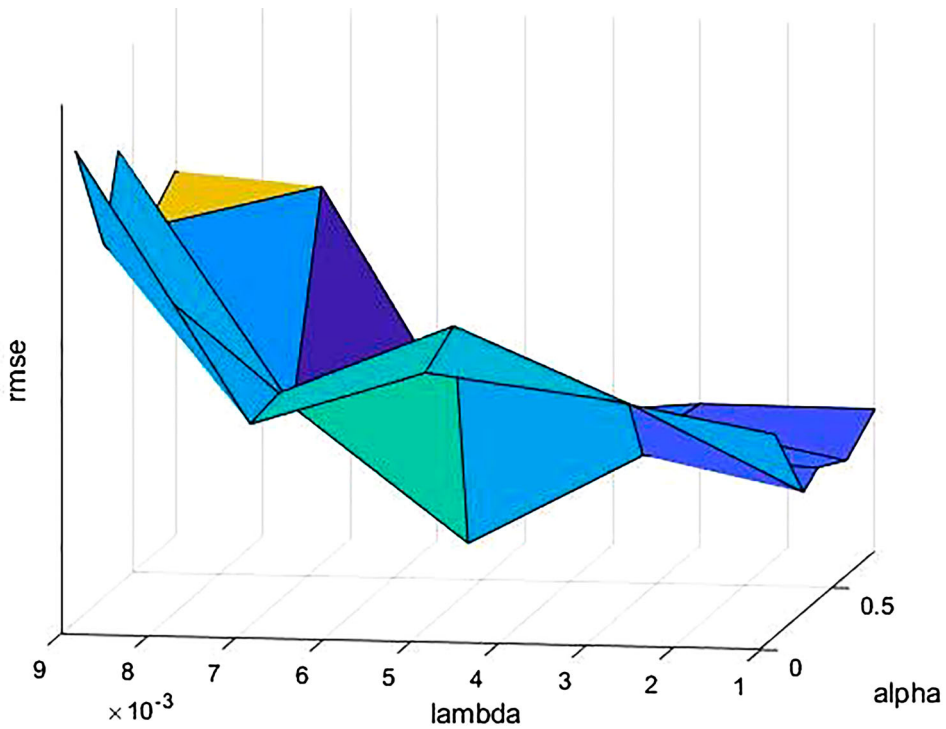


Figure 3. Graphical representation of RMSE error of method BSDG when α and λ change.

Table 2. Results of the methods ARF, SGD, BSDG, PQ, and RSVD on MovieLens100k dataset.

Method	k	iter (if any)	MAE	RMSE
ARF	18		0.6882	0.8741
SGD	18	10	0.7227	0.7813
	18	20	0.7210	0.6973
	18	40	0.6506	0.6586
	18	10	0.4829	1.0721
BSDG	18	20	0.4821	0.9198
	18	40	0.5824	0.8043
	18		1.6458	1.8986
PQ	18		0.7664	0.9512
RSVD	18			

Table 3. Accuracy related to RSVD and PREC methods on MovieLens100k dataset.

Method	MAE	RMSE
RSVD	0.7664	0.9512
PREC	0.8757	1.1618

The accuracy provided by the RSVD method is better than the recommendation method exploited in Casillo et al. (2019). The Mean Absolute Error and the Root Mean Square Error are lower than the errors related to PREC.

4.2. Numerical tests on movieLens10M

This subsection compares the RSVD method with SGD and BSGD on MovieLens10M. This database contains 10 million ratings and 100,000 tags. The \hat{P} is built only with tags. If there are not enough ratings in order to determine a profile of user a standard one is used. Defined $p = (1, \dots, 1) \in \mathbb{R}^k$, the standard user profile is provided by Equation (30).

$$p_u = \left(\frac{1}{\|p\|} \quad \dots \quad \frac{1}{\|p\|} \right) \in \mathbb{R}^k \quad (30)$$

Instead, the SGD and BSGD methods are tested with training sets selected in various cardinalities to prove that RSVD can give better rating forecasts with less information. The cardinalities chosen for training sets are:

- 100,000 training ratings;
- 165,000 training ratings;
- 500,000 training ratings;
- 1,000,000 training ratings;

The methods exploit all possible cardinality of the training sets to the training phase of SGD and BSGD. Instead, RSVD uses all ratings of the dataset in the test phase because this method does not need available ratings to predict forecasts.

Table 4 provides the numerical results of the methods with the specific cardinality related to the training phase. The comparison between SGD and RSVD underlines that the first method needs at least 1 million known ratings to provide better MAE and RMSE than RSVD. The numerical results testify that SGD provides better forecasts than RSVD through an information quantity greater than 100 times that of RSVD. The accuracy difference provided by numerical results can be related to the different information exploited by Collaborative Filtering methods (SGD and BSGD) and the proposed approach RSVD. The RSVD method does not exploit users' preferences but only the items category and the standard user profile (30). Moreover, RSVD does not need a training phase and builds predictions without using available ratings, and it gives good values of MAE and RMSE. In addition, using an average profile without information allows for overcoming the problem of a cold start in practice.

Comment on Numerical Results.

Section 4 focuses on the numerical results of the proposed approach. The provided results return an accuracy measure worse than the ARF, SGD, and BSGD methods on

Table 4. Results of the methods SGD, BSGD and RSVD.

Method	Training cardinality	MAE	RMSE
SGD	100,000 ratings	2.4420	2.6583
	165,000 ratings	1.9197	2.2261
	500,000 ratings	0.9354	1.2274
	1 million ratings	0.7800	1.0194
BSGD	100,000 ratings	0.8612	1.0975
	165,000 ratings	0.8870	1.1322
	500,000 ratings	0.9274	1.1878
	1 million ratings	0.8978	1.1547
RSVD	100,000 tags	0.9588	1.1877

MovieLens100k. Instead, the comparison with the PQ method provides evidence of the consistency of the Theorem 3.4 and the need to modify the users and items profiles to improve the rating forecasts' reliability. Moreover, the comparison with PREC testifies that the proposed approach improves the recommendation method exploited by Casillo et al. (2019) that represents the field in which the RSVD method is born. On MovieLens10M, the ability of the proposed approach to work in the information lack field is tested. The comparison methods exploited the users' preferences through a few known ratings, and the training cardinality varies in the four possibilities described before. Instead, RSVD works with the user interaction with the movies or creates a standard profile that can not exploit the user preference. Despite this disadvantage, the accuracy is promising because the test phase is based on known rating datasets. These datasets are focused on the Collaborative Filtering methods of testing. Indeed, the proposed approach can work in the classical Content-Based field, with a process to create users and items profiles, with the ability to deal with the scalability problem.

The experiments testify that the proposed approach provides reliable rating forecasts. It does not overcome the Collaborative Filtering comparison methods but obtains satisfactory accuracy results, despite the proposed approach being Content-Based. Moreover, on the MovieLens10M dataset, the recommendation methodology provides an MAE of 0.9588 and an RMSE of 1.1877, working with tags for user profile creation. Moreover, the Content-Based recommendation methods have application fields different from Collaborative methods. Then, to validate the proposed approach, efficacy experiments in various application domains must be organised.

5. Conclusions

This paper aims to propose a novel methodology that provides rating forecasts without known ratings. The experimental results are challenging compared to the methods presented in Section 3. After the theoretical formalisation of the RSVD method, it may be concluded that the possibility of making forecasts, without the need for known ratings through the information associated with the characteristics of users and items, allows providing an adequate starting point for a system that can not take advantage of standard procedures. Due to the encouraging results regarding the RSVD method, future works include introducing the context variables to complete and finalise the Decode Phase within the presented case study.

Disclosure statement

No potential conflict of interest was reported by the author(s).

ORCID

Domenico Santaniello  <http://orcid.org/0000-0002-5783-1847>

References

Aberger, C. R. (2014). Recommender: An analysis of collaborative filtering techniques. *Personal and Ubiquitous Computing Journal*. <https://doi.org/10.25103/jestr.104.18>

- Adomavicius, G., & Tuzhilin, A. (2015). Context-aware recommender systems. In *Recommender systems handbook* (2nd ed., pp. 191–226).
- Allam, Z., & Dhunny, Z. A. (2019). On big data, artificial intelligence and smart cities. *Cities*, 89, 80–91. <https://doi.org/10.1016/j.cities.2019.01.032>
- Anastasiu, D., Christakopoulou, E., Smith, S., Sharma, M., & Karypis, G. (2016). Big data and recommender systems. Retrieved from the University of Minnesota Digital Conservancy. <https://hdl.handle.net/11299/215998>
- Barragáns-Martínez, A. B., Costa-Montenegro, E., Burguillo, J. C., Rey-López, M., Mikic-Fonte, F. A., & Peleteiro, A. (2010). A hybrid content-based and item-based collaborative filtering approach to recommend TV programs enhanced with singular value decomposition. *Information Sciences*, 180(22), 4290–4311. <https://doi.org/10.1016/j.ins.2010.07.024>
- Bhavana, P., Kumar, V., & Padmanabhan, V. (2019). Block based singular value decomposition approach to matrix factorization for recommender systems. *Pattern Recognition Letters*. <https://doi.org/10.48550/arXiv.1907.07410>
- Bobadilla, J., Ortega, F., Hernando, A., & Gutiérrez, A. (2013). Recommender systems survey. *Knowledge-Based Systems*, 46(July), 109–132. <https://doi.org/10.1016/j.knosys.2013.03.012>
- Bokde, D., Girase, S., & Mukhopadhyay, D. (2015). Matrix factorization model in collaborative filtering algorithms: A survey. *Procedia Computer Science*, 49(1), 136–146. <https://doi.org/10.1016/j.procs.2015.04.237>
- Carbone, M., Colace, F., Lombardi, M., Marongiu, F., Santaniello, D., & Valentino, C. (2021). An adaptive learning path builder based on a context aware recommender system. In *2021 IEEE frontiers in education conference (FIE)* (pp. 1–5). IEEE.
- Casillo, M., Colace, F., Conte, D., Lombardi, M., Santaniello, D., & Valentino, C. (2021). Context-aware recommender systems and cultural heritage: A survey. *Journal of Ambient Intelligence and Humanized Computing*, 1–19. <https://doi.org/10.1007/s12652-021-03438-9>
- Casillo, M., Colace, F., Santo, M., Lemma, S., & Lombardi, M. (2019). CAT: A context aware teller for supporting tourist experiences. *International Journal of Computational Science and Engineering*, 20(1), 69–87. <https://doi.org/10.1504/IJCSE.2019.103255>
- Casillo, M., Conte, D., Lombardi, M., Santaniello, D., Troiano, A., & Valentino, C. (2022). A content-based recommender system for hidden cultural heritage sites enhancing. In *Proceedings of sixth international congress on information and communication technology* (pp. 97–109). Springer.
- Casillo, M., Conte, D., Lombardi, M., Santaniello, D., & Valentino, C. (2021). Recommender system for digital storytelling: A novel approach to enhance cultural heritage. In *International conference on pattern recognition* (pp. 304–317). Springer.
- Casillo, M., De Santo, M., Lombardi, M., Mosca, R., Santaniello, D., & Valentino, C. (2021). Recommender systems and digital storytelling to enhance tourism experience in cultural heritage sites. In *2021 IEEE international conference on smart computing (SMARTCOMP)* (pp. 323–328). IEEE.
- Casillo, M., Gupta, B. B., Lombardi, M., Lorusso, A., Santaniello, D., & Valentino, C. (2022). Context aware recommender systems: A novel approach based on matrix factorization and contextual bias. *Electronics*, 11(7), 1003. <https://doi.org/10.3390/electronics11071003>
- Castiglione, A., Colace, F., Moscato, V., & Palmieri, F. (2018). CHIS: A big data infrastructure to manage digital cultural items. *Future Generation Computer Systems*, 86(September), 1134–1145. <https://doi.org/10.1016/j.future.2017.04.006>
- Conte, D. (2020). Dynamical low-rank approximation to the solution of parabolic differential equations. *Applied Numerical Mathematics*, 156(October), 377–384. <https://doi.org/10.1016/j.apnum.2020.05.011>
- Conte, D., & Lubich, C. (2010). An error analysis of the multi-configuration time-dependent Hartree method of quantum dynamics. *ESAIM: Mathematical Modelling and Numerical Analysis*, 44(4), 759–780. <https://doi.org/10.1051/m2an/2010018>
- Desrosiers, C., & Karypis, G. (2011). A comprehensive survey of neighborhood-based recommendation methods. In *Recommender systems handbook* (pp. 107–144) Springer.
- Eirinaki, M., Gao, J., Varlamis, I., & Tserpes, K. (2018). Recommender systems for large-scale social networks: A review of challenges and solutions. *Future Generation Computer Systems*, 78(1), 413–418. <https://doi.org/10.1016/j.future.2017.09.015>

- Geuens, S., Coussement, K., & De Bock, K. W. (2018). A framework for configuring collaborative filtering-based recommendations derived from purchase data. *European Journal of Operational Research*, 265(13), 208–218. <https://doi.org/10.1016/j.ejor.2017.07.005>
- Golub, G. H., & Van Loan, C. F. (2013). *Matrix computations* (4th ed.). The Johns Hopkins University Press.
- Gorripati, S. K., & Vatsavayi, V. K. (2017). A community based content recommender systems. *International Journal of Applied Engineering Research*, 12(22), 12989–12996.
- Gu, Y., Yang, X., Peng, M., & Lin, G. (2020). Robust weighted SVD-type latent factor models for rating prediction. *Expert Systems with Applications*, 141(March), Article 112885. <https://doi.org/10.1016/j.eswa.2019.112885>
- Harper, F. M., & Konstan, J. A. (2015). The movielens datasets: History and context. *Acm Transactions on Interactive Intelligent Systems (tiis)*, 5(4), 1–19. <https://doi.org/10.1145/2827872>
- He, X., He, Z., Song, J., Liu, Z., Jiang, Y.-G., & Chua, T.-S. (2018). Nais: Neural attentive item similarity model for recommendation. *IEEE Transactions on Knowledge and Data Engineering*, 30(12), 2354–2366. <https://doi.org/10.1109/TKDE.2018.2831682>
- Hernando, A., Bobadilla, J., & Ortega, F. (2016). A non negative matrix factorization for collaborative filtering recommender systems based on a Bayesian probabilistic model. *Knowledge-Based Systems*, 97(April), 188–202. <https://doi.org/10.1016/j.knosys.2015.12.018>
- Jain, G., Mahara, T., & Tripathi, K. N. (2020). A survey of similarity measures for collaborative filtering-based recommender system. In *Soft computing: Theories and applications* (pp. 343–352). Springer.
- Javed, U., Shaukat, K., Hameed, I. A., Iqbal, F., Alam, T. M., & Luo, S. (2021). A review of content-based and context-based recommendation systems. *International Journal of Emerging Technologies in Learning (iJET)*, 16(3), 274–306. <https://doi.org/10.3991/ijet.v16i03.18851>
- Karabadjji, N. E. I., Beldjoudi, S., Seridi, H., Aridhi, S., & Dhifli, W. (2018). Improving memory-based user collaborative filtering with evolutionary multi-objective optimization. *Expert Systems with Applications*, 98(May), 153–165. <https://doi.org/10.1016/j.eswa.2018.01.015>
- Koch, O., & Lubich, C. (2007). Dynamical low-rank approximation. *SIAM Journal on Matrix Analysis and Applications*, 29(2), 434–454. <https://doi.org/10.1137/050639703>
- Koren, Y., Bell, R., & Volinsky, C. (2009). Matrix factorization techniques for recommender systems. *Computer*, 42(8), 30–37. <https://doi.org/10.1109/MC.2009.263>
- Kumaravel, A., & Dutta, P. (2014). Application of Pca for context selection for collaborative filtering. *Middle-East Journal of Scientific Research*, 20(1), 88–93. <https://doi.org/10.5829/idosi.mejsr.2014.20.01.11254>
- Liang, B., Xu, B., Wu, X., Wu, D., Yang, D., Xiao, Y., & Wang, W. (2019). A community-based collaborative filtering method for social recommender systems. In *2019 IEEE international conference on web services (ICWS)* (pp. 159–162). IEEE.
- Liu, X., Wang, G., & Bhuiyan, M. Z. A. (2021). Personalised context-aware re-ranking in recommender system. *Connection Science*, 34(1), 319–338. <https://doi.org/10.1080/09540091.2021.1997915>
- Lops, P., De Gemmis, M., & Semeraro, G. (2011). Content-based recommender systems: State of the art and trends. In *Recommender systems handbook* (pp. 73–105). Springer.
- Luo, X., Wang, D., Zhou, M., & Yuan, H. (2019). Latent factor-based recommenders relying on extended stochastic gradient descent algorithms. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 51(2), 916–926. <https://doi.org/10.1109/TSMC.6221021>
- 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. <https://doi.org/10.1109/TII.2014.2308433>
- Mnih, A., & Salakhutdinov, R. R. (2008). Probabilistic matrix factorization. *Advances in Neural Information Processing Systems*, 1257–1264.
- Nguyen, J., & Zhu, M. (2013). Content-boosted matrix factorization techniques for recommender systems. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, 6(4), 286–301. <https://doi.org/10.1002/sam.11184>
- Nickolls, J., Buck, I., Garland, M., & Skadron, K. (2008). Scalable parallel programming with cuda: Is cuda the parallel programming model that application developers have been waiting for?. *Queue*, 6(2), 40–53. <https://doi.org/10.1145/1365490.1365500>

- Nonnenmacher, A., & Lubich, C. (2008). Dynamical low-rank approximation: Applications and numerical experiments. *Mathematics and Computers in Simulation*, 79(4), 1346–1357. <https://doi.org/10.1016/j.matcom.2008.03.007>
- Pérez-Almaguer, Y., Yera, R., Alzahrani, A. A., & Martínez, L. (2021). Content-based group recommender systems: A general taxonomy and further improvements. *Expert Systems with Applications*, 184(December), Article 115444. <https://doi.org/10.1016/j.eswa.2021.115444>
- Portugal, I., Alencar, P., & Cowan, D. (2018). The use of machine learning algorithms in recommender systems: A systematic review. *Expert Systems with Applications*, 97(May), 205–227. <https://doi.org/10.1016/j.eswa.2017.12.020>
- Quijano-Sánchez, L., Cantador, I., Cortés-Cediel, M. E., & Gil, O. (2020). Recommender systems for smart cities. *Information Systems*, 92(September), 101545. <https://doi.org/10.1016/j.is.2020.101545>
- Rendle, S. (2010). Factorization machines. In *2010 IEEE international conference on data mining* (pp. 995–1000). IEEE.
- Renjith, S., Sreekumar, A., & Jathavedan, M. (2020). An extensive study on the evolution of context-aware personalized travel recommender systems. *Information Processing & Management*, 57(1), Article 102078. <https://doi.org/10.1016/j.ipm.2019.102078>
- Ricci, F., Rokach, L., & Shapira, B. (2015). Recommender systems: Introduction and challenges. In *Recommender systems handbook* (pp. 1–34). Springer.
- Sinha, B. B., & Dhanalakshmi, R. (2019a). Evolution of recommender system over the time. *Soft Computing*, 23(23), 12169–12188. <https://doi.org/10.1007/s00500-019-04143-8>
- Sinha, B. B., & Dhanalakshmi, R. (2019b). Evolution of recommender paradigm optimization over time. *Journal of King Saud University-Computer and Information Sciences*, 34(4), 1047–1059. <https://doi.org/10.1016/j.jksuci.2019.06.008>
- Symeonidis, P., & Zioupos, A. (2016). *Matrix and tensor factorization techniques for recommender systems* (Springer).
- Tarus, J. K., Niu, Z., & Mustafa, G. (2018). Knowledge-based recommendation: A review of ontology-based recommender systems for e-learning. *Artificial Intelligence Review*, 50(1), 21–48. <https://doi.org/10.1007/s10462-017-9539-5>
- van der Vlugt, Y. (2018). Large-scale SVD algorithms for latent semantic indexing, recommender systems and image processing.
- Vozalis, M. G., & Margaritis, K. G. (2007). A recommender system using principal component analysis. In *Published in 11th panhellenic conference in informatics* (pp. 271–283).
- Wang, D., Liang, Y., Xu, D., Feng, X., & Guan, R. (2018). A content-based recommender system for computer science publications. *Knowledge-Based Systems*, 157(October), 1–9. <https://doi.org/10.1016/j.knosys.2018.05.001>
- Wang, R., Cheng, H. K., Jiang, Y., & Lou, J. (2019). A novel matrix factorization model for recommendation with LOD-based semantic similarity measure. *Expert Systems with Applications*, 123(June), 70–81. <https://doi.org/10.1016/j.eswa.2019.01.036>
- Yu, M., Quan, T., Peng, Q., Yu, X., & Liu, L. (2021). A model-based collaborate filtering algorithm based on stacked AutoEncoder. *Neural Computing and Applications*, 34(4), 1–9. <https://doi.org/10.1007/s00521-021-05933-8>
- Zhang, F., Yuan, N. J., Lian, D., Xie, X., & Ma, W-Y. (2016). Collaborative knowledge base embedding for recommender systems. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 353–362).
- Zhang, G., Liu, Y., & Jin, X. (2020). A survey of autoencoder-based recommender systems. *Frontiers of Computer Science*, 14(2), 430–450. <https://doi.org/10.1007/s11704-018-8052-6>
- Zhang, Y., & Chen, X. (2020). Explainable recommendation: A survey and new perspectives. *Foundations and Trends in Information Retrieval*, 14(1), 1–101. <https://doi.org/10.1561/15000000066>
- Zhao, J., Geng, X., Zhou, J., Sun, Q., Xiao, Y., Zhang, Z., & Fu, Z. (2019). Attribute mapping and autoencoder neural network based matrix factorization initialization for recommendation systems. *Knowledge-Based Systems*, 166(February), 132–139. <https://doi.org/10.1016/j.knosys.2018.12.022>