




# Context-aware recommender systems and cultural heritage: a survey

Mario Casillo<sup>1</sup> · Francesco Colace<sup>1</sup> · Dajana Conte<sup>2</sup> · Marco Lombardi<sup>1</sup>  · Domenico Santaniello<sup>1</sup> · Carmine Valentino<sup>2</sup>

Received: 22 February 2021 / Accepted: 5 August 2021 / Published online: 19 August 2021  
© The Author(s) 2021

## Abstract

In the Big Data era, every sector has adapted to technological development to service the vast amount of information available. In this way, each field has benefited from technological improvements over the years. The cultural and artistic field was no exception, and several studies contributed to the aim of the interaction between human beings and artistic-cultural heritage. In this scenario, systems able to analyze the current situation and recommend the right services play a crucial role. In particular, in the Recommender Systems field, Context-Awareness helps to improve the recommendations provided. This article aims to present a general overview of the introduction of Context analysis techniques in Recommender Systems and discuss some challenging applications to the Cultural Heritage field.

**Keywords** Context-aware computing · Context-aware recommender systems · Cultural heritage · Recommender systems

## 1 Introduction

Since the early years of the twentieth century, the birth of the first e-commerce websites has given a decisive impetus to the development of Recommender Systems. They represent tools able to analyze and filter a large amount of information, aiming to suggest suitable objects or services to a user.

Over the years, the concept of Context-Aware Recommender Systems has been introduced to obtain increasingly reliable rating forecasts. The term “context” does not have a single definition, as it is applied in many areas of study. For example (Adomavicius and Tuzhilin 2015, 2011):

- *Data Mining*, in which the context indicates the events that characterize the states of a user and can determine the variations in the preferences of the latter;
- *E-commerce Personalization*, in which the context is exploited to determine classes of user behavior according to purchasing intentions;
- *Mobile Context-Aware Systems*, where the context represents information that can be accessed via mobile devices (location, people or places of interest nearby);
- *Marketing and Management*, where context information makes the system aware of the specific user’s preferences to apply the appropriate sales strategy.

---

✉ Marco Lombardi  
malombardi@unisa.it

Mario Casillo  
mcasillo@unisa.it

Francesco Colace  
fcolace@unisa.it

Dajana Conte  
dajconte@unisa.it

Domenico Santaniello  
dsantaniello@unisa.it

Carmine Valentino  
cvalentino@unisa.it

The difficulty of finding a specific purpose is expressed by the words of Bazire and Brezillon (Bazire and Brézillon 2005; Adomavicius and Tuzhilin 2015, 2011):

*“... it is difficult to find a relevant definition satisfying in any discipline. Is context a frame for a given object? Is it the set of elements that have any influence on the object? Is it possible to define context a priori or state the effects of a posteriori? Is it something static or dynamic?”*

To make the concept of “context” more comprehensible, some of the main definitions used in the field of Recommender Systems are set out below:

<sup>1</sup> DIIn - University of Salerno, Fisciano, Italy

<sup>2</sup> DIPMAT - University of Salerno, Fisciano, Italy

- **K. Prahaland:** “the ability to reach out and touch customers anywhere at anytime means that companies must deliver not just competitive products but also unique, real-time customer experiences shaped by customer context” (Adomavicius and Tuzhilin 2011);
- **Abowd et al:** “any information useful to characterize the situation of an entity that can affect the way users interact with systems”(Abowd et al. 1999);
- **Schilit et al:** location, nearby people and things, and change which happens to them” (Shin et al. 2009);
- **Dey:** “emotion, focus of concentration, location, adaptation, date and time, nearby things, nearby people” (Shin et al. 2009; Dey 2001).

The ability to analyze a contextual environment allows for improving recommendations provided to the user.

This capacity is functional in Cultural Heritage, where Context-Aware Recommender Systems are fundamental means to make more addictive the experience of visiting a museum or an archaeological park. This occurs because it will be possible to suggest the path more suitable based on a single user’s preferences (Colace et al. 2020a). The analysis of the context will allow a more significant interaction of the user with the chosen point of interest.

For this purpose, some techniques and some systems, capable of integrating contextual information with recommendations, will be presented. Subsequently, some applications of interest in the specific field of Cultural Heritage will be presented.

This work is divided as follows:

- Section 2 provides a general introduction to Recommender Systems;
- Section 3 presents an overview of techniques for introducing contextual information into Recommender Systems;
- Section 4 analyzes the datasets available for Context-Aware Recommender Systems;
- Section 5 presents several system architectures related to Recommender Systems in the field of Cultural Heritage;
- Section 6 reports the final comments and conclusions.

## 2 Background

### 2.1 Recommender systems

Recommender systems (RSs) are a powerful means of information analysis and filtering intended to provide appropriate recommendations about an item to a user of the system.

The entities with which a RS operates are users, items (services or objects that the system wants to recommend), and transaction that represents an interaction between the

system and the user. The most common form of transaction is represented by the rating, a user’s judgment about a specific item. The rating can be expressed in the ordinary form (very agree, agree, neutral, disagree, very disagree), numerically (five-point scale, from 1 to 5, ten-point scale, from 1 to 10, ...) and through a binary approach (I like/do not like) or unary (presence or lack of information).

Formally, the rating is defined as follows:

**Definition 1** Let  $U$  be the set of users and  $I$  the set of items. The rating function or the utility function is defined as the function  $r$  which to each pair of the domain  $U \times I$  associates the rating  $r_{ui} \in \mathbb{R}$ .

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

Since  $r$  is not defined for all pairs of the domain, one of the main purposes of the Recommender System is to deal with this problem in order to provide rating forecasts  $\hat{r}_{ui}$  for each element of the domain  $\forall (u, i) \notin r^{-1}(\mathbb{R})$  that does not belong to the preimage of the codomain.

#### 2.1.1 Classification of recommender systems

The ability to generate reliable rating forecasts distinguishes efficient Recommender Systems from inefficient ones, making this feature central in evaluating the techniques used.

RSs can be classified according to how the forecasts are generated. The following are the three most common strategies:

- **Content-Based RS:** this technique exploits the feature vector  $\mathbf{y} = (y_1, \dots, y_d) \in \mathbb{R}^d$  for item  $i$  (Content Analyzer De Gemmis et al. 2015) and create the feature vector  $\mathbf{x} = (x_1, \dots, x_d) \in \mathbb{R}^d$  for user preferences  $u$  (Profile Learner De Gemmis et al. 2015). To determine whether item  $i$  is suitable for user  $u$ , similarity metrics are exploited (De Gemmis et al. 2015) (Filtering Component De Gemmis et al. 2015) such as, for example, calculating the cosine of the angle between the vectors  $\mathbf{y}$  and  $\mathbf{x}$  representing item and user:

$$\cos(\mathbf{x}, \mathbf{y}) = \frac{\sum_{t=1}^d x_t y_t}{\sqrt{\sum_{t=1}^d x_t^2} \sqrt{\sum_{t=1}^d y_t^2}}$$

- **Collaborative Filtering RS:** this strategy aims to provide rating forecasts through the most appropriate opinions for a given user by exploiting the selection of information available (Colace et al. 2015). Collaborative systems are divided into two groups:

- **Memory-Based Collaborative Filtering:** users or items are divided into Neighborhoods in order to

generate rating predictions through similarity coefficients and known ratings (Koren et al. 2009);

- *Model-Based Collaborative Filtering*: a system model is obtained through matrix factorization (Koren et al. 2009) using known ratings. The most used decomposition techniques are:
  - Principal Component Analysis (Bokde et al. 2015);
  - Probabilistic Matrix Factorization (Bokde et al. 2015; Salakhutdinov and Mnih 2008);
  - Non-Negative Matrix Factorization (Bokde et al. 2015);
  - Singular Value Decomposition (Symeonidis and Zioupos 2016; Golub and Van Loan 2013; Comincioli 2010; Quarteroni et al. 2010);
- *Hybrid RS*: it combines two or more techniques to improve the ability of the system to make predictions. The techniques can be used separately and then combined or, finally, a single model is created to integrate the properties of the chosen strategy (Ricci et al. 2015).

### 2.1.2 The limits of recommender systems

The main problems facing Recommender Systems are:

- *Scalability*: the system capacity to manage the increase in the number of available data;
- *Sparsity (small number of known ratings)* which should not affect the quality of the forecasts carried out;
- *Cold Start*: the difficulty of the Recommender Systems to carry out forecasts about new users or items.

In the Table 1 (Bokde et al. 2015; Thorat et al. 2015) a detailed description of the advantages and disadvantages of the recommendation techniques described above is provided.

## 2.2 Context-aware recommender systems (CARS)

The purpose of Recommender Systems, as seen above, is to generate forecasts through the chosen strategy and information about users and items.

Over the years, new approaches to the development of RSs have also considered information related to the contingent situation that the system can use to predict rating values. This information is defined *contextual information*.

Contextual information can therefore be considered information that modifies the transaction mode of a Recommender System through the data obtained from the situation involving the system (user and item). Consequently, it changes the numerical value of the rating as contextual situations change.

In this way, the Context can be defined as the specific situation defined by the Recommender System’s contextual information.

The introduction of the Context within what are called Context-Aware Recommender Systems (CARS) changes the definition of the utility function (rating).

**Definition 2** Let  $U$  be the set of users,  $I$  the set of items and  $C$  the set of contextual variables. The rating function or utility function for CARS is defined as the function  $f$  which to each  $(u, i, c)$  of the domain  $U \times I \times C$  associates the evaluation  $f_{uic} \in \mathbb{R}$ .

$$f: (u, i, c) \in U \times I \times C \mapsto f(u, i, c) = f_{uic} \in \mathbb{R}$$

The introduction of CARS shows the importance of the acquisition of contextual information. These can be obtained in various ways, such as:

- *explicitly*: by questions to the user or by other means (Ricci et al. 2015);

**Table 1** Limits of recommendation techniques

Recommendation technique	Advantages	Limitations
<i>Content-based RS</i>	Easiness in suggesting new item Easiness of implementation	Cold Start ( new user) Diversity
<i>Collaborative Filtering RS</i>	Memory-Based Easiness of data updating Easiness of implementation	Cold start (new user /new item) Sparsity Scalability
	Model-Based Compares well with sparcity e scalability The resulting prediction performance is better	Cold start (new user/new item) Loss of information because of the use of factorization techniques
<i>Hybrid RS</i>	Provides better suggestions Overcomes the limitation of individual techniques	Complexity Model Development Cost

- *implicitly*: via mobile devices (location, temporal data, climatic data) or by changing the user's environmental conditions (Ricci et al. 2015).
- *through statistical or data mining techniques*.

### 2.2.1 Classification of contextual information

The acquisition of contextual information naturally leads to the need for their schematization for easier management. In addition, it is essential to identify the relevant contextual information. This information significantly modifies the ratings since it is precisely in these cases that the identification of the context allows the improvement of the performance of a Recommender System (Adomavicius et al. 2005).

A comprehensive classification is provided by Villegas et al. (2018). Preliminarily it is specified that the term "entity" indicates the main elements of the CARS: users and items. The classification of contextual information is as follows:

- *Individual Context*: contextual information associated with independent system entities that have common characteristics. This information can be further broken down into:
  - *Natural*: information associated with characteristics of an entity which are required without human intervention (for example, atmospheric conditions);
  - *Human*: information associated with the behavior or preferences of a specific user;
  - *Artificial*: information derived from human actions or technical processes describing the system entities;
  - *Groups of Entities*: information which brings together elements of the system through common features;
- *Location context*: information associated with the location of the entity, which is divided into:
  - *Physical*: information linked to geographical coordinates;
  - *Virtual*: information containing digital coordinates such as the IP address;
- *Time Context*: notions acquired on the physical time associated with the activity of an entity, which can be classified into:
  - *Defined*: when it is known time interval through a specified start and endpoint;
  - *Indefinite*: when information on the occurrence of the event is linked to the occurrence of others or is not specified the duration of the event;

- *Activity Context*: information related to activities performed by system entities. These concepts form the basis for predicting the entity's future preferences or activities itself;
- *Relational Context*: information related to relationships developed between entities' according to contingent circumstances. They are further subdivided into:
  - *Social*: information on interpersonal relationships between users, affiliations, associations, ...;
  - *Functional*: information on the use that some entities make of others.

### 2.2.2 Classification of context-aware recommender systems: integration of the context in RS

Once the methods of acquiring the contextual information and their classification are defined, it is necessary to use the same ones inside the recommendation process.

The possibilities of insertion of contextual information within the Recommender System are divided in three approaches:

- *Contextual Pre-Filtering*: information is applied to data prior to the development of recommendations. In this case, it can be assumed that the context is an application

$$c: (u, i) \in U \times I \mapsto c(u, i) = (\bar{u}, \bar{i}) \in U \times I$$

such that the utility function  $r$  of the Recommender System is calculated on the image obtained by  $c$

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

Therefore, the context-related information is exploited before proceeding with the recommendation method; this will allow only data related to the specific context to be considered for recommendation.

- *Contextual Post-Filtering*: the contextual information acquired is initially ignored by proceeding with a classic recommendation approach; it will be used for a final filtering of the recommendations obtained. The context can be imagined as a function that modifies the value of the utility function  $r$

$$c: r_{ui} \in \mathbb{R} \mapsto \bar{r}_{ui} \in \mathbb{R}$$

where the rating function is filtered through the information acquired;

- *Contextual Modelling*: contextual information is integrated into the recommendation process by creating an integrated model in the RS. In particular, contextual information is integrated into the utility function for the calculation of recommendations (Fig. 1).

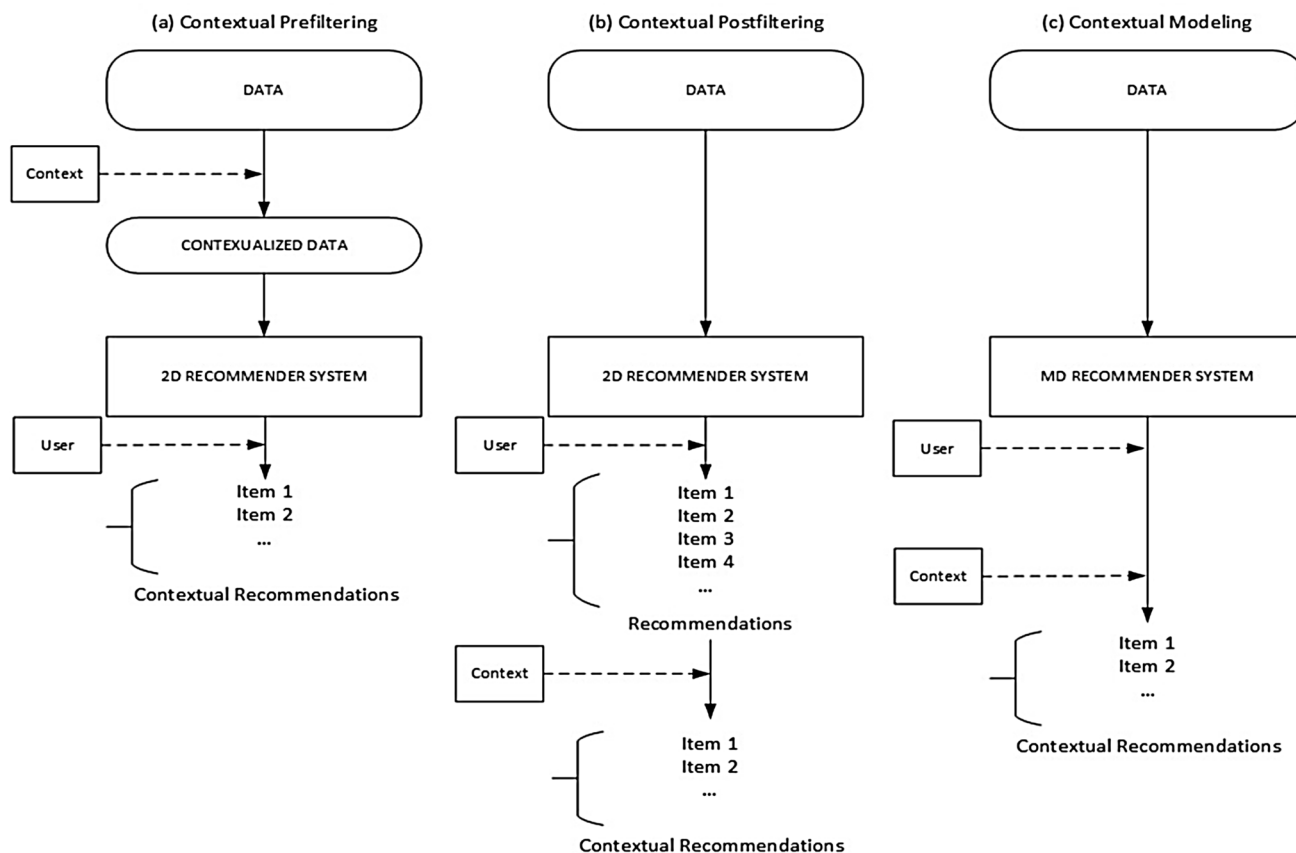


Fig. 1 Context-aware recommender systems approaches (Adomavicius and Tuzhilin 2011)

The structure for the integration of contextual information in the framework of the two most common recommendation strategies is presented below:

#### • Content-Based CARS

- *Pre-Filtering*: the context is integrated into the generation of user and item profiles. Contextual data is usually focused on user profiles. This approach involves the creation of a profile referring to the same entity of the System and referring to every possible assumption from the context;
- *Post-Filtering*: the context data are used to filter the recommendations generated by the Recommender System. This approach provides for an additional number of calculations proportional to the dimensions of the elements of the sets  $U \in I$ ;
- *Modelling*: contextual information is integrated into the computation of the cosine similarity between specific user and item. This process generally involves heuristic formulas.

#### • Collaborative Filtering CARS

- *Pre-Filtering*: the rating matrix may have columns or rows of dictionaries compared to the classical one. Additional columns or rows are associated with different preferences in different contexts. This approach reduces computational complexity but requires further effort from the system for data acquisition;
- *Post-Filtering*: as in the case of Content-Based CARS, contextual information is used to filter generated rating forecasts. This intuitively involves several additional calculations that will not be useful for the contingent purposes of the system;
- *Modelling*: this approach involves the integration of contextual data at the heart of the recommendation process, which is divided into:
  - *Heuristic-Based*: contextual information is integrated into the calculation of Neighborhoods (analogous to the case of Content-Based CARS with the cosine similarity);

- *Model-Based*: tensor factorization is exploited by integrating contextual data into additional dimensions. This procedure aims to find a numerical structure of the problem, which includes the context.

In addition to these classification groups, there are techniques, such as Context Aware Matrix Factorization (CAMF), classified as Modelling that integrate contextual information within the calculation of forecast obtained through the matrix factorization. The CAMF technique will be presented in deep below.

### 3 Related works

Over the years, the study of Recommender System and Context-Aware Recommender Systems gave rise to numerous operational strategies, which eventually led to classifications seen before. In order to classify the recommendation strategies, it is possible to find, in the scientific literature, several surveys that focus on various aspects of the strategies treated. In particular, Adomavicius and Tuzhilin (2005) provides an overview of the Recommender Systems presented in Sect. 2.1 and gives interesting hints for the future developments of RS, including the evaluation of contextual information. Adomavicius et al. (2011) provides a comprehensive overview of the definition of context and its role within Recommendation Systems. This work provides a clear and in-depth overview of the concept of “context” and its possible application by elaborating on the concepts presented in Sect. 2.2 of this paper. It also describes CARS applied in various fields such as, for example, travel guides, music, mobile information search. Champiri et al. (2015) conduct extensive and structured research about Context-Aware Recommender Systems in the context of digital libraries. Particularly in the paper described in detail the phase of research, study, and construction of the paper according to the methodology introduced by Kitchenham and Charters (2007). The main purpose is to determine the contextual information used the most, the most exploited recommendation approaches, and the researchers’ awareness of applying context to recommendations in academic settings. Haruna

et al. (2017) present an extensive survey of Context-Aware Recommender Systems presenting the various domains in which they have been applied and metrics to evaluate them. The contextual information considered in each application domain is also studied. Finally, Pavlidis (2019) also provides an overview of Recommender Systems with a focus on applications in the field of Cultural Heritage, with a particular focus on museums. The approaches covered include both classical RSs and Context-Aware. In addition, the author provides some considerations about the works analyzed and the future potential of Recommender Systems in the domain of Cultural Heritage. Unlike the surveys just mentioned, the purpose of this paper is to provide an overview of the state of the art of CARS in the domain of Cultural Heritage only. Moreover, it is specified that Group Recommender Systems will not be treated because these techniques need an appropriate and in-depth study that will not be treated in this paper. Obviously such techniques are widely used, as proven, for example, in Nguyen and Ricci (2017), Baltrunas et al. (2010), Cao et al. (2018), Huang et al. (2020). Some challenging strategies will be shown below (Table 2).

#### 3.1 Content-based

The strategies within the Content-based CARS are presented below. For greater clarity, a paragraph will be devoted to each strategy to outline further the methodologies analyzed.

An interesting proposal of a Content-Based technique is proposed by LOOKER Missaoui et al. (2019). This work proposes a pre-filtering strategy for introducing contextual information: a mobile Recommender System aimed to provide tourism and travel-related services.

This technique consists of two modules:

- *Spatio-Temporal Filtering Module*, which aims to select only the relevant items based on contextual information related to time and physical location;
- *Content-Based Filtering Module*, which provides rating forecasts. In turn it is divided into three main components:
  - Multi-layer User Profile: through Statistical language modeling (Ponte and Croft 1998) the comments of

**Table 2** Content-based approach: summary table of techniques and contextual information

Content-based approach	Contextual information			
	Pre-filtering	Context modelling	Post-filtering	
LOOKER Missaoui et al. (2019)	X			Location; Time
Hong et al. (2009)		X		Activity; Human; Time
Shin et al. (2009)		X		Time
Colombo-Mendoza et al. (2015)	X		X	Location; Time

each user, within the categories Tourism-services set, are analyzed. In particular, for each category  $c$ ,  $R_c = \{r_i, i = 1, \dots, \text{tot}_c\}$  is the set of positive user reviews about the category under consideration, the component  $\theta_c$  of the user profile  $\theta$  is calculated as follows:

$$\theta_c = \frac{1}{|R_c|} \sum_{r_i \in R_c} P(w|r_i)$$

where  $w$  is a word present in a subset of  $R_c$  and  $P(w|r_i)$  is estimated by the Dirichlet prior smoothing (Zhai and Lafferty 2004);

- TR-Services Profile: the service profiles are generated in a similar way as for users. Defined the set  $R_s$  of positive reviews of users about the item  $s$ , the component  $\omega_s$ , associated with the corresponding category, of the  $\omega$  profile is obtained as follows:

$$\omega_s = \frac{1}{|R_s|} \sum_{r_j \in R_s} P(w|r_j)$$

- Content-Based Filtering Algorithm: rating forecasts for each user  $u$ , service  $s$  and category  $c$  are calculated using the following formula:

$$\bar{r}_{u,s,c} = \frac{1}{D_{KL}(\theta_c || \omega_s)}$$

where  $D_{KL}(\theta_c || \omega_s)$  is the Kullback-Leibler divergence (Missaoui et al. 2019).

As part of the Content-Based Recommender techniques, Hong et al. (2009) provide a different integration of contextual information proposing a framework based on a decision tree algorithm to insert the context through modeling strategy. The method is developed in 4 stages:

1. Data Gathering Layer: the user profile is generated, and the services are selected. The raw context information are collected through context wrappers;
2. Context Management Layer: context aggregator collects contextual information in a vector then transformed into a high-level context by context inference agent. The user profile and the selected services of the first phase united to the high-level context form the context history further elaborated through filtering agent;
3. Preference Management Layer: the decision tree algorithm is used to extract user preferences for each service. The association agent deduces the association rules for the selected services;
4. Application Layer: personalized services are provided.

Another example of Content-Based Recommender System is provided by Shin et al. (2009), which presents a Context-Modelling integration. This approach develops as follows:

1. Calculation of vectors integrated with the context:
  - $u_i = \langle uc_{i1}, \dots, uc_{ij}, \dots, uc_{iq} \rangle \in \mathbb{R}^q$  correlation vector between the user  $i$  and the contextual information  $q$  considered;
  - $s_k = \langle sc_{k1}, \dots, sc_{kj}, \dots, sc_{kq} \rangle \in \mathbb{R}^q$  correlation vector between the item  $k$  and the contextual information  $q$  considered;
  - $h_i = (\langle hcc_{i1}, \dots, hcc_{iq} \rangle, \langle hcd_{i1}, \dots, hcd_{iq} \rangle) \in \mathbb{R}^{2q}$  vector that stores the contextual information.
2. Are computed:
  - $f_{UH}(i, j) = \frac{\sum_{x=1}^q |uc_{ix}| \times |hcc_{jx}|}{\sqrt{\sum_{x=1}^q |uc_{ix}|^2} \sqrt{\sum_{x=1}^q |hcc_{jx}|^2}}$  matching between user  $i$  and context history  $j$
  - $f_{HI}(j, k) = \frac{\sum_{x=1}^q |hcd_{jx}| \times |sc_{kx}|}{\sqrt{\sum_{x=1}^q |hcd_{jx}|^2} \sqrt{\sum_{x=1}^q |sc_{kx}|^2}}$  matching between context history  $j$  and item  $k$ ;

3. Calculation of the rating forecast with the integrated time context:

$$f_{UI}(i, k) = \sum_{j=1}^l f_{UH}(i, j) f_{HI}(j, k)$$

A Content-Based recommendation model with hybrid integration Pre-filtering and Post-filtering of contextual information (location, crowd, time) is proposed by Colombo-Mendoza et al. (2015). The context element “Location” represents the distance between the user and the target considered, the context element “Crowd” considers the preferences of the active user and rating forecasts made on other users, the context element “Time” divided as time available to the user and associated with the user profile is the time needed to access the service (evaluated on three modes of transport). The Recommender System proposed by Colombo-M et al can be summarized as follow:

1. Update of the user profile;
2. Pre-filtering associated with Time (step 1): the items that are not accessible are discarded;
3. Pre-filtering associated with the Location: items too far away are discarded through a specific Distance-Decay function;
4. Pre-filtering associated with Time (step 2): items are discarded according to information obtained from the previous point;
5. Calculation of Similarity;

6. Collection of the Information associated with the user in consideration;
7. Calculation of the rating forecast;
8. Post-filtering associated with Location and Crowd: calculated ratings are changed based on contextual information related to the context obtained by considering Crowd and Location.

### 3.2 Collaborative-filtering

In this section, the techniques for the introduction of the context to the Collaborative Filtering Recommender System are analyzed.

Also in this case the methodologies analyzed will be shown through a division into paragraphs (Tables 3, 4, 5).

The work of Baltrunas and Ricci (2009), Baltrunas and Ricci (2014) fits into the Pre-filtering integration modes. This strategy in a system of  $m$  users and  $n$  items apply to the columns of the  $R \in \mathbb{R}^{m \times n}$  rating matrix and changes it according to the changes in preferences in the different

contexts. In particular, the rating column is split into two columns: the set of values assumed by the context  $C$  is given, the column of the rating matrix is split over the value  $c_j \in C$ , which mostly alters the known ratings. This results in a matrix of contextualized ratings  $\hat{R} \in \mathbb{R}^{m \times (n+l)}$  with  $l$  number of columns that have suffered the split. At this point, it is possible to proceed by using a classic Recommender System. The ability to split an item is limited to only two columns in order to preserve linear complexity to the method described. In fact, splitting the column of an item, as described, can generate two more columns and not a greater number.

Similarly to the procedure described above, it is possible to perform splitting either on the rating matrix's row using the User-Splitting technique (Said et al. 2011; Zheng et al. 2013) or it is possible to operate both on the columns and the rows of the matrix according to the User-Item Splitting technique (Zheng et al. 2013).

In particular, the User Splitting technique is used to break down user ratings based on the user's contextual information

**Table 3** Collaborative filtering: summary table of techniques and contextual information

Collaborative filtering					
	Technique	Pre-filtering	Context modeling	Post-filtering	Contextual information
Item-splitting Baltrunas and Ricci (2014)	Model-Based	X			Social; Time
User splitting Said et al. (2011) Baltrunas and Ricci (2014)	Model-Based	X			Location
User-item splitting Zheng et al. (2013) Baltrunas and Ricci (2014)	Model-Based	X			Location; Time
Karatzoglou et al. (2010) Baltrunas and Ricci (2014)	Model-Based		X		Time; Human; Social
Liu et al. (2013) Baltrunas and Ricci (2014)	Model-Based		X		Social
Koren (2009)	Model-Based		X		Time
Baltrunas et al. (2011b)	Model-Based		X		Location; Depends on Dataset
Xu et al. (2015)	Memory-based			X	Location; Time

**Table 4** Number and typology of contextual information in the dataset described

Dataset	#contexts	Contextual Information
<i>LDOS-CoMoDa</i>	12	Time, Day Time, Season, Location, Weather, Social, Emotion, Dominant Emotion, Mood, Physical, Decision, Interaction
<i>Frappe</i>	6	Daytime, Weekday, Homework, Weather, Country, City
<i>DePaul Movie</i>	3	Time, Location, Companion
<i>InCarMusic</i>	8	Driving Style, Landscape, Mood, Natural Phenomena, Road Type, Sleepiness, Traffic Condition, Weather
<i>TripAdvisor</i>	1	Trip Type
<i>STS</i>	14	Distance, Time Available, Temperature, Crowdedness, Knowledge of Surrounding, Season, Budget, Daytime, Weather, Companion, Mood, Weekday, Travel Goal, Means of transport
<i>TijuanaRestaurant</i>	2	Time, Location
<i>JapanRestaurant</i>	14	Month, Hour, Weekday Area Type, Budget, Holiday # male, #female, lowest age, highest age, relation, status Weather, Temperature



**Table 5** Summary table of techniques and contextual information applied in cultural heritage field

Approach	Technique	Contextual pre filtering	Con-textual modeling	Contextual post filter-ing	Contextual information
Context evolution system (CES) Chianese and Piccialli (2016)	Content Based			X	Location
SMART MUSEUM Ruotsalo et al. (2013)	Content Based			X	Location - Information provided by user
Bartolini et al Bartolini et al. (2016)	Content Based	X		X	Location - Environmental Situations - Weather
Turist@ Batet et al. (2012)	Hybrid	X			Location
del Carmen Rodríguez Hernandez et al del Carmen et al. (2017)	Collaborative Filtering	X			Location - Mood - Temperature - Noise level - # of people in a room
Chat-bot Casillo et al. (2020)	Content Based			X	All possible contexts

acquired through known ratings. This technique treats the individual user as a set of different users labeled based on online ratings’ contextual scope.

User Splitting and User-Item Splitting also introduce contextual information with a Pre-filtering strategy.

Karatzoglou et al. (2010) expand Model-Based strategies based on machine learning (Anastasiu et al. 2016; Zhao et al. 2019; Wang et al. 2019) to the tensor case. In this way, it is obtained a strategy of the introduction of contextual information Modelling.

The decomposition used is HOSVD (De Lathauwer et al. 2000), which exploits the singular value factorization technique to multidimensional matrices. Given  $n$  users,  $m$  items and  $c$  contexts, defined the function  $l : (f, y) \in \mathbb{R} \times \mathbb{Y} \mapsto l(f, y) \in \mathbb{R}$  estimating the error between the known  $\mathbb{Y}$  data and the forecast data, algorithm 1, present in the reference (Karatzoglou et al. 2010), is used to generate the contextual rating forecasts.

**Algorithm 1** Tensor Factorization

```

1: Input:  $Y$  tensor of known ratings,  $d \in \mathbb{R}$ 
2: Initializing of  $U \in \mathbb{R}^{n \times d_U}$ ,  $M \in \mathbb{R}^{m \times d_M}$ ,  $C \in \mathbb{R}^{c \times d_C}$ 
3: Initializing of  $S \in \mathbb{R}^{d_U \times d_M \times d_C}$ 
4: Fixing of  $t = t_0$ 
5: while  $(i, j, k)$  indices of tensor  $Y$  do
6:    $\eta \leftarrow \frac{1}{\sqrt{t}}$ 
7:    $t \leftarrow t + 1$ 
8:    $F_{ijk} = S \times_U U_{i^*} \times_M M_{j^*} \times_C C_{k^*}$ 
9:    $U_{i^*} \leftarrow U_{i^*} - \eta \lambda_U U_{i^*} - \eta \partial_{U_{i^*}} l(F_{ijk}, Y_{ijk})$ 
10:   $M_{j^*} \leftarrow M_{j^*} - \eta \lambda_M M_{j^*} - \eta \partial_{M_{j^*}} l(F_{ijk}, Y_{ijk})$ 
11:   $C_{k^*} \leftarrow C_{k^*} - \eta \lambda_C C_{k^*} - \eta \partial_{C_{k^*}} l(F_{ijk}, Y_{ijk})$ 
12:   $S \leftarrow S - \eta \lambda_S S - \eta \partial_S l(F_{ijk}, Y_{ijk})$ 
13: Output:  $U, M, C, S$ 

```

The constants  $\lambda_U, \lambda_M, \lambda_C, \lambda_S$  have been fixed by numerical experiments and  $U_{i^*}, M_{j^*}, C_{k^*}$  represent the columns of the homonymous matrices evaluated for the iteration indices current.

Another Context Modelling approach for Model-Based Collaborative Filtering is provided by Liu et al. (2013). This procedure extends the Model-Based techniques based on Probabilistic Matrix Factorization (Bokde et al. 2015; Salakhutdinov and Mnih 2008) through the context integration on latent factors associated to users and items. In this way, two strategies are born:

- *Bayesian Probabilistic Matrix Factorization with Social Relations (BPMFSR)*: the calculation of latent factors associated with users are integrated with personalized hyperparameters (unlike the matrix case) and the parameters associated with social context information are inserted;
- *Bayesian Probabilistic Matrix Factorization with Social Relations and Item Contents (BPMFSRIC)*: such strategy has the BPMFSR’s measures on users and the further integration on the calculation of the latent factors associated to the personalized hyperparameter items and the association of the latter on the basis of the social context information (tags) and the specific properties.

The last example of Model-Based Collaborative Filtering with Integration Context Modelling is presented by Koren (2009). To integrate the temporal context into the Recommender System, Koren develops various analysis models.

Let  $b_{ui}$  be defined as the forecast about the user  $u$  and the item  $i$ , the models developed are the following:

- *Static*:  $b_{ui}(t) = \mu + b_u + b_i$   
where  $\mu$  is the ratings average of known items,  $b_u$  is the bias of user and  $b_i$  is the bias of item;
- *Mov*:  $b_{ui}(t) = \mu + b_u + b_i + b_{i, Bin(t)}$   
where  $b_{i, Bin(t)}$  is the bias of item on time interval  $Bin(t)$ ;

- *Linear*:  $b_{ui}(t) = \mu + b_u + \alpha_u \text{dev}_u(t) + b_i + b_{i, \text{Bin}(t)}$   
the undefined elements of the sum are:
  - $\text{dev}_u = \text{sign}(t - t_0) |t - t_0|^\beta$   
where  $\beta$  is a constant empirically determined;
  - $\alpha_u$  is the coefficient associated with the user  $u$ ;
- *Spline*:  $b_{ui}(t) = \mu + b_u + \frac{\sum_{l=1}^{k_n} \exp\{-\gamma|t-t_l^n|\} b_{il}^n}{\sum_{l=1}^{k_n} \exp\{-\gamma|t-t_l^n|\}} + b_i + b_{i, \text{Bin}(t)}$   
where  $k_n$  is the number of control time points and  $\gamma$  is a constant empirically evaluated;
- *Linear with daily effect on user bias*:  
 $b_{ui}(t) = \mu + b_u + \alpha_u \text{dev}_u(t) + b_{u,t} + b_i + b_{i, \text{Bin}(t)}$   
where  $b_{u,t}$  is the user  $u$  bias and has daily variability;
- *Spline with daily effect on user bias*:

$$b_{ui}(t) = \mu + b_u + \frac{\sum_{l=1}^{k_n} \exp\{-\gamma|t-t_l^n|\} b_{il}^n}{\sum_{l=1}^{k_n} \exp\{-\gamma|t-t_l^n|\}} + b_{u,t} + b_i + b_{i, \text{Bin}(t)}$$

Defined the set  $K$  of triads  $(u, i, t)$  for which the rating  $r_{ui}(t)$  is known and, chosen the model, Context-Aware strategy proceeds minimizing the error function:

$$\min \sum_{(u,i,t) \in K} (r_{ui}(t) - b_{ui}(t))^2 + \lambda \left( \|b_{ui}^{(t)}\|_E \right)^2$$

The case of Memory-Based Collaborative Filtering with Post-filtering integration of contextual information is presented in Xu et al. (2015). This work aims to build a Context-Aware Recommender System for tourist purposes through geolocation of user photos.

Both the geographical coordinates of a given location  $l$ ,  $v = (l, u, t)$  the visit associated with location  $l$ , user  $u$ , time  $t$  and the topic based context-aware query  $Q = (u_p, s, w, d)$  associated with the target user  $u_p$ , the season  $s$ , the weather  $w$  and the target city  $d$  relative to  $u_p$  which returns in output a list of places associated with the city  $d$  are given. Assuming a set of geo-referenced photos are provided as input.

$\{P_u : P_u$  list of user  $u$  geotagged photos $\}$ ,

the technique is developed as follows:

1. Construction of tourist locations profiles containing contextual information;
2. Construction of the location database  $LDB = \{l_1, \dots, l_n\}$  with  
 $l_i = \{v_i, \text{pop}(s), \text{pop}(w)\} \forall i = 1, \dots, n$

The elements that make  $l_i$  are the visits  $v_i$  associated with the locations  $l_i$ , the contextual information  $\text{pop}(s)$  about the most popular season for location  $l_i$ , the

- contextual information  $\text{pop}(w)$  about the most popular weather conditions of the location  $l_i$ ;
3. Construction of the user-location matrix  $M = (M_{ul})$  such that  $M_{ul}$  indicates the number of times that the user  $u$  visited the location  $l$ ;
4. Through the information of the user-location matrix  $M$  is built the travel history of each user;
5. Calculation of the similarity between users;
6. Calculation of rating forecasts through
  - (a) Recovery of  $N$  users closest to the specific user;
  - (b) Calculation of the rating forecast for each location;
  - (c) Filtering information obtained through contextual information;
  - (d) Construction of list of  $m$  locations with best rating forecasts.

*Context-Aware Matrix Factorization* (Baltrunas et al. 2011b)

This technique aims to extend matrix factorization models through the analysis of deviations (defined baselines for contextual conditions) determined by contextual parameters.

The calculation of the contextual rating forecast takes place after having derived through a rating matrix factorization technique the  $p_i, q_j \in \mathbb{R}^d$  vectors associated with the  $i$ -th user and  $j$ -th item where  $d$  is the number of latent factors considered.

It is possible to calculate the forecast  $\hat{r}_{ijc_1 \dots c_k}$  of the  $k$  dimensional nodes associated to evaluated contexts as follows:

$$\hat{r}_{ijc_1 \dots c_k} = \bar{r}_j + p_i \cdot q_j + b_i + \sum_{z=1}^k B_{jz c_z}$$

where  $\bar{r}_j$  is the average of ratings of item  $j$ ,  $b_i$  is the baseline parameter for user  $i$ ,  $B_{jz c_z}$  is the parameter that represents the interaction between item  $j$  and contextual condition  $c_z$ .

Three techniques based on the same idea are presented in this reference:

- *CAMF-C*: a single parameter is analyzed for each contextual condition which generates deviations on ratings. In this method you have  $k = 1$ , then you have an only  $B_{j c_1}$ ;
- *CAMF-CI*: in the analysis of deviations are also considered items so to determine that, and have on the latter. Such an approach provides the increase in the number of parameters to be analysed. This approach provides for the  $B_{j z c_z}$  rating for each item and contextual condition;
- *CAMF-CC*: items are placed in categories in order to reduce the number of parameters of the CAMF-CI approach losing only part of the specialization obtained on the calculated deviation. Items of the same category have the same  $B_{j z c_z}$  parameter in common.

The calculation of the training values is made through the use of the Stochastic Gradient Descent on the learning procedure.

$$\min_{p_*, q_*, b_*, B_*} \sum_{r \in R} \left[ (r_{ijc_1 \dots c_k} - \hat{r}_{ijc_1 \dots c_k})^2 + \lambda \left( b_i^2 + \|p_i\|^2 + \|q_j\|^2 + \sum_{z=1}^k \sum_{c_z=1}^{v_z} B_{jzc_z}^2 \right) \right]$$

where  $c_z = 1, \dots, v_z$  or  $c_z = 0$  if it is unknown,  $z = 1, \dots, k$ .

## 4 CARS evaluation: dataset and evaluation metrics

The ability to analyze the effectiveness of the recommendations given in various contexts is a central issue. Based on the chosen method of recommendation and the intended purposes, it will be necessary to initially study the system's behavior developed through a dataset of data collected. In this first case, a study will be carried out on the accuracy of the recommendations provided. However, when switching to tests with real users, it will need to study additional properties of the suggestions provided.

For this purpose, datasets are presented below that allow testing on CARS and, subsequently, usable valuation metrics.

### 4.1 Datasets

The possibility to use contextual data to analyze the performance of the various methods is crucial.

In this regard, a useful tool is *CARSKit* (Zheng et al. 2015). This software is freeware and developed through the Java programming language. It is also convenient to use thanks to the presence of a guide associated (Zheng 2015), although a limited number of datasets may be used (Raza and Ding 2019).

Some of the datasets available for CARS analysis are the following:

- *LDOS-CoMoDa* (Kořir et al. 2013; Ilarri et al. 2018): movie dataset including 2296 ratings divided on 12 contextual dimensions. The contextual information considered are time, day type, season, location, weather, social, emotion (and emotion, dominant emotion), mood, physical, decision, interaction;
- *Frappe Dataset* (Baltrunas et al. 2015; Ilarri et al. 2018): dataset developed under the develop of a context-aware mobile app. Contains 96203 ratings of 957 users about 4082 items. The contextual information

analyzed are 6: daytime, weekday, homework, weather, country, city;

- *DePaul Movie* (Zheng et al. 2015; Ilarri et al. 2018): database consisting of ratings released by students in different moments. It contains 5043 evaluations, the users are 97, the items considered are 79. The contextual information considered are 3: time (weekend, weekday), location (home, cinema), companion (alone, partner, family);
- *InCarMusic* (Baltrunas et al. 2011a; Ilarri et al. 2018): database developed to recommend music to the passengers of a vehicle. It contains 4012 ratings of 42 users about 139 items. The contextual information considered are 8: driving style (relaxed, sport), landscape (coastline, country side, mountains/hills, urban), mood (active, happy, lazy, sad), natural phenomena (afternoon, day time, morning, night), road type (city, highway, serpentine), sleepiness (awake, sleepy), traffic conditions (free road, lots of cars, traffic jam), weather (cloudy, rainy, snowing, sunny);
- *TripAdvisor* (Ilarri et al. 2018): dataset related to the problem of hotel recommendations to users with the assessment of a single contextual dimension. This provides information on the type of user travel (family, couples, business, travel only, friends). There are also the user and item features associated with geographical positions. The dataset presents the following two versions:

- TripAdvisor v1 (Zheng et al. 2012): contains 4669 ratings of about 1202 users and 1890 items;
- TripAdvisor v2 (Zheng et al. 2014): contains 14175 ratings of about 2731 users and 2269 items.

There is also an additional version containing 28350 ratings of about 2371 users and 2269 items;

- *STS* (Braunhofer et al. 2014; Ilarri et al. 2018): dataset containing the data collected through the South Tyrol Suggest app that aims to recommend points of interest to users. 2534 ratings are available from 325 users on 249 items. The main peculiarity is the number of contextual dimensions; in fact, the context is obtained by combining 14 different types of contextual information. The dimensions are: distance, time available, temperature, crowdedness, knowledge of surroundings, season, budget, daytime, weather, companion, mood, weekday, travel goal, means of transport;
- *TijuanaRestaurant* (Ramirez-Garcia and García-Valdez 2014): this dataset is composed by 1422 ratings of 50 users on 40 items. The users involved had to answer eight questions aimed at both the user's characterization and the evaluation of the items available. These ratings are provided both in contextless and contextual

environments. The contextual dimensions evaluated are 2: time (weekday, weekend) and location (school, home, work);

- *JapanRestaurant* (Oku et al. 2006): dataset in which 938 restaurants are classified. The classification is based on 4 categories: “is equipped with”, “has services of”, “recommended for”, ‘environment includes”. There are 14 contextual dimensions that are divided into 4 groups:

- Time: month, hour, weekday;
- Schedule: Area type, budget, holiday;
- Partner: number of male, number of female, lowest age, highest age, relation, status;
- External Factor: weather, temperature;

## 4.2 Evaluations metrics

Evaluation metrics are a key tool for the analysis of the goodness of a Recommender Systems. These are essential for selecting the appropriate recommendation model in a specific field and comparing the different recommendation techniques and their effectiveness (Shani and Gunawardana 2011).

Several aspects of the recommendation model can be assessed based on the evaluation metric used (Kane 2018).

In the following, what will be asserted in general for the Recommender System will also apply to Context-Aware Recommender Systems.

To test the effectiveness of an RS (or in the specific case of a CARS), you must first decide the type of experiment to be carried out.

The simplest and least expensive to implement is the offline experiment (Shani and Gunawardana 2011; Gunawardana and Shani 2009) that exploits a dataset to simulate user’s behavior interacting with the Recommender System. The behavior is simulated by storing the actions of the user over time labeled through the time-stamp.

The dataset used can be divided into training sets and test sets. For this purpose, one of the most widely used techniques is k-fold cross validation (Rodriguez et al. 2009), which allows randomly partitioning the data set available. The error will be assessed on each of the k partitions generated by using the remaining parts, in turn, as a training set (Kane 2018).

Other types of experiments are user studies (Shani and Gunawardana 2011) and online evaluation (Shani and Gunawardana 2011; Gunawardana and Shani 2009). The first consists of recruiting a group of users who must perform some tasks related to interaction with the Recommender System. The second point to make the RS interact with a greater number of real users to assess the impact and the effectiveness of the recommendations provided

by developing an online testing system (Shani and Gunawardana 2011; Kohavi et al. 2007).

Once the experiment modalities have been determined, it is necessary to decide the RS properties to be evaluated.

*Accuracy Measures* It represents the most used measure in offline experiments and aims to assess the accuracy of the Recommender System in suggesting items to the user.  $T = \{(u, i) : \exists r_{ui}\}$  is the set of user-item pairs on which the test phase is carried out and  $\hat{r}_{ui}$  is the prediction made by the System about the pair  $(u, i)$ . The measurement of the accuracy of the Recommender System in the prediction of ratings can be evaluated through:

- *Mean Absolute Error* (MAE) (Shani and Gunawardana 2011; Kane 2018; Gunawardana and Shani 2009; Herlocker et al. 2004): the name itself shows that the accuracy is estimated by the mean of the absolute value of the errors.

$$\text{MAE} = \frac{1}{|T|} \sum_{(u,i) \in T} (\hat{r}_{ui} - r_{ui})$$

- *Root Mean Square Error* (RMSE) (Shani and Gunawardana 2011; Kane 2018; Gunawardana and Shani 2009; Herlocker et al. 2004): allows to calculate the accuracy of the Recommender System provided by penalizing more than the MAE bigger errors.

$$\text{RMSE} = \sqrt{\frac{1}{|T|} \sum_{(u,i) \in T} (\hat{r}_{ui} - r_{ui})^2}$$

A different measure to be implemented to assess the accuracy of the Recommender System is the one that aims to understand if you can suggest items that the user could use (Shani and Gunawardana 2011; Gunawardana and Shani 2009). A valid method in this regard is the measurement of precision, recall and false positive rate. These, note the amounts of elements true positive ( $tp$ ), false positive ( $fp$ ), true negative ( $tn$ ) e false negative ( $fn$ ) shall be calculated as follows:

$$\text{Precision} = \frac{tp}{tp + fp}$$

$$\text{Recall} = \frac{tp}{tp + fn}$$

$$\text{False Positive Rate} = \frac{fp}{fp + tn}$$

Other possible measures are the  $F$ -measure (Shani and Gunawardana 2011; Van Rijsbergen 1979) and the Area Under the ROC Curve (Shani and Gunawardana 2011; Hand and Till 2001).

Within the Top-N recommendation list you can also evaluate how many correct items are identified in the list

provided to the user. Defining hit each items identified in an exact manner, we present the following assessment methods:

- *Hit Rate* (Kane 2018): let be  $h_u$  the number of hits associated with user  $u \in U$ , the hit rate is calculated as follow:

$$\text{HIT RATE} = \frac{1}{|U|} \sum_{u \in U} h_u$$

- *Average Reciprocal Hit Rate (ARHR)* (Kane 2018): let be  $n$  the total number of hits and  $r_i$   $i = 1, \dots, n$  the rank of  $i$ -th hit, the average reciprocal hit rate is calculated as follow:

$$\text{ARHR} = \frac{1}{n} \sum_{i=1}^n \frac{1}{r_i}$$

In this way the hits far from fist positions of the Top-N list are penalized.

- *Cumulative Hit Rate* (Kane 2018): the hits whose forecast is below a set threshold are excluded. In this way items that would not be appreciated by user are not considered by the evaluation.

Normalized Distance Based Performance (NDBP) (Shani and Gunawardana 2011; Yao 1995), and the R-score metric (CarlKadie 1998; Shani and Gunawardana 2011). In particular, the latter makes it possible to assess the usefulness of the recommendations made.

*Other Possible Measurements* There are many other possible properties of an analyzable Recommender System.

The coverage aims to measure the percentage of recommendable items. A good value may indicate the system's ability to quickly suggest new items to appropriate users (Kane 2018).

The reliability of the recommendations provided (confidence), the trust the user has in the RS (trust), the ability to balance between the recommendations of known items and items not known to the user (novelty), the serendipity (Wang et al. 2018), the diversity of recommended items and the stability of the recommendations provided in the presence of false information (robustness) can also be measured.

Finally, we can also evaluate:

- *Adaptivity*: the ability of the Recommender System to adapt to changes in the user behaviour or changes in the value of items over time;
- *Scalability*: the ability of the recommendation algorithm to adapt the increase in data to be managed with reasonable memory occupancy and slowdown.

These measures represent an alternative to accuracy as they aim to evaluate different aspects. In particular, serendipity, diversity, novelty, and coverage tend to guarantee the efficiency of a Recommendation System. Serendipity guarantees surprising recommendations (Kaminskas and Bridge 2016), diversity allows for the suggestion of various items, novelty ensures the possibility of recommending unfamiliar items and is closely related to serendipity (Kaminskas and Bridge 2016), coverage guarantees the possibility of guaranteeing suggestions on a wide choice of items available to the System. Therefore, it is clear that in general, and in the field of Cultural Heritage in particular, Recommendation Systems must not only guarantee the accuracy of the suggestions provided but must aspire to a good level of serendipity, diversity, coverage, and novelty. In the Cultural Heritage field, it is crucial to guarantee serendipity, which can increase the user's satisfaction who takes advantage of the suggestions provided. On the other hand, it can provide diversified options to the user, avoiding thematic monotony, especially if the objective of the RS is to suggest paths. On the other hand, novelty is a separate issue, since in the world of Cultural Heritage, it is complex to give relevance to little-known sites, and ad hoc recommendation techniques are needed (Casillo et al. 2021a, b).

## 5 CARS in cultural heritage

Below will be shown applications in which the Context-Aware Recommender Systems are central to providing appropriate suggestions for experiences in cultural heritage.

The Context Evolution System (CES), presented in Chinese and Piccialli (2016), provides an initial model for the application of CARS within the Cultural Heritage field. Specifically, an application introduced for an exhibition called "the Beauty or the Truth" in Naples is presented as a case study.

The main elements of this approach are:

- *Service Engine*: it is the main component of architecture and consists of three modules.
  - *Events Detector*, which aims at event recognition and activation of the Context Switching Computation Module;
  - *Context Switching Computation Module*, which aims at identifying possible changes in context and selection of Contextual Data Views, containing relevant data for the specific situation, and Basket of Ser-

- vices, content of the services adapted to the contextual situation;
- Visiting paths generation, which aims to determine a path to the user when visiting items;
- Services Deliverer: component dedicated to the analysis of the status of user-items systems. After the analysis of the nearby items through the Events Detector and after having determined the contextual context through the Context Switching Computation Module, the Deliverer Services allows you to transmit information adapted to the user by means of the multimedia guide;
- Knowledge base and user LOG: data management element for propose items appropriate to the user preferences and to the storage of the user behavior;
- Context manager: component suitable for the analysis of information to determine the contextual state.

This architecture fits into an environment capable of assessing specific situations contextual. This is possible through a set of factors that are underlying the concept of Smart Cities Schaffers et al. (2011). In particular, the concept of Single is developed Smart Space  $S^3$  Chianese and Piccialli (2016).

The fundamental phase of the analysis of the contextual situation on  $S^3$  is the study of Context Evolution Graph  $CEG = (C; \Sigma; l_c; l_s)$  consisting of the following components::

- $C = \{c_1, \dots, c_n\}$  set of contextual variables;
- $\Sigma \subseteq C \times C$  set of graph edges;
- $l_c : c \in C \mapsto (v, S)$  function that links each contextual variable with the associated contextual data  $v \in V$  and the Basket of Services set  $S$ ;
- $l_s : t \in \Sigma \mapsto e \in E$  functions thath links each edge of the graph with the specific event  $e$ .

Another interesting approach built around visiting a museum is provided by SMARTMUSEUM Ruotsalo et al. (2013). The application described has features Content-Based with the introduction of contextual information of Pre-filtering type.

The architecture presented is composed of four main components:

- Metadata Service, which stores the data obtained through the Web;
- Context Service, aimed at conceptual mapping data through the use of Ontologies (Gašević et al. 2006). The user can provide the collected data through the application or acquired through the sensors in the museum. In this way, the information about the user's location and about the item sites in its vicinity is acquired.
- User Profile Service, a component that generates the user profile through contextual information and ontolo-

gies. Besides, the known feedbacks that are relevant in the contextual context modeled through a probabilistic approach;

- Filtering Service aims to index items through information acquired by the Web and subsequently carry out the recommendation process.

In Bartolini et al. (2016) is reported an alternative work involving a Content-Based recommendation technique with dual filtering phase of the context. This is introduced with a dual Pre-post Filtering strategy.

The CARS model is evaluated in two case studies: an outdoor case from Paestum's archaeological site and another indoor area consisting of the Capodimonte National Museum.

The main components of the work are:

- Multimedia Data Management Engine (MDME): the component that manages numerous functions and can be considered the heart of architecture. Specifically, this component:
  - accesses the Indexing and Access Manager Module containing the associated content to items;
  - through the Feature Extraction Module captures multimedia data. These are used for indexing items and for obtaining a Structural Description data. Finally, the data is stored in the Multimedia Storage and Staging;
- Sensor Management Middleware: an element able to interact with sensors in the area under analysis to determine contextual information. These are stored in the Knowledge Base of the system;
- Knowledge Base: the component that manages location information geographical user and stores contextual information provided by the Sensor Middleware management. Also contains information about preferences user descriptions of points of interest;
- Multimedia Recommender Engine: the component that through various components constitutes the recommendation form and is composed of:
  - Candidate Set Building Module that selects suitable items to recommend to the user;
  - Object Ranking Module generating ratings for items selected from Candidate Set Building Module;
  - Visiting Paths Generation that dynamically selects a subset item based on contextual information and contingent actions of the user and possibly proposes a route to visit the site or the museum considered.

The Pre-filtering phase consists of the selection of candidates in the Candidate Set Building Module. These are chosen based on the user's location and his preferences or needs. The Post-filtering phase is carried out based on contextual information not considered in the pre-filtering phase and finalized the construction of the tour route.

The Recommender System consisting of the Object Ranking Module is a reworking of the recommendation mode associated with the Pagerank (Page et al. 1999; Albanese et al. 2011).

Turist@ (Batet et al. 2012) presents an architecture based on a Multiagent structure that can be classified as follows:

- **User-Agent:** this is the component that allows the user to interact with the application and its features. It also allows, through some preliminary questions, to build a preference vector for the specific user sent to the Recommender Agent. Finally, through the Agent in analysis, the user can filter the activities and request personalized suggestions;
- **Activity Agents:** they are Agents associated with single activities. Moreover, they present a specific database for the activity to which they are dedicated;
- **Broken Agent:** it aims to connect the User-Agent with the Activity Agents to end of making more performing communication and decrease the time to select the correct information;
- **Recommender System:** from the name, it is guessed that the component is finalized to the calculation of recommendations. The profile provided initially by the User-Agent will subsequently be dynamically modified based on the information acquired from the interaction between user and system. It also contains a database of the main features of some items to provide quick initial suggestions.

The most interesting features of the proposed application are the ability to dynamically update the user profile and the double choice to make the recommendation (Content-Based/Collaborative Filtering). The suggestions take into account the user's position, acquired through the GPS of the mobile device, and the activities to be considered whose location is stored in the dedicated database.

In del Carmen et al. (2017) is proposed a CARS where a Pre-Filtering strategy is exploited as part of a Collaborative Filtering recommendation technique. This approach is based on the connection between mobile devices via wireless communication so that no fixed infrastructure or central server is required for data storage. In fact, each mobile device will be exploited to acquire and store information related to rating, context, item, and user within a radius of 200–300 m. Moreover, an update of the stored data is foreseen due to the

variability of the data. According to the specific context in which they are released, ratings are stored in mobile devices, and only those that correspond to the identified context are exploited to provide suggestions. In particular, in the experimental phase, obtained through a simulation of some works of the Museum of Modern Art (MoMA) in New York, are considered the contexts: location, user's mood (happy, sad, neutral), Temperature of a room (warm, hot, cold), Number of people of a room (large, medium, small) and noise level in a room (high, medium, low). The operation of this approach can be summarized as follows:

- Data storing on the mobile device
- Computing users with similar preferences in the vicinity and, subsequently, rating predictions using the User-Based Collaborative Filtering technique (Ning et al. 2015) (which falls under Memory-Based techniques). If no similar user is found, the data will be updated;
- Adding to the list of items obtained in the previous step additional items in the vicinity with ratings above a tolerance set by the authors;
- The list of items to suggest is ordered based on the ratings associated with each item, and the user is provided with a suitable path to continue the visit.

Finally, Chat-Bot (Casillo et al. 2020) is an architecture that aims to interface with the user to guide him in an interactive experience immersed in the tourism and cultural heritage sector specifically. The application communicates with the user through text messages, which are simplified and analyzed with the Latent Dirichlet Allocation (LDA) (Blei et al. 2003). The LDA allows extracting the correlation between keywords of the text message and the topics. This step enables the system in order to create a user profile across his purposes and needs.

The application is based on the following modules:

- *Digital Storytelling Manager:* module aimed to guide the user across the experience without limiting the possibility of freely living the experience itself. The Digital Storytelling (Lambert 2013; Casillo et al. 2019) technique, which gives the module its name, is adequate for this purpose;
- *Context-Aware Manager:* the module aims to analyze all possible contexts. The Context Dimension Tree (CDT) is exploited for this purpose (Bolchini et al. 2009; Colace et al. 2020b);
- *Human/Computer Interaction Supervisor:* module to supervise the messages underlying the interaction between user and application. The primary purpose of the module is to identify unclear and excessively long

questions in order to provide unambiguous sensitive data to the Inference Engine;

- *Interaction Quality Tracker*: module designed to supervise the messages underlying the interaction between user and application. In this way, ambiguous and excessively long questions will be identified;
- *Knowledge Base*: the component that acquires the data made available by developers and the data about external services that can be recommended to users;
- *Inference Engine*: it is the main module of the architecture and is composed of two components:
  - Mixed Graph of Terms (mGT) building module: module that builds the mGT graph. The mGT allows linking the text message and associated domain. This module can also be exploited for the construction of the CDT;
  - Context Mining Module: module that can extract the specific context through the use of mGT;

The domain obtained by the interaction between a user, system, and context obtained by the Inference Engine is exploited to recommend.

This approach can be classified as Post-Filtering CARS because the forecast results from the filtering based on the topic obtained by text message. The recommendation without context values is already implemented in the Knowledge Base and is subsequently adapted through CDT and mGt.

*Evaluation Metrics exploited* In Table 6 are reported the evaluation metrics used to test the validity of the approaches just proposed in the field of Cultural Heritage. It can be seen that CES (Chianese and Piccialli 2016), and Bartolini et al. RS Bartolini et al. (2016) (concerning the experiment on the archaeological area of Paestum), Turist@ Batet et al. (2012), and Chat-bot Casillo et al. (2020) exploit user satisfaction. This method consists of observing when the Recommender System is able to provide appropriate suggestions to the user. This measure is an excellent method to evaluate multiple metrics simultaneously through the use of means such as questionnaires created ad hoc. In fact, both the accuracy of the recommendations provided and the

other metrics evaluated in Sect. 4.2, above all serendipity and diversity, contribute to user satisfaction. However, in the cases of SMARTMUSEUM (Ruotsalo et al. 2013), and Bartolini et al. RS Bartolini et al. (2016) (concerning the experiment on the Capodimonte Museum in Naples), accuracy measures are used.

## 5.1 Future research direction in cultural heritage domain

The study conducted in this paper has shown that there are many different approaches in the CARS sector. In dealing with context-aware recommender systems in the field of cultural heritage, it can be noted that many of the treated work present Content-Based approaches with the introduction of contextual information of Pre-Filtering, Post-Filtering, or both. The only Collaborative Filtering approaches are represented by del Carmen et al. (2017), and Turist@ Batet et al. (2012). In the first case, the ratings are provided by the mobile devices with which users are connected via wireless communication. In contrast, in the second case, the ratings of the Collaborative filtering approach, built in a Hybrid RS and supported by a Content-Based method, are not acquired from external sources. From the above consideration, one of the main limitations of CARS applied in Cultural Heritage is the lack of flexible APIs applicable in various fields. To overcome this lack, a possible solution could be to adopt a sharing standard. In this way, it would be possible to give an essential impact to the development of CARS in this paper's specific field of study. Moreover, since one of the limitations of Content-Based approaches is represented by the lack of diversity, it would be possible to recommend different items. In addition, there is also a lack of work integrating contextual information with Contextual Modeling approaches. This lack implies that an appropriate selection is made through the specific context, and the recommendation is developed through 2D techniques. Instead, outside the world of Cultural Heritage, Contextual Modeling approaches are widely used. It would be interesting to see if this approach would be beneficial in this particular field of study. Despite these considerations, the approaches proposed in Section 5 are efficient and give a heterogeneous overview of CARS applied to Cultural Heritage. The evolution of Tourism in Tourism 2.0 (Casillo et al. 2019) allows exploiting the available technologies to acquire contextual information (De Santo et al. 2020), which is common to the presented works. Moreover, it is interesting how the presented works are diverse despite having a common goal: to give visibility to the artistic and cultural heritage. This evaluation arises from the need for each system to acquire specific contextual information, as shown in Table 5 where the only contextual information in common is defined as location.

**Table 6** Evaluation metrics exploited by papers analyzed in Sect. 5

Approach	Evaluation metrics
CES	User satisfaction
SMARTMUSEUM Bartolini et al	Precision, Recall User satisfaction - MAE, RMSE
Turist@ del Carmen Rodríguez Hernández et al	User satisfaction MAE
ChatBot	User satisfaction



## 6 Conclusions

Context-aware recommender systems have been described and classified, and several approaches for introducing contextual information within Recommender Systems have been shown. The variety and heterogeneity of these approaches testify to the significant number of studies about this field and the wide diffusion that CARS have had in recent years. This research work also discussed the context classification used in the literature and the main datasets used in different application domains.

In particular, Recommender Systems that exploit contextual information represent crucial tools for improving users' experience in the Cultural Heritage field: through the CARS, visit archaeological sites and museums are made interactive and unique.

Future challenges concern new complex scenarios in which a change of context causes a transformation of the experience that is about to be lived.

**Funding** Open access funding provided by Università degli Studi di Salerno within the CRUI-CARE Agreement.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

## References

- Abowd GD, Dey AK, Brown PJ, Davies N, Smith M, Steggle P (1999) Towards a better understanding of context and context-awareness. In: International symposium on handheld and ubiquitous computing, Springer, New York, pp 304–307
- Adomavicius G, Tuzhilin A (2005) 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
- Adomavicius G, Tuzhilin A (2011) Context-aware recommender systems. In: Recommender systems handbook, Springer, New York, pp 217–253
- Adomavicius G, Tuzhilin A (2015) Context-aware recommender systems
- Adomavicius G, Sankaranarayanan R, Sen S, Tuzhilin A (2005) Incorporating contextual information in recommender systems using a multidimensional approach. *ACM Trans Inf Syst (TOIS)* 23(1):103–145
- Adomavicius G, Mobasher B, Ricci F, Tuzhilin A (2011) Context-aware recommender systems. *AI Magaz* 32(3):67–80. <https://doi.org/10.1609/aimag.v32i3.2364>
- Albanese M, d'Acierno A, Moscato V, Persia F, Picariello A (2011) A multimedia semantic recommender system for cultural heritage applications. In: 2011 IEEE fifth international conference on semantic computing, IEEE, pp 403–410
- Anastasiu DC, Christakopoulou E, Smith S, Sharma M, Karypis G (2016) Big data and recommender systems. *Novática: J Spanish Comput Sci Assoc* 240
- Baltrunas L, Ricci F (2009) Context-based splitting of item ratings in collaborative filtering. In: Proceedings of the third ACM conference on Recommender systems, pp 245–248
- Baltrunas L, Ricci F (2014) Experimental evaluation of context-dependent collaborative filtering using item splitting. *User Model User-Adapt Interact* 24(1–2):7–34
- Baltrunas L, Makcinskas T, Ricci F (2010) Group recommendations with rank aggregation and collaborative filtering. In: Proceedings of the fourth ACM conference on Recommender systems, pp 119–126
- Baltrunas L, Kaminskas M, Ludwig B, Moling O, Ricci F, Aydin A, Lüke KH, Schwaiger R (2011a) Incarmusic: context-aware music recommendations in a car. In: E-Commerce and web technologies, Springer, New York, pp 89–100
- Baltrunas L, Ludwig B, Ricci F (2011b) Matrix factorization techniques for context aware recommendation. In: Proceedings of the fifth ACM conference on Recommender systems, pp 301–304
- Baltrunas L, Church K, Karatzoglou A, Oliver N (2015) Frappe: Understanding the usage and perception of mobile app recommendations in-the-wild. [arXiv:150503014](https://arxiv.org/abs/150503014)
- Bartolini I, Moscato V, Pensa RG, Penta A, Picariello A, Sansone C, Sapino ML (2016) Recommending multimedia visiting paths in cultural heritage applications. *Multimed Tools Appl* 75(7):3813–3842
- Batet M, Moreno A, Sánchez D, Isern D, Valls A (2012) Turist@: agent-based personalised recommendation of tourist activities. *Exp Syst Appl* 39(8):7319–7329
- Bazire M, Brézillon P (2005) Understanding context before using it. In: International and interdisciplinary conference on modeling and using context, Springer, New York, pp 29–40
- Blei DM, Ng AY, Jordan MI (2003) Latent dirichlet allocation. *J Mach Learn Res* 3(Jan):993–1022
- Bokde D, Girase S, Mukhopadhyay D (2015) Matrix factorization model in collaborative filtering algorithms: a survey. *Proc Comput Sci* 49:136–146
- Bolchini C, Curino CA, Quintarelli E, Schreiber FA, Tanca L (2009) Context information for knowledge reshaping. *Int J Web Eng Technol* 5(1):88–103
- Braunhofer M, Elahi M, Ricci F (2014) Sts: A context-aware mobile recommender system for places of interest. *CEUR*
- Cao D, He X, Miao L, An Y, Yang C, Hong R (2018) Attentive group recommendation. In: The 41st International ACM SIGIR conference on research & development in information retrieval, pp 645–654
- CarlKadie JBD (1998) Empirical analysis of predictive algorithms for collaborative filtering. Microsoft Research Microsoft Corporation One Microsoft Way Redmond, WA, p 98052
- del Carmen Rodríguez-Hernández M, Ilarri S, Trillo R, Hermoso R (2017) Context-aware recommendations using mobile p2p. In: Proceedings of the 15th international conference on advances in mobile computing & multimedia, pp 82–91
- Casillo M, Clarizia F, Colace F, Lombardi M, Pascale F, Santaniello D (2019) An approach for recommending contextualized services in e-tourism. *Information* 10(5):180
- Casillo M, Colace F, Santo MD, Lemma S, Lombardi M (2019) Cat: a context aware teller for supporting tourist experiences. *Int J Comput Sci Eng* 20(1):69–87
- Casillo M, Clarizia F, D'Aniello G, De Santo M, Lombardi M, Santaniello D (2020) Chat-bot: a cultural heritage aware

- teller-bot for supporting touristic experiences. *Pattern Recognit Lett* 131:234–243
- Casillo M, Colace F, De Santo M, Lombardi M, Mosca R, Santaniello D (2021a) A recommender system for enhancing coastal tourism. In: *Research and innovation forum 2020: disruptive technologies in times of change*, Springer International Publishing, pp 113–122
- Casillo M, Conte D, Lombardi M, Santaniello D, Valentino C (2021b) Recommender system for digital storytelling: A novel approach to enhance cultural heritage. In: *Pattern Recognition. ICPR International Workshops and Challenges: Virtual Event, January 10–15, 2021, Proceedings, Part VII*, Springer International Publishing, pp 304–317
- Champiri ZD, Shahamiri SR, Salim SSB (2015) A systematic review of scholar context-aware recommender systems. *Exp Syst Appl* 42(3):1743–1758
- Chianese A, Piccialli F (2016) A smart system to manage the context evolution in the cultural heritage domain. *Comput Electr Eng* 55:27–38
- Colace F, De Santo M, Greco L, Moscato V, Picariello A (2015) A collaborative user-centered framework for recommending items in online social networks. *Comput Hum Behav* 51:694–704
- Colace F, De Santo M, Lombardi M, Mosca R, Santaniello D (2020) A multilayer approach for recommending contextual learning paths. *J Int Serv Inf Secur (JISIS)* 10(2):91–102
- Colace F, De Santo M, Lombardi M, Pascale F, Santaniello D, Tucker A (2020b) A multilevel graph approach for predicting bicycle usage in London area. In: *Fourth International congress on information and communication technology*, Springer, New York, pp 353–362
- Colombo-Mendoza LO, Valencia-García R, Rodríguez-González A, Alor-Hernández G, Samper-Zapater JJ (2015) Recommetz: a context-aware knowledge-based mobile recommender system for movie showtimes. *Exp Syst Appl* 42(3):1202–1222
- Comincioli V (2010) *Analisi numerica: metodi, modelli, applicazioni*. Apogeo Editore
- De Gemmis M, Lops P, Musto C, Narducci F, Semeraro G (2015) Semantics-aware content-based recommender systems. In: *Recommender systems handbook*, Springer, New York, pp 119–159
- De Lathauwer L, De Moor B, Vandewalle J (2000) A multilinear singular value decomposition. *SIAM J Matrix Anal Appl* 21(4):1253–1278
- De Santo M, Fabbri L, Mosca R, Lombardi M, Romano A, Santaniello D (2020) A multilevel approach to recommend working paths in industry 4.0. In: *2020 IEEE International conference on teaching, assessment, and learning for engineering (TALE)*, IEEE, pp 754–757
- Dey AK (2001) Understanding and using context. *Person Ubiquit Comput* 5(1):4–7
- Gašević D, Djuric D, Devedžić V (2006) *Model driven architecture and ontology development*. Springer, New York
- Golub GH, Van Loan CF (2013) *Matrix computations*, 4th. Johns Hopkins
- Gunawardana A, Shani G (2009) A survey of accuracy evaluation metrics of recommendation tasks. *J Mach Learn Res* 10(12)
- Hand DJ, Till RJ (2001) A simple generalisation of the area under the roc curve for multiple class classification problems. *Mach Learn* 45(2):171–186
- Haruna K, Akmar Ismail M, Suhendroyono S, Damiasih D, Pierewan AC, Chiroma H, Herawan T (2017) Context-aware recommender system: a review of recent developmental process and future research direction. *Appl Sci* 7(12):1211
- Herlocker JL, Konstan JA, Terveen LG, Riedl JT (2004) Evaluating collaborative filtering recommender systems. *ACM Trans Inf Syst (TOIS)* 22(1):5–53
- Hong J, Suh EH, Kim J, Kim S (2009) Context-aware system for proactive personalized service based on context history. *Exp Syst Appl* 36(4):7448–7457
- Huang Z, Xu X, Zhu H, Zhou M (2020) An efficient group recommendation model with multiattention-based neural networks. *IEEE Trans Neural Netw Learn Syst* 31(11):4461–4474
- Ilarri S, Trillo-Lado R, Hermoso R (2018) Datasets for context-aware recommender systems: Current context and possible directions. In: *2018 IEEE 34th international conference on data engineering workshops (ICDEW)*, IEEE, pp 25–28
- Kaminskas M, Bridge D (2016) Diversity, serendipity, novelty, and coverage: a survey and empirical analysis of beyond-accuracy objectives in recommender systems. *ACM Trans Interact Intell Syst (TiIS)* 7(1):1–42
- Kane F (2018) *Building Recommender Systems with Machine Learning and AI: Help people discover new products and content with deep learning, neural networks, and machine learning recommendations*. Independently published
- Karatzoglou A, Amatriain X, Baltrunas L, Oliver N (2010) Multiverse recommendation: n-dimensional tensor factorization for context-aware collaborative filtering. In: *Proceedings of the fourth ACM conference on Recommender systems*, pp 79–86
- Kitchenham B, Charters S (2007) *Guidelines for performing systematic literature reviews in software engineering*
- Kohavi R, Henne RM, Sommerfield D (2007) Practical guide to controlled experiments on the web: listen to your customers not to the hippo. In: *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp 959–967
- Koren Y (2009) Collaborative filtering with temporal dynamics. In: *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp 447–456
- Koren Y, Bell R, Volinsky C (2009) Matrix factorization techniques for recommender systems. *Computer* 42(8):30–37
- Košir A, Odić A, Kunaver M, Tkaličič M, Tasič J (2013) Ldos-comoda dataset
- Lambert J (2013) *Digital storytelling: capturing lives. Creating community*
- Liu J, Wu C, Liu W (2013) Bayesian probabilistic matrix factorization with social relations and item contents for recommendation. *Decis Support Syst* 55(3):838–850
- Missaoui S, Kassem F, Viviani M, Agostini A, Faiz R, Pasi G (2019) Looker: a mobile, personalized recommender system in the tourism domain based on social media user-generated content. *Person Ubiquit Comput* 23(2):181–197
- Nguyen TN, Ricci F (2017) A chat-based group recommender system for tourism. In: *Information and communication technologies in tourism 2017*, Springer, New York, pp 17–30
- Ning X, Desrosiers C, Karypis G (2015) A comprehensive survey of neighborhood-based recommendation methods. *Recommender systems handbook*, pp 37–76
- Oku K, Nakajima S, Miyazaki J, Uemura S (2006) Context-aware svm for context-dependent information recommendation. In: *7th International conference on mobile data management (MDM'06)*, IEEE, pp 109–109
- Page L, Brin S, Motwani R, Winograd T (1999) The pagerank citation ranking: bringing order to the web. Tech. rep, Stanford InfoLab
- Pavlidis G (2019) *Recommender systems, cultural heritage applications, and the way forward*. *J Cult Herit* 35:183–196
- Ponte JM, Croft W (1998) Language modeling approach to information retrieval. In: *SIGIR Forum (ACM Special Interest Group on Information Retrieval)*, pp 275–281
- Quarteroni A, Sacco R, Saleri F (2010) *Numerical mathematics*, vol 37. Springer, New York
- Ramirez-García X, García-Valdez M (2014) Post-filtering for a restaurant context-aware recommender system. In: *Recent advances*

- on hybrid approaches for designing intelligent systems, Springer, New York, pp 695–707
- Raza S, Ding C (2019) Progress in context-aware recommender systems—an overview. *Comput Sci Rev* 31:84–97
- Ricci F, Rokach L, Shapira B (2015) Recommender systems: introduction and challenges. In: *Recommender systems handbook*, Springer, New York, pp 1–34
- Rodriguez JD, Perez A, Lozano JA (2009) Sensitivity analysis of k-fold cross validation in prediction error estimation. *IEEE Trans Pattern Anal Mach Intell* 32(3):569–575
- Ruotsalo T, Haav K, Stoyanov A, Roche S, Fani E, Deliai R, Mäkelä E, Kauppinen T, Hyvönen E (2013) Smartmuseum: a mobile recommender system for the web of data. *J Web Semant* 20:50–67
- Said A, De Luca EW, Albayrak S (2011) Inferring contextual user profiles—improving recommender performance. In: *Proceedings of the 3rd RecSys workshop on context-aware recommender systems*
- Salakhutdinov R, Mnih A (2008) Probabilistic matrix factorization. *Adv Neural Inf Process Syst* 20
- Schaffers H, Komninos N, Pallot M, Trousse B, Nilsson M, Oliveira A (2011) Smart cities and the future internet: Towards cooperation frameworks for open innovation. *The future internet assembly*. Springer, Berlin, pp 431–446
- Shani G, Gunawardana A (2011) Evaluating recommendation systems. In: *Recommender systems handbook*, Springer, New York, pp 257–297
- Shin D, Lee Jw, Yeon J, Lee Sg (2009) Context-aware recommendation by aggregating user context. In: *2009 IEEE conference on commerce and enterprise computing*, IEEE, pp 423–430
- Symeonidis P, Zioupos A (2016) *Matrix and tensor factorization techniques for recommender systems*, vol 1. Springer, New York
- Thorat PB, Goudar R, Barve S (2015) Survey on collaborative filtering, content-based filtering and hybrid recommendation system. *Int J Comput Appl* 110(4):31–36
- Van Rijsbergen C (1979) *Information retrieval*. Butterworth-heinemann
- Villegas NM, Sánchez C, Díaz-Cely J, Tamura G (2018) Characterizing context-aware recommender systems: a systematic literature review. *Knowl Based Syst* 140:173–200
- Wang CD, Deng ZH, Lai JH, Philip SY (2018) Serendipitous recommendation in e-commerce using innovator-based collaborative filtering. *IEEE Trans Cybern* 49(7):2678–2692
- Wang R, Cheng HK, Jiang Y, Lou J (2019) A novel matrix factorization model for recommendation with lod-based semantic similarity measure. *Exp Syst Appl* 123:70–81
- Xu Z, Chen L, Chen G (2015) Topic based context-aware travel recommendation method exploiting geotagged photos. *Neurocomputing* 155:99–107
- Yao Y (1995) Measuring retrieval effectiveness based on user preference of documents. *J Am Soc Inf Sci* 46(2):133–145
- Zhai C, Lafferty J (2004) A study of smoothing methods for language models applied to information retrieval. *ACM Trans Inf Syst (TOIS)* 22(2):179–214
- 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. *Knowl Based Syst* 166:132–139
- Zheng Y (2015) A user’s guide to carskit. [arXiv:151103780](https://arxiv.org/abs/151103780)
- Zheng Y, Burke R, Mobasher B (2012) Differential context relaxation for context-aware travel recommendation. In: *13th International conference on electronic commerce and web technologies (EC-WEB 2012)*, pp 88–99
- Zheng Y, Mobasher B, Burke RD (2013) The role of emotions in context-aware recommendation. *Decisions@ RecSys 2013*:21–28
- Zheng Y, Mobasher B, Burke R (2014) Contexts recommendation using multi-label classification. In: *Proceedings of the 13th IEEE/WIC/ACM international conference on web intelligence (WI 2014)*, IEEE/WIC/ACM
- Zheng Y, Mobasher B, Burke R (2015) Carskit: A java-based context-aware recommendation engine. In: *2015 IEEE international conference on data mining workshop (ICDMW)*, IEEE, pp 1668–1671

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.