Computer Science Review 37 (2020) 100255

Contents lists available at ScienceDirect

Computer Science Review

journal homepage: www.elsevier.com/locate/cosrev

Review article

Context Aware Recommendation Systems: A review of the state of the art techniques☆

Saurabh Kulkarni [a,*], Sunil F. Rodd [b]

[a] Fr. Conceicao Rodrigues College of Engineering, Bandra, Mumbai, India
[b] Gogte Institute of Technology, Belagavi, India

ARTICLE INFO

Article history:
Received 17 May 2019
Received in revised form 27 April 2020
Accepted 6 May 2020
Available online 20 May 2020

Keywords:
Recommendation systems
Context awareness
Bio-inspired computing techniques
Statistical computing techniques

ABSTRACT

Recommendation systems are gaining increasing popularity in many application areas like e-commerce, movie and music recommendations, tourism, news, advertisement, stock markets, social networks etc. Conventional recommendation systems either use content based or collaborative filtering based approaches to model user preferences and give recommendations. These systems usually fail to consider evolving user preferences in different contextual situations. Context Aware Recommendation Systems take different contextual attributes into consideration and try to capture user preferences correctly. This survey focuses on the state-of-the art computational intelligence techniques trying to improve conventional design using contextual information. Further, these techniques are grouped into bio-inspired computing techniques and statistical computing techniques. The literature related to these techniques mentioning their ability to handle challenges faced by Context Aware Recommendation System are presented in this survey. The survey also talks about context inclusion strategies, classification of the contexts used in the literature reviewed, their impact on the problems faced by the recommendation systems, effective usage of these contexts, datasets used in the domain, future research scope in all the reviewed techniques and overall future research directions and challenges.

Contents

☆ No author associated with this paper has disclosed any potential or pertinent conflicts which may be perceived to have impending conflict with this work.

* Corresponding author.
E-mail address: saurabh.kulkarni@fragnel.edu.in (S. Kulkarni).

https://doi.org/10.1016/j.cosrev.2020.100255

1. Introduction

Recommendation is an important task in the decision making process about selection of items especially when an item space is large, diverse and constantly updating. In such an item space, there is a high probability that certain items and their characteristics are unknown to the users and therefore it is likely that the users may miss out items having characteristics aligned with their preferences. Before the advent of web based systems, majority of the times, the recommendation task was carried out using the *word of mouth* paradigm [1]. With the revolution in web based systems, many items like products, businesses, services etc. are now becoming available online and as a consequence, a recommendation task is also becoming online. So, the recommendation systems have evolved to cater to the needs of online recommendation process. Recommendation system is an online information retrieval system which helps in finding and suggesting appropriate items by aligning the search with user preferences [2,3]. It suggests relevant items to the users based on their preferences and try reducing their efforts in searching usually large and unknown item space. In certain business centric applications, it helps to improve visibility of items and may lead to branding and promotion of items thus creating the *word of mouth* effect. In the process of suggesting relevant items to the users, recommendation systems have to learn user preferences. Many a times user preferences on many items are not available and therefore recommendation systems have to predict user preferences while suggesting items. Current recommendation systems use certain techniques and models to learn user preferences in order to perform recommendation task. They are conventionally classified into following three categories [4,5]:

- *Content based recommendation systems* — Recommendations are based on content of the items towards which the user has shown inclination in the past. e.g. Genre of the movie.
- *Collaborative Filtering (CF) based recommendation systems* — Recommendations are based on items preferred by users with similar preferences in the past.
- *Hybrid recommendation systems* — Recommendations are based on both content based as well as CF based techniques.

In content based recommendation systems, similarity between items is computed based on the features of the items. Items possessing similar features are grouped together based on similarity value between them [5]. So it includes building an item profile using usually the text associated with an item which describes its features e.g. Genre of movie, cast of the movie. Measures like term-frequency inverse document frequency(tf–idf) scores are used to find similarity in keywords in the documents related to the items [5]. Apart from this, even techniques like Bayesian methods, Artificial Neural Network etc. are also used in content based recommendation [4]. As the users have to enter relevant keywords while searching for interested items, search results depend on usage of keywords. Therefore, few techniques make use of Rocchio's algorithm which does modifications in the initial search query of the users incorporating feedback [6].

In real life, apart from content of the item, suggestions given by peers with similar interests also impact the decision making process of the user. The model of this paradigm in online recommendation systems is called CF [5]. Such systems try to find users having similar interests with the target user using similarity measures like cosine similarity or Jaccard similarity [5]. It then forms cluster of similar users and try to find normalized rating for an item based on the ratings given by similar users. The items whose normalized ratings are above predetermined threshold are then suggested to the user [5]. So CF involves prediction of rating for an unrated item by a target user based on the ratings given by similar users on that item. CF techniques are conventionally grouped into two categories [7,8]:

- *Memory based algorithms* — These are heuristic based algorithms that try to predict target user rating for an item based on partial information available about the target user and normalized weights obtained from the dataset [7,8]. Commonly used techniques in memory based algorithms are Pearson correlation coefficient and vector similarity techniques [7,8]. Some advanced techniques in memory based algorithms include default voting, inverse user frequency, case amplification and imputation-boosted CF algorithms [8].
- *Model based algorithms* — These are machine learning models trying to recognize patterns in datasets available for CF [7,8]. Commonly used methods in this category include Bayesian networks, clustering models, regression models, latent semantic models etc.[8].

Hybrid recommendation systems try to overcome the limitations posed by content based and CF techniques. Hybrid recommendation approaches are broadly classified into following categories [4]:

- Combination of individual predictions made by content based and CF techniques
- Inclusion of features of content based systems in CF
- Inclusion of features of CF based systems in content based systems
- Development of a model which includes features of content-based and CF systems together

All the conventional approaches in the recommendation systems discussed so far, work on preferences given by users usually in the form of ratings in the user–item space. But these systems usually fail to consider the fact that user preferences change over a period of time. There are certain contextual factors which may cause this change in preferences of the users [9]. Therefore, a new type of recommendation systems called Context Aware Recommendation Systems are developed to incorporate contextual influence on user preferences [10]. Since context can be multi-dimensional in nature, the user–item space along with the contextual attributes may become multi-dimensional. Therefore, the recommendation systems have to learn user preferences in different contextual situations. In order to deal with this challenge, recommendation systems have to either go for developing new approaches or improvise the existing ones.

A survey of classification of Context Aware Recommendation Systems is presented in Hong et al. [11]. This survey classifies Context Aware Recommendation Systems in user interface layer, application layer, middleware, network infrastructure and concept and research. This survey gives suggestions on capturing context, finding their relevance and saving of the user information. But the recommendation systems developed were between the years 2000–2007. So, they reflect the few extensions of the conventional content based and CF techniques.

Haruna et al. [12] presents different applications of Context Aware Recommendation Systems, context modeling techniques, context inclusion approaches and evaluation metric. But there is no detailed explanation of the non-conventional and adapted conventional recommendation techniques in Context Aware Recommendation Systems.

A survey of Context Aware Recommendation Systems including context inclusion methods and recommendation techniques is presented by Villegas et al. in [13]. The recommendation techniques used in the survey are content based recommendations, CF based recommendations and hybrid recommendations.

Seyednezhad et al. [14] also includes review of literature related to context inclusion techniques but does not focus on improvisations done in the recommendation techniques to improve accuracy of recommendation systems in the contextual setting.

The surveys in [12–14], focus on the development of the conventional recommendation approaches in the Context Aware Recommendation Systems setting. This survey extensively covers the computational intelligence techniques applied in the area of Context Aware Recommendation Systems. Their usage in the domain of Context Aware Recommendation Systems started approximately in the year 2006 with lesser contextual information. Since then, there has been a development and evolution in the application of these techniques in recommendation systems. More research was carried out on their application in this domain apart from conventional recommendation approaches. Therefore, these are the state of the art techniques in this domain. These are the techniques which are used either along with conventional recommendation approaches to improve their accuracy or can be used in a standalone manner. This survey concentrates on the application of these techniques in Context Aware Recommendation Systems from the year 2006–2019. Therefore, this survey covers techniques not discussed in [12–14].

Apart from this, [15] mentions different types of contexts which are discussed in Section 2. This survey categorizes contexts by the taxonomy specified in [15] and tries to discuss the development of different techniques used and issues faced by Context Aware Recommendation Systems in the light of these context types. It tries to explore the impact of these contexts on the development of Context Aware Recommendation Systems which was not done in any of the surveys in [12–14].

This survey describes context aware datasets and their usage in Context Aware Recommendation Systems. It also explains different performance evaluation measures and their usage in the area of Context Aware Recommendation Systems. For all the discussed techniques, a possible future scope for experimentation is also given in the survey. Finally, a future research direction in Context Aware Recommendation Systems is also discussed.

The computational intelligence techniques evolved in the domain of Context Aware Recommendation Systems based on the available literature can be broadly classified into bio-inspired computing techniques [16] and statistical computing techniques [17,18]. Such classification can help in the understanding of the inherent features of the techniques which can be used to tackle different issue faced by the system e.g. some of the bio-inspired techniques like ACO can help in optimization of the recommendation list whereas some statistical learning techniques like LDA can be used to get latent context knowledge. So the classification may also help in future development of hybrid model considering pros and cons of the techniques. So the literature is selected from the reputed international journals and conference proceedings published in Elsevier, Springer, IEEE, ACM etc. which discusses the usage and development of the computational intelligence techniques from the above mentioned classification. The usage of these techniques in addressing the different issues faced by the recommendation system, the way of incorporating context information is also considered during selection of the technique. Fig. 1 gives the listing of the techniques used in the survey. They are broadly classified into bio-inspired computing techniques and statistical computing techniques. The remainder of the paper is organized as follows. Section 2 discusses Context Aware Recommendation Systems including meaning of context, classification of context, processing of contexts in recommendation systems etc. A survey of state of the art bio-inspired algorithms used in the domain of Context Aware Recommendation Systems including their key aspects and possible improvements are presented in Section 3. Section 4 gives advances in statistical computing techniques with their major contribution in the field of Context Aware Recommendation Systems along with probable improvisations in the proposed approaches. The discussion on the techniques and future research directions is given in Section 5 and Section 6 is the conclusion of the survey.

2. Context Aware Recommendation Systems

The task of a recommendation systems is to correctly obtain user preferences and give suggestions based on them. Recommendation systems work with available set of ratings given by the users on set of items and try to predict the remaining ratings. As per the discussion in [10], the conventional recommendation systems work on predicting in a rating function R which can be represented as

$$R : User \times Item \rightarrow Rating$$

These systems work in user–item space having two dimensions. As per the discussion done earlier, the context helps in shaping of the preferences of the user which is not dealt with in conventional recommendation systems. Context Aware Recommendation Systems try to incorporate context into conventional user–item space. Therefore, for such systems, prediction of a rating function R as per [10] can be represented as

$$R : User \times Item \times Context \rightarrow Rating$$

As seen from the above representation, the search space is multi-dimensional and hence becomes computationally expensive. The challenge in front of such systems is to learn user preferences in different contextual situations. Capturing of the contextual attributes suitable for the domain under consideration and incorporating it in the recommendation process is a key to develop such systems.

Before incorporating context in recommendation systems, understanding the meaning of the term context is very important. As per the definition given in [19], "the context is any information that characterizes a situation related to the interaction between humans, applications and the surrounding environment". This implies that the context has an impact on the interaction between users and the system having set of items under certain environmental conditions. All these together form a situation which may help in constructing user preferences.

Contexts may also change over a period of time. Based on this concept, [15] categorizes context into:

- *Static* — Contextual attributes and their structure remains same over a period of time

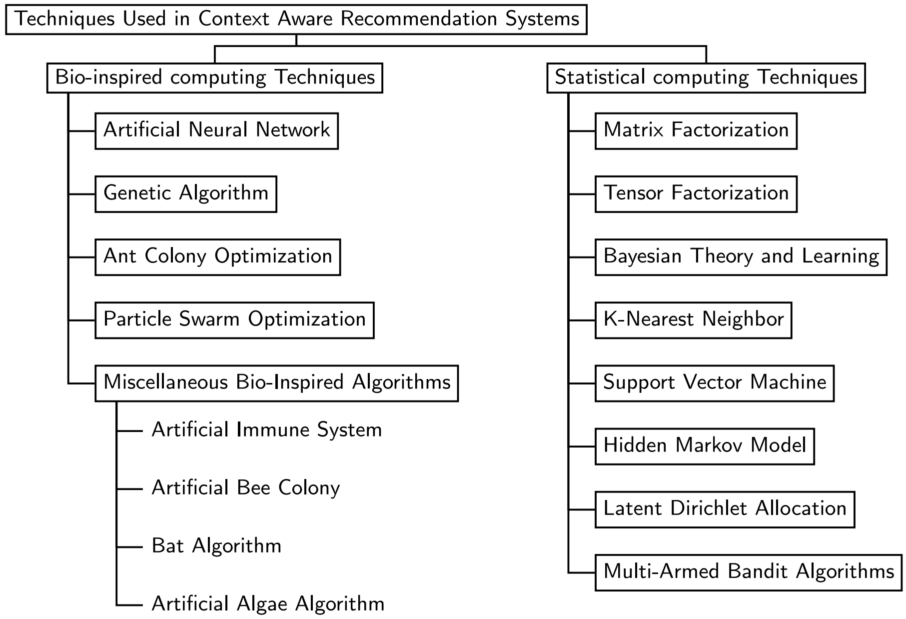

Fig. 1. Layout of the techniques used in the survey.

- *Dynamic* — Contextual attributes and/or their structure changes over a period of time. Some contextual attributes may become obsolete and hence can be removed from the system. Sometimes, few new attributes that consider new relevant situations can be added to the system.

Based on the context information available with the recommendation systems, [15] classify context into:

- *Fully observable* — Recommendation system knows everything about the contextual attributes including their structure
- *Partially observable* — Recommendation system only knows certain aspects of contextual attributes
- *Unobservable* — Recommendation system does not explicitly know anything about contextual attributes

Combining these two context taxonomies, there are six possible context types that may be helpful in designing Context Aware Recommendation Systems namely static fully observable, static partially observable, static unobservable, dynamic fully observable, dynamic partially observable and dynamic unobservable. Table 1 describes these context types with the examples.

Another challenge in Context Aware Recommendation System development is the way to include context in the recommendation methods. Adomavicius et al. [10] note that there are two general methods exist to perform context inclusion task. The first one is context driven search and querying approach which uses context information obtained from the user or by using sensors sensing environment that collect such information. After such information is collected a relevant database is queried to get aligned matches with the query. E.g. COMPASS is a system proposed in [20] is a tourism recommendation system which takes users contextual attributes like location into consideration while giving suggestions not only related to services like restaurants but also tourist places like museums, parks etc. This will be helpful for tourists where they do not know about the services and tourist places available in the vicinity as the place may be unknown to them. A tour recommendation system based on geotagged social media photos using context driven search and querying is proposed in [21]. Using the travel history gathered from geotagged photos and information on social media, new locations are recommended. The proposed method handles dynamic queries including contexts like time and weather to give personalized recommendations of locations to the travelers. The user query is improvised with the context information of time and weather. Then, the user preferences learned from geotagged social media photos are used to find user preferences. Based on the preferences of similar users, location database is scanned to recommend similar traveling options. Christos et al. also describe some of the applications of context driven search and querying in [22]. This approach can be used in applications like tourism, police patrolling, military etc. [22]. It also highlights that the current research in this approach is concentrated on sensing the contexts and identifying the contexts. Another travel recommendation system using context driven search and querying is suggested in [23]. In this approach, preference feedback given by the users is used to build their profiles which are used for context driven querying and search. As per [22], context driven querying and search is comparatively static approach and does not make use of sophisticated recommendation algorithms. Therefore, this survey has a less focus on this approach.

Another method to include context is contextual preference elicitation and estimation. In this method, recommendation systems attempt to learn user preferences in different contextual situations. Learning depends on implicit or explicit feedback obtained from the users. For learning contextual preferences, the systems either use existing recommendation techniques or use machine learning techniques. There are three basic approaches namely contextual pre-filtering, contextual post-filtering and contextual modeling which perform contextual preference elicitation and estimation [10].

In contextual pre-filtering, values of contextual attributes are used as constraints for selection of ratings which are then used in conventional user–item space. Therefore, irrelevant ratings are filtered out before they are used by recommendation systems [10]. There are certain reduction based techniques used to reduce the dimensionality of contextual dimensions but they often lead to local prediction models [10]. Reduction technique involves converting a multidimensional contextual recommendation space to appropriate two dimensional space [10]. Because the problem space is reduced to two-dimensional space, the conventional two dimensional recommendation algorithms can

Table 1
Context Taxonomies.

	Fully observable	Partially observable	Unobservable
Static	• Recommendation system knows everything about the relevant contextual attributes including their structure. These attributes and their structure remains same over a period of time. • Example: while predicting a rating for a movie, the system is aware of all the relevant contexts and the values those contexts can assume like time when the movie is watched(weekday or weekend), companion with whom the movie is watched(friends, family, alone)	• Recommendation system has a partial knowledge of the relevant contextual attributes. It maybe aware of few contexts out of all the relevant contexts. But the structure and values of all the contexts remain more or less same over a period of time. • Example: a movie recommendation system maybe aware of time when the movie is watched and the companion with whom the movie is watched but it may not be aware of the device on which the movie is watched(Multiplex screen, Television set, Computer, Tablet, smart phone)	• Recommendation system has no explicit knowledge of the contextual attributes but it makes use of latent knowledge to know about them. In this case, the structure of latent attributes will not change over a period of time. • Example: in case of a system recommending music to a driver in a car, it may not be aware of the surrounding traffic conditions or the type of road but it may learn about them by latent knowledge like location of the car, news feeds, social media data etc.
Dynamic	• Recommendation systems has all the prior knowledge about the contexts. The relevance of the context changes over a period of time. • Example: a movie recommendation system knows about the contexts like time,companion and may learn over a period of time that the companion with whom the movie is watched is not having any impact on the recommendations and therefore, it can drop this context.	• Recommendation system has a partial knowledge of the contexts and the relevance of the context changes over a period of time. • Example: a movie recommendation system may learn over a period of time that the companion with whom the movie is watched is not having any impact on the recommendations and therefore, can drop this context. It may also learn that a device on which the movie is watched is having an impact on the user preferences but it may not be aware of all the available device types.	• Recommendation system has no prior knowledge of the contexts and the relevance of the contexts keep on changing. • Example: a music recommendation system does not know about the emotions of the user. It may eventually learn about the emotions by the implicit feedback (e.g. changing of the music track or type of music) or the explicit feedback (e.g. a review or a social media comment) given by the user i.e. a system may learn from the feedback that a user is in a romantic mood and try to recommend romantic music tracks. Even emotions keep on changing over a period of time so is their relevance on the kind of music recommended. So these kind of recommendation systems adapt themselves based on the interaction with the user.

be applied to the data [10]. Using exact context in reduction technique may lead to data sparsity problem as it can pre-filter so much data that there is no sufficient data left to be used for preference prediction [10,15]. To tackle this problem, context generalization approach is used [10,15,24]. In context generalization, instead of using the exact context value, a generalized value for the context is used e.g. if it is found that a sell of a particular product is almost same over Saturday and Sunday, then these values can be replaced by weekend as a generic value. A reduction technique based on similarity computation among contextual conditions is proposed in [25]. This approach uses the notion that preferences of users in similar contextual conditions are similar [10,25]. Item-splitting is also one of the techniques used in contextual pre-filtering where if a notable difference is found in the ratings in different contextual values, ratings are split and new artificial items are created [26,27]. It uses the ratings in non-target contextual conditions also to predict ratings in target contextual conditions, e.g. a recommendation system considering weather as a context and winter as a target contextual condition also uses ratings from summer and rainy season considering they may have an impact on conditions in the winter season [27]. The difference ratings in different contextual values can be computed using the information gain obtained by the split on the knowledge of rating on a particular item [27]. Empirical evaluation shows that reduction and item-splitting leads to less values of Mean Absolute Error(MAE) compared to non-contextual strategies [26,27]. Singh et al. [28] propose an additional user-splitting along with item-splitting and found lesser value of MAE than User-based CF. Area of interest based techniques are used for personalized recommendations where users specify certain preferences. Based on those preferences data is filtered out before going to recommendation system [29]. Area of interest based contextual pre-filtering strategy for location based recommendation is proposed in [29] where the users decide the distance. All the items outside the circumference of the circle formed by the specified distance and user's current location acting as a center are filtered from the recommendation process. A tourist guide heritage system is developed using contextual attributes like time, companion and weather which uses area of interest of the users along with generalization of contexts as a pre-filtering technique [24]. In semantic similarity pre-filtering, similarity between the contextual conditions is estimated using meaning of the condition [30,31]. For instance, in a tourist recommendation system, adverse weather conditions like cold or rain will have positive effect on the indoor attractions like museum and a negative effect on outdoor places like forts [30,31]. A semantic similarity based pre-filtering technique using vector space model for contextual attributes is proposed in [30]. The experimental evaluation of this technique is carried out in [31]. Cordina et al. have proposed Distributional Semantics Pre-Filtering(DSPF) in [32]. This approach uses similarity between two contextual conditions decided by the similarity threshold.

Contextual post-filtering involves generation of predicted ratings first and then using contextual information, those ratings are adjusted for every user. Heuristic based approaches and model based approaches are the two types of post-filtering approaches available [10]. A music recommendation system using sequential pattern mining and latent topic modeling is proposed in [33] which takes contextual post-filtering approach to arrange the rankings of songs in the playlists. A CF based contextual post-filtering approach for restaurant recommendation is proposed in [34].

In contextual modeling approach, contextual attributes are used in the process of prediction of the ratings by recommendation systems. It can also be categorized into heuristic based and model based approaches. Heuristic based approaches use extension of nearest neighbor techniques in multi-dimensional space. Model based approaches use models like hierarchical regression based Bayesian models, Support Vector Machine(SVM) based models etc. [10]. Fig. 2 gives the categorization of the techniques used in context inclusion in recommendation systems. The detailed evaluation and comparison of the context inclusion techniques can be found in [35,36].

The recommendation systems face three major issues as described below.

- *Cold start* — Whenever a new user or new item is introduced in the recommendation system, there is insufficient information available about such user or item. This problem is called a cold start problem. In case of new user, it becomes difficult to learn about the preferences due to lack of data available. In case of new item, as there are less number of users who have used such items, incorporation of such items in recommendation process takes time [37,38].
- *Data Sparsity* — If feedback given by customers is insufficient for the recommendation system models to work on, the situation is termed as data sparsity [39]. It can happen due to cold start or it can be due to tendency of the users of not giving enough feedback. This can impact the range and quality of recommendation [40].
- *Scalability* — As number of users and items increase, the computation cost also increases. This may affect the performance and response time of the recommendation system [41].

To measure the performance of a Context Aware Recommendation System, evaluation criteria have been specified in the literature. The evaluation criteria are broadly classified into offline and online evaluation [12,42,43].

- Offline evaluation — This type of evaluation is performed when the dataset is collected prior to design of the system [12,42,43]. The system then operates on this data and predicts the preference of the users on the items. This performance measure essentially evaluates the accuracy of the system in terms of its capacity to predict preferences [12,42]. The advantage of using offline evaluation is it can be done faster compared to online evaluation. But the disadvantage is that the system is unable to track the changes in user preferences in real time as there is no real time feedback from the users [12,42]. There are different types of offline performance evaluation metric. Some of the commonly used offline evaluation metric are described below.

  1. *Root Mean Squared Error (RMSE)* — It is the square root of sum of the squares of the differences between predicted values and corresponding actual value specified number of observations [42]. It gives the standard deviation of the prediction errors [12,42].
  2. *Mean Absolute Error (MAE)* — It is the arithmetic average of the absolute difference between predicted and actual values [42].
  3. *Precision* — It is the number of true positives over the sum of true positives and true negatives [42]. In other words, it tells the proportion of data that the model predicts relevant is actually relevant [12].
  4. *Recall* — It is the number of true positives over the sum of true positives and false negatives [42]. It is the ability of the model to find all the relevant data from the dataset [12].
  5. *F-measure* — It is the harmonic mean of precision and recall [42].
  6. *Area Under the Curve (AUC)* — It is the capability of the model to differentiate different classes in the dataset [42].
  7. *Normalized Discounted Cumulative Gain (NDCG)* — It is a measure related to the ranking of the items [12,42] A recommendation system returns a list of recommended items. Every item in the list has a relevance score associated with it called gain. The summation of all such gains is the cumulative gain [42]. Before computing cumulative gain, each of the gain value is divided by the logarithm of the position of the item. This process is called discounting and by using it, DCG is calculated. The normalized DCG is called NDCG [42].
  8. *Mean Average Precision (MAP)* — It is the mean of the average precision values over the ranks in the relevant recommended items [12].
  9. *Hit rate* — Number of items in the test set that were also present in the recommended items given by the system for each user is called number of hits. Number of hits over total number of users is called hit rate [44].
  10. *Perplexity* — It is a metric to evaluate topical models. It measures the quality of topics extracted by the topical model using training documents which allow to predict the occurrence of the words in testing documents [45].

- Online evaluation — This evaluation is used when the experiment is conducted in real time [12,42]. It evaluates the real time feedback of the users. It takes comparatively more time to evaluate online metric as it has to be monitored over a longer period of time [12,42,43]. This type of evaluation helps to understand user interaction behavior with the system which also is an important consideration while assessing the quality of the recommendations [12,42,46]. Some of the common online evaluation metric used in Context Aware Recommendation Systems are given below.

  1. *Click Through Rate (CTR)* — It is the count of recommendations that are clicked by the user. It measures the real time feedback of the users about their preferences [12,43].
  2. *Bounce Rate* — It is the percentage of users who have seen the list of recommendations given by the system but instead of exploring those recommendations further, they chose to exit the recommendation system [46–48].

There are few other evaluation criteria suggested in [42]. Some of them are discussed below.

- Coverage — It is either the percentage of total available items a system can recommend(called as item space coverage) or the percentage of total available users for which a system can recommend(called as user space coverage) [42].
- Confidence — It is the system's trust in prediction of the recommended items. It is usually measured in terms of the probability of correctness of the predicted value [42].
- Trust — It is the trust users have on the recommendations given by the system [42].
- Novelty — It is the percentage of items recommended out of total recommended items that are unknown to the user [42].

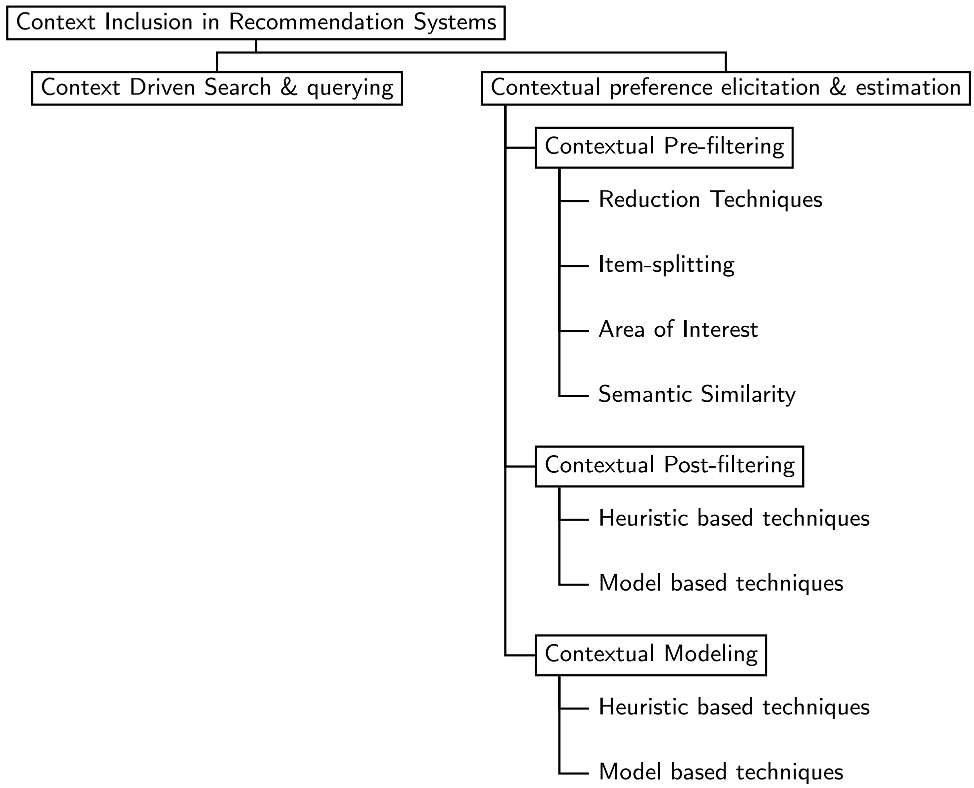

Fig. 2. Representational view of context inclusion techniques discussed in the survey.

- Diversity — It measures how many recommended items belong to diverse group of items. It helps to understand interest of users in different types of items [42].
- Adaptivity — It measures the system's capacity to adapt to the changes in the item space or changing trends of the user preferences [42].
- User satisfaction — It measures the satisfaction level of the user after going through the recommendation list. It can be measured using implicit or explicit feedback from the users [42].

The detailed discussion of the evaluation measures and their usage can be found in [36,42,43].

Another important aspect in Context Aware Recommendation Systems is the availability of the datasets having context information. This factor plays a key role especially in offline experimentation [12,42]. The discussion on the datasets used in Context Aware Recommendation System literature is given in Table 18 of Section 5.

Next two sections describe different techniques used in Context Aware Recommendation System research along with their contribution in dealing with cold start, data sparsity and scalability problems and the type of contextual attributes used in them.

3. Bio-inspired computing techniques in Context Aware Recommendation Systems

Bio-inspired algorithms are the computational intelligence techniques inspired by the biological phenomena occurring in nature. They have applications in many domains including information processing, optimization, decision making, information security, computer networks etc. They are generally used to solve complex real world problems [49]. Usually such complex problems deal with multi-dimensional space wherein, the problem and constraints specified are dynamic in nature. Bio-inspired algorithms are suitable for handling such problems [16]. Due to addition of contexts, the problem space for recommendation systems become multidimensional with dynamic nature of user preferences. Hence, these techniques are suitable for handling this problem. This section gives overview of such techniques used in Context Aware Recommendation Systems.

3.1. Artificial Neural Network (ANN)

ANN is a computational model depicting the way network of neurons work in the human brain. ANN consists of nodes connected by links. Links are used to propagate signal generated through an activation function. Every link has certain weight associated with it. Every node computes weighted sum of the inputs and then applies activation function to the sum to compute output [50].

Liu et al. [51] have proposed music playlist generator using time parameter in ANN. Music lovers usually listen variety of music during different times of the day. Therefore time is an important context to consider while recommending music playlists and should change depending on time of the day. Authors call it smart music playlist generator. User's profile is stored when users use application for the first time which includes personal data of user. User's likings are stored with timestamp. Extracting features of music accurately is the first step to improve performance of the system. Therefore this system extracts features from symbols of digital music using MIDI file format. Various pitch parameters are extracted from it which give information about melody in song which user may like. Even vocal properties of all artists in song are extracted as there is a strong relation between user's liking and vocal properties. These features and time context is input to playlist generator module which is based on ANN. It has two ANNs, one long term ANN and one short term ANN for reducing retraining time of ANN. To handle cold start problem, collaborative method is used. Experimental evaluation indicates that using time context, gives better performance than the system does not that consider time context. Use of other contextual factors like mood of the user can help in aligning recommendations to interests of the users.

Utility based recommender system for E-Commerce application is proposed in [52]. The proposed model is based on multi-attribute utility theory(MUAT). Content-based recommendations usually face cold start issue where as collaborative filtering techniques suffer from sparsity problems. Therefore a method is required to find user's preferences. Usually a decision is based on many decision attributes. Therefore MUAT is suggested in [52]. MAU function is built using Radial Basis Function Neural Network (RBFN). Simple multi attribute rating technique is used along with RBFN. It was tested in two different application contexts namely movie and notebook datasets. Results show that performance of RBFN is steady in many application contexts. But experiments are conducted on limited dataset, so the actual performance can be evaluated on real time large dataset. Even inclusion of environmental contextual attributes can further improve performance of the system.

Claudio Biancalana et al. [53] tried to improve accuracy of traditional CF using ANN and majority voting classification. Since current context of use helps showing interesting recommendations to user and improve overall accuracy of the system. Usually movie recommendation systems only make use of users and movies to predict ratings but context aware recommendation system proposed in [53] uses time as context along with user and movie for rating prediction. Generally users submit preferences to the system in short period of time. Therefore it is pertinent to give more weight to movies in specified time period in current movie recommendation systems. Ratings are grouped as per defined time interval and signal is constructed and samples are taken. There are two measures to the signal including number of movies watched in given time interval and second measures deals with successive intervals. Then a cross correlation function is used on the samples. Movies which form intersection of two sets formed by this new approach and original collaborative filtering approach are boosted by some constant value. The other part of the technique points to identifying particular member of household who has given the rating. The input tuple contains household, movie, rating and timestamp. System considers analysis of distribution of values by user, analysis of distribution of time intervals when the rating was given and analysis of movies to find users who have watched same movie. These three factors are consolidated using ANN based learning algorithm which picks up best combination of output. Output layer has three nodes where each nodes corresponds to value for each of the three factors. As system uses signal processing and neural network based learning algorithm, it improves the performance of existing recommendation systems. Incorporation of contextual factors other than time may further improve accuracy of the system.

In [54], a methodology for understanding preferences of viewers of digital TV channel programs based on ANN has been proposed. Useful information in TV Programs is extracted as 24 dimension feature vector with values consisting of action, sport, sci-fi, thriller etc. As number of dimensions are more, dimensionality reduction technique will transform them into three main genre namely relaxing, informative and emotional. These genre act as input nodes of single layer feed forward neural network along with two contextual nodes specifying days of the week having values weekday, weekend, holiday etc. and time of the day with values consisting of morning, midday, afternoon, evening and late night etc. ANN will classify programs as like or dislike for a particular user so that it becomes easy for recommender system to recommend them. For training ANN faster, extreme learning machine algorithm was chosen. Contextual modeling used in this methodology shows improvement in accuracy compared to non-contextual modeling. Contextual attributes other than time can also be tested with this model to evaluate their impact on accuracy of prediction.

The authors in [55] proposed context aware recommendation system for tourism. Current tourism applications consider queries having nearest tourist places. Actions taken by tourist highly depends on context within which his/her preference lies. Many contexts like location, time, weather, social media sentiments and personalization etc. are considered while designing application in their application. Meehan et al. [55] have suggested use of hybrid model using techniques like ANN, fuzzy logic, Principal Component Analysis (PCA). However, there is no suggestion of any method used to incorporate contextual attributes in the proposed technique. Further, there is no mention of appropriate preprocessing required for the data.

In [56], A hybrid recommendation system for various businesses is proposed. This approach uses content-based filtering along with deep learning neural network. The neural network was experimented with three types of activation functions namely tanh, relu and sigmoid. In all the three cases, it shows better accuracy than CF approach.

Zahra et al. [57] proposed a tour recommendation system using ANN and case based reasoning. It uses Multi Layer Perceptron ANN(MLP ANN). System recommends set of tours using MLP ANN and then based on user's response. It also modifies this set using case based reasoning if user gives negative response. Experimental evaluation on survey data captured in Tehran via developed application gives better accuracy of this technique than CF.

A music recommendation system using ANN is proposed in [58]. The ANN learns user preferences from past music sequences of users captured by the system. Those are used as contexts to give recommendation to the users. Experimental evaluation shows better accuracy and better ability to handle data sparsity than the baseline techniques selected for comparison.

A Context Aware Recommendation System for sequential recommendations is proposed in [59]. Sequential interactions are modeled using Recurrent Neural Network (RNN) that incorporates temporal context. Experimental evaluation shows better accuracy than selected baseline methods. Table 2 compares Context Aware Recommendation Systems based on ANN.

3.2. Genetic algorithm

Genetic algorithms are based on Darwin's theory of evolution and the concept of survival of the fittest. The algorithm begins with creation of random population of chromosomes considering it as a first generation. A fitness function will be defined for each chromosome and fit chromosomes will be selected to create next generation using crossover and mutation genetic operators. Once the termination criteria are satisfied, the algorithm terminates giving optimal solution to the problem. Therefore, genetic algorithms are usually used for optimization problems [60]. If we consider, recommendation as an optimization problem, genetic algorithms can be used to solve certain aspects of this problem.

An approach to personalize resource recommender using context history is proposed in [61]. Every user has certain habits or preferences which can be implicitly deduced using context history. This approach has two main layers namely semantic layer and fuzzy layer. Semantic layer characterized by ontologies is helpful to explain domain knowledge & fuzzy layer deals with ambiguity in context information. These two layers together give list of situations in descending order of certainty. Fuzzy layer indicates each rule using linguistic variables which can take certain

Table 2
Context Aware Recommendation Systems using ANN.

Reference	Main feature	Data source	Application domain	Contexts used	Context type	Future scope
[51]	• Use of long term and short term ANN to reduce retraining time	Music files uploaded on the server for experimentation	Music	Time	Static fully observable	Contexts like mood, demographics can be added to improve accuracy
[52]	• Application of Multi Attribute Utility Function with many decision attributes	KingNet movie data, Yahoo purchase data	E-commerce	Application context of movie and notebook	Static fully observable	System's environmental contexts like time, demographics can be added to improve accuracy
[53]	• Use of signal processing and ANN for rating prediction	CAMRa 2nd challenge movie data	Movie	Time	Static fully observable	Contexts other than time like location, companion can be tested
[54]	• Dimensionality reduction is used as preprocessing steps to convert 24 features to 3 genre.	TV Program guide loaded in the system	TV program	Time of the day, Day of the week	Dynamic fully observable	Other contextual information like companion, mood can be tested
[55]	• Use of ANN with many contextual attributes	Data generated by the created application	Tourism	Location, time, weather, social media sentiments	Dynamic partially observable	Use of appropriate pre-processing & contextual modeling technique can be investigated to improve system accuracy.
[56]	• Use of ANN with content based filtering to improve accuracy	Yelp academic dataset	Business recommendation	Location, time, reviews	Static partially observable	Comparison with approaches like MF or TF can give more insight
[57]	• Use of ANN with case based reasoning to improve accuracy	Data collected by survey in Tehran	Tourism	location, time, user mobility history	Dynamic partially observable	Use of contexts like companion, weather can also be tested
[58]	• Use of ANN to learn user preferences via past music sequences	Data collected from Xiami music	Music	user music sequences	Dynamic partially observable	Inclusion of contexts like mood and its impact can be tested
[59]	• Use of RNN to model sequential interactions	Movielens 1M, Amazon books	Movie, E-commerce	time	Static fully observable	Use of other contexts like location, companion can be tested for Movie recommendation

values explained by fuzzy set. Usually fuzzy set describe rules for generic user. In order to personalize values in fuzzy rules, genetic algorithm is proposed in this approach. Genetic algorithm is used to tune membership function of linguistic variables. Ciaramella et al. [61] took responsiveness as metric and compared their genetic algorithm based on approach with recommender system defined by experts in the domain. They found that responsiveness of their approach is better. Further testing in real time environment can reveal more features associated with this approach.

Recommendation problem is considered to be a multiple objective optimization problem in [62]. A model based on contextual information is formed and every factor influencing user's decision is set to have specific objective function to be maximized. All these functions are processed in parallel. Along each parallel path, genetic algorithm with the path's fitness function is used to select best candidates. Candidates from each path are combined and given to user. Based on user feedback, next iteration is set up until system learns user preferences. Appropriate use of contextual information is required to further enhance the system performance.

Dao et al. [63] proposed context-aware CF model using genetic algorithm for location based advertising. This model uses conventional CF and incorporate different contexts like location, day, time and interest in order to improve accuracy. It tries to find optimal values for context similarity for which it uses genetic algorithm. User's tastes are hard to detect as they evolve over time and depends on context. This model uses user's need type as one of the contexts and can take values as hedonic (product requirements having social or aesthetic utility), utilitarian (product requirements that eliminate problems) or both. Since there are multiple contexts used, it becomes tedious to calculate effect of all of them together on user's choice. Therefore genetic algorithm is used to find optimal value of context similarity. This model uses a new user–item matrix with contextual information unlike conventional user–item matrix used in CF. It also generates context similarity matrix to find similarity coefficients among different contexts. To generate this matrix, the model uses genetic algorithm. These coefficients are incorporated in Pearson coefficient to find user similarity in varied contexts using a modified Pearson coefficient. Empirical evaluation shows that this method performs better than other baseline methods like simple average, pure CF etc. More insight can be obtained if this model is tested in real time environment.

Irfan et al. [64] proposed cloud-based venue recommendation system using genetic algorithm and contextual parameters. This approach tackles cold start, sparsity and scalability problems faced by recommendation systems. In the preprocessing step, framework uses ranking phase makes use of Hub Average method to generate ranks for users and venues. Mapping phase calculates similarity among expert users generated by ranking phase using Pearson coefficient. Ranking phase does bi-objective optimization for famous venues and venue's closeness. The framework uses CF based and greedy bi-objective recommendation framework methods. Bi-objective vector optimization is done using NSGA-II which is based on genetic algorithm. Result analysis shows improvement in precision, recall and F-measure values compared to baseline algorithms. Exploring the use of online learning algorithms like contextual ϵ-greedy [65], CCMAB(Contextual Combinatorial Multi Armed Bandit) [66] may further improve the accuracy of venue recommendation.

A contextual similarity computation approach in recommendation systems using genetic algorithm is proposed in [67]. A user profile is created using demographic information. Content based user profile is also created considering movie ratings and movie genre. It then uses a hybrid user model and computes contextual similarity using genetic algorithm. It then finds set of relevant movies and gives recommendation. Experimental evaluation on LDOS-CoMoDa dataset gives better accuracy for the proposed approach than CF approaches.

Kumar et al. [68] propose a restaurant recommendation system using ANN and genetic algorithm. The initial phase includes building ANN which takes data including contexts like location from sensors of the mobile device. Genetic algorithm is used to optimize weights of ANN. Once the ANN is trained, it is used to predict restaurants and food items in the next phase. Experimental evaluation shows 98% accuracy on Chicago restaurant recommendation dataset.

A venue recommendation system using genetic algorithm is proposed in [69]. It uses location and time contexts together in the model for venue recommendation. The model uses context aware CF to predict rating. In this processes, optimization of temporal weights is achieved using genetic algorithm as temporal aspects help in personalized recommendations. Experimental evaluation shows better accuracy than only context aware CF. Table 3 compares genetic algorithm based approaches used in Context Aware Recommendation Systems.

3.3. *Ant Colony Optimization (ACO)*

This approach represents artificial intelligence systems which are inspired by behavior of ants to solve complex problems and is especially useful in optimization problems. It uses a concept of stigmergy which can obtain solution without direct communication. It is based on pheromone trails left by ants. It was formed as a stochastic model. For real life artificial intelligence systems, concept of evaporation of pheromone trail can help to explore new solutions. This way, the approach helps to handle exploration–exploitation trade-off [70].

An ACO based route selection approach for context-aware recommendation systems in map navigation applications is suggested in [71]. ACO algorithm can help user in selecting suitable route based on current location and contextual information of the user. A domain related ontology having contextual information is built for guiding ACO to select probable route. Therefore, authors named the algorithm as Semantic ACO (SACO) algorithm. For choosing routes under contextual constraints, restriction sets are built which contain contextual terms, their values and relevance with respect to context. A semantic score is computed and assigned to each item. There are three types of nodes defined in this approach. SequenceNode deals with sequence of contextual data. MetaNode handles metadata. ModelNode represents previously learnt sequence of contextual data. Experiments based on Time, location and activity type as contextual factors demonstrated the use of SACO in finding possible route by system by balancing the exploration and exploitation trade-off. Comparison with baseline algorithms like simulated annealing and tabu search shows that SACO outperforms them for large problem size. Also, the average time required to get solution using SACO is comparable with both the algorithms. Since this algorithm can be executed in parallel, scalability issue faced by recommendation systems is handled. However inclusion of more contextual data and their impact on each other in real time dataset can further test the effectiveness of the approach.

Mocholi et al. [72] proposed a model for context-aware recommendation for route finding by extending their SACO algorithm [71] using Allen-temporal operators. This helps making queries more specific. The knowledge base and contextual information is stored using ontology specifying semantic relationships in them. Every node represents contextual information and arc represents sequence in contextual information. Then a restriction set is made considering relevance value of context. After applying restriction set, model computes semantic assignment score. It then uses ACO algorithm to generate set of contextual sequence sorted by scores. Results show inclusion of few results for sample queries neglected by SACO algorithm.

An ACO based music playlist generation approach using ontology is proposed in [73]. In music playlist, sequencing of song tracks play a vital role. Therefore similarity between song tracks should be maximized when they are played in sequence. While doing so, restriction set imposed by the user(which can be treated as part of contextual information) should also be taken into consideration. The problem of recommendation is modeled as orienteering problem by the authors. The proposed solution includes use of SACO-OP which is ontology based ACO algorithm for orienteering problem. A restriction set is formed specifying restrictions that can be applied in recommendation search which is defined by ontology. Experimental results include computation of frequency and saturation for different ranges of semantic distances for specific restriction set. Results include restriction set with all folk song track, all trance song track, 50% folk and 50% trance song track and finally 50% folk song track and 50% track with artist named Javier Krahe(includes different ontology terms namely track type and artist). The algorithm shows promising results as far as saturation of solutions is concerned. But there is still a scope to test handling of cold start and data sparsity issues. Other measures apart from frequency and saturation like precision which are frequently used in recommendation system literature can be used for testing accuracy of results.

Viswanathan et al. [74] proposed semantic ACO based approach for finding relevant items. In recommendation systems, relevance of the recommended items is a key factor. Therefore, exploring semantic relationships between relevant entities is a crucial factor. Authors propose the use of semantic ACO algorithm to find these relationships effectively. Entities and their relations are expressed as a RDF graph and semantic scores of the edges of the graph is computed. Higher the semantic scores, higher is the probability of selection of the edges, hence increasing the chances of inclusion of relevant entities. Authors consider subsumption, context, popularity and rarity as parameters for deriving semantic score which decides relevance of entities. In subsumption, specialized entities are given more weightage than generalized entities in the hierarchy. There is a score associated contextual attribute like location. There is also a score pertaining to popularity of items. As rare relationships shows more interesting or unusual pattern, a score is associated with such rarity. Finally a semantic score is computed by addition of all these parameter scores. ACO is applied to choose optimal path between entities so that they become part of the relevance set. Experimental evaluation shows slight increase in average precision level compared to baseline algorithms. This algorithm can be further tested with real contextual datasets. If more contextual factors are incorporated, their effect on relevance set can be investigated. [75] proposes a feature selection strategy using fuzzy c-means and ACO algorithms. Since there can be many features available for different items to be recommended, subset of features, influencing the decision of recommendation must be selected. There can be scenarios where a movie may belong to more than one cluster with different values of membership functions for different clusters e.g. A movie can be comedy and romantic at the same time but the degree to which it belongs to comedy and romantic may vary. Therefore, fuzzy c-means algorithm is used to form clusters. Maximum value of membership function denotes

Table 3
Context Aware Recommendation Systems using genetic algorithm.

Reference	Main feature	Data source	Application domain	Contexts used	Context type	Future scope
[61]	• Use of genetic fuzzy system to personalize recommendation	Meeting data collected by Pharmaceutical consultants	resources in business	User habits in the form of contextual history	Dynamic unobservable	May have to be tested in real time environment
[62]	• Use of parallel processing of multiple objective functions	Simulation on Youtube data	Video	Youtube metadata information	Static fully observable	More testing of data with additional contexts like device type, network type etc. can help in effective usage of the system
[63]	• Empirical evaluation shows that this method is better than baseline methods like SimAvg, DayCF, TimeCF etc.	Data collected in the system from mobile users in South Korea	Mobile advertising	Location, day, time, interest	Dynamic partially observable	More testing of the system in real-time environment can be done to get insight of the challenges
[64]	• Improvement in performance compared to baseline algorithms like Matrix Factorization(MF), User based CF and Random Walk with Restart	Gowalla venue dataset	Mobile venue recommendation	Location	Static fully observable	Comparative study with techniques like kNN can be explored.
[67]	• A hybrid model using genetic algorithm to compute context similarity	LDOS-CoMoDa dataset	Movie	day type, location, end emotions, dominant emotions, mood, physical state	Dynamic unobservable	Implicit ways of capturing emotions and real time testing
[68]	• Use of genetic algorithm to optimize weights of ANN to improve accuracy	Chicago restaurant recommendation dataset	Restaurant	Location, type of cuisine, whether expensive	Static fully observable	Comparison of the accuracy with other methods like MF, TF etc. to get more insight
[69]	• Use of genetic algorithm to optimize temporal weights in context aware CF	Foursquare, ConcertTweets datasets	Venue recommendation	Time, location, demographic information	Static fully observable	Comparison of the accuracy with other methods like genetic algorithm optimized ANN [68] can give more insight

maximum influence of the feature on preference of the user. In order to maximize fuzzy values, ACO is used along with fuzzy c-means. After significant features are selected, those features are given as input to backpropagation neural network with sigmoid activation function in the hidden layer. The output layer determines ratings of the user. The algorithm shows slight reduction in the values of MAE in rating prediction for MovieLens dataset. But features are small part of contextual data. Other significant contextual information describing environment and its impact on user–system interaction is not taken into consideration in this approach. Extending this approach for incorporating such information may further improve accuracy of prediction.

An intelligent adaptive tutoring system recommending personalized learning objects is proposed in [76]. Different learners have different learning styles. Therefore, the content presented in online tutoring system must adapt to learner specific styles. Considering learning style as a context, the proposed model uses ACO to find suitable sequence of learning objects for all the learners. Simulation results show accuracy improvement in learning objects recommended to the learners. Similar system is also proposed in [77] where along with learning style, use of prior knowledge of students is also considered. But there are no experimental results provided in [77].

A social trust based CF method using ACO is proposed in [78]. The first phase of the model computes similarity using social trust among users. Then user graphs are created and using ACO the weights are updated. Using set of similar users and their associated weights rating prediction is done. The accuracy of the model is better compared to MF and CF based approaches used as baseline in the experiment. Table 4 compares Context Aware Recommendation System approaches based on ACO techniques.

3.4. Particle Swarm Optimization (PSO)

PSO is a stochastic optimization approach inspired by behavior of congregation of birds. It is used for optimizing continuous non-linear functions. When a congregation of birds randomly search for food in an area, all the birds are unaware of location of food. They will come to know the distance of food in successive iterations from the bird nearest to the food. Every bird represents a solution in the search space and is called *particle*. Every particle has a fitness value to be evaluated by fitness function. Every particle also possess velocity. The particles fly by the information received through current optimum particles. The process starts with group of random particles. Each one stores best solution it has obtained so far and is called as *pbest*. *gbest* is the best value obtained by any particle in population. Particles update their velocity and positions based on the values of pbest and gbest in each iteration using equations given in [79].

Zheng et al. [80] propose differential context relaxation technique for CF approach used in recommendation systems. Contexts are applied to different components at different stages. Hence, the technique is differential in nature. Inclusion of the contextual attributes imposes constraints on the data which serve as input to the different components of the algorithm. This indicates that contexts act as filters. For recommendation systems, we require optimal set of contextual factors influencing user's preferences. Hence this problem can be treated as finding relaxation of the

Table 4
Context Aware Recommendation Systems using ACO.

Reference	Main feature	Data source	Application domain	Contexts used	Context type	Future scope
[71]	• Alignment to user preferences due to inclusion of contextual data	Data collected in the system from 120 trips on 15 routes	navigation systems	Time, Location, Activity type	Static partially observable	Testing in real time contextual conditions needs to be explored
[72]	• Use of Allen Temporal Operators and ontology to find nearest neighbor routes	Data collected in the system from 120 trips on 15 routes	navigation systems	Time, Location, Activity type	Static partially observable	Testing in real time contextual conditions needs to be explored
[73]	• Use of ontology to find optimal playlist using SACO	Data collected by scraping multimedia audio databases	Music	Time, music type, artist	Static fully observable	Involving contextual attributes like companion, mood can be investigated
[74]	• Improvement in precision due to context weight consideration	SWETO test bed and developed test bed	Movie, music	Location	Static fully observable	Only the architecture is proposed. Testing in real time contextual datasets needs to be explored
[75]	• Slight reduction in Mean Absolute Error	Movielens 1M dataset	Movie	Features of the items, time	Static fully observable	Contextual attributes like location, companion can be included to enhance the working of the system
[76],[77]	• Use of ACO for personalized learning object sequence	Simulation	Education	Learning style	Static fully observable	Real time testing of such system
[78]	• Use of ACO for rating prediction using social trust	FilmTrust, Epinions datasets	Movie	Social trust	Static partially observable	Use of modal contexts like mood can be explored

constraints to balance accuracy and coverage of recommendation. Therefore, this is an optimization problem. Zheng et al. [80] use Binary PSO (BPSO) to find the optimal solution. They use Resnick's algorithm for Collaborative Filtering and modify it for incorporating context. They also model relaxation as a process of binary selection with value 1 indicating filter is applied using the specified constraint. They use BPSO to minimize RMSE in prediction of user preferences. Position of the particles in space specifies the set of constraints. Experiments were carried out on AIST food preference dataset where the degree of hunger was considered influential contextual attribute and demographics, gender, food type etc. as context-linked attributes. Experimental results show that this algorithm has reduced RMSE compared to normal CF and pre-filtering approaches. Further tests with other algorithms apart from CF and with more contextual attributes can be investigated to check the effectiveness of the technique.

Differential context relaxation algorithm in [80] increases data sparsity problem because of the constraints getting relaxed after the application of filtering technique. Hence, [81] introduces differential context weighting scheme where contribution of each context is weighted. It requires optimal set of weights for each context. It identifies influential context variables. Differential context relaxation works well when contextual data is dense. Even algorithm components can be dependent. Given a target context, weightage of the rating issued in some other context can be found using weighting vector. Authors have used weighted Jaccard similarity as a measure to compute similarity between contexts. A set of similarity threshold was used to filter ratings. Context values below threshold are ignored. Weighting vectors give different weights to each context for showing influence of few contexts on user preferences. PSO is used to minimize RMSE. Position in space represents set of weights. [82] found PSO useful and have applied it for feature weighting but [81] use it to incorporate contextual data. The algorithm was evaluated on AIST food preference dataset which is having dense contextual information and movie dataset with sparse contextual information. Experimental evaluation shows that differential context weighting outperforms CF, pre-filtering and differential context relaxation for RMSE on both the datasets. Coverage for differential context weighting is more for AIST food preference dataset but less for movie dataset. This can be due to objective function which is to minimize RMSE. So coverage is compromised. If number of contextual attributes are more, time to learn is more. So the algorithm takes more time. If contextual attributes are dense, weight calculation takes time. Efforts can be put in for improving coverage for sparse contextual data. Even use of semantic similarity can be evaluated. Other methods like MF can also be tested.

Kataria et al. [83] propose hybrid model for movie recommender system using PSO and fuzzy c-means. Data has two parameters namely movie id and rating given by user for that movie. Type division method is used to distribute ratings as per the genre of the movie. K-means algorithm is used to generate cluster centers for entire movie data. PSO is used on this data to optimize cluster centers. Output of this step is given to fuzzy c-means algorithm. As a movie may belong to more than one genre, fuzzy c-means generate final cluster centers along with membership values of each movie. As per previous user ratings in each division, a combined rating is produced to get final predicted rating. For experimental evaluation, a Movielens dataset was used. It was divided into 19 types of movies based on type division method. Mean Absolute Error was used as a metric for checking the accuracy of predicted rating. Experimental evaluation shows that this method has around 3% less MAE compared to baseline algorithms used like PCA-GAKM, PCA-SOM, SOM-cluster, k-means, GA-KM, PCA-KM. Genre is used as context to determine user preferences. If more contextual attributes which specify user–system interaction can be included, this method may show further reduction in MAE thus improving the performance of the system.

A movie recommendation system using PSO is proposed in [84]. A dataset is divided based on different contextual criteria. Similarity is computed between these divisions. Similarity aggregation is performed using PSO and then relevant items are selected and recommendations are given to the user.

Similar to the system in [84], an ANN based system is built in [85]. ANN is used to predict the user preferences in the contextual conditions. The learning of ANN is done with the help of PSO.

Experimental evaluation shows the improvement in accuracy of this model over only ANN based models. A music recommendation system using CF, depth first search, Bellman–Ford algorithm and PSO is proposed in [86]. Using the user's listening pattern stored in logs, contextual information is extracted. Contextual information is split into three categories.

- item-context which contains information about item say artist, genre etc.
- user-context which has information about user preferences, feedback etc.
- decision-context which specifies information about attributes making contribution in decision making ability of the users. This can influence preferences of the user.

Item-context is used to find similar users using CF and thus generates the possible playlist. All the available contextual information is used to to form a graph based structure for each user maintaining all the contextual information with different paths which is traversed using depth first search and Bellman–Ford algorithm. PSO is used to optimize the ranked list to improve performance quality of recommendation. Experimental evaluation was carried out on Last.fm dataset having log entries of users listening to variety of music. Recall which is used as a performance metric shows higher values for this approach compared to random, most popular and pure Singular Value Decomposition(SVD) approaches. Even this approach shows comparable performance with approaches like complex number representation link prediction, music genre weight based method, CF, user–item graph and preference graph. This approach makes use of only time as decision-context. Further improvement is possible by including more such contexts like demographics, mood etc. Since music recommender systems are usually online systems, this approach must be tested in real time with addition metric like Click Through Rate(CTR) etc. to evaluate the performance.

Dixit et al. [87] propose a PSO based Context Aware Recommendation model. In this approach, a k-prototype clustering is used for grouping contextually similar users. PSO is applied to closest cluster to determine the contribution of different contexts. The experimental evaluation shows better accuracy compared to non clustered and other PSO approaches. Table 5 compares Context Aware Recommendation Systems based on PSO technique.

3.5. Miscellaneous bio-inspired algorithms

An Artificial Immune System(AIS) is a computational model inspired by natural immune system. Natural immune system is a defense mechanism of the body against intruder organisms. Immune system iteratively learns difference between intruder cells called antigens and body's own cells. When the intrusion takes place, few of the immune cells identify the pattern in antigens and create lot of antibodies to neutralize or kill those antigens. Few of these cells are stored in memory to tackle any subsequent similar intrusion rapidly. AIS is based on same computational model used by natural immune system to generate immune networks and immune response. Detailed explanation and applications of AIS can be found in [88].

An AIS based approach for movie recommendation is proposed in [89]. CF is commonly used method for movie recommendation. Aim of this approach is to enhance the output of CF. The proposed approach has two main phases namely training phase and prediction phase. Training data is used by AIS to build immune network. Antigens in the immune system use information about users like gender, age etc. Apart from that they also contain rating information. Antibodies are generated and immune network is formed. After this, affinity scores of antigens are calculated. A new immune network is formed by identifying the antigen with worst affinity score. For basic data like gender, Hamming distance is used to find affinity. For rating data, modified Pearson Correlation coefficient is used to find affinity value. Prediction of a rating is calculated by finding similarity between different users in a group and different items in the group. Experiments are performed on MovieLens and EachMovie datasets. MAE, precision, recall and F1 measure are used as metric for finding accuracy of the approach. Results show improvement in accuracy of prediction compared to state of the art methods like CF, RSVD etc. This approach handles data sparsity issue but cold start and scalability problems are not addressed. Inclusion of real time contextual information apart from user and item specific information may still improve the accuracy of the system.

Palacios et al. [90] propose a framework called POST-VIA 360 for context aware mobile recommendation in the tourism industry. Tourism is considered to be one of the leading industries in the world. Since suggestions on different tourist destinations, nearby restaurants etc. can be given to users, recommendation systems play a vital role in tourism industry. POST-VIA 360 is a recommendation system developed to help tourists before the visit, during th visit and after the visit by giving suggestions. The architecture of the system has three main layers namely user interface, business logic and persistence. User interface has a web portal or mobile interface to interact with the system. Business logic layer has AI based recommendation engine, CRM engine, opinion mining engine and travel experience engine. Travel experience engine does the evaluation of travel experiences of the users. CRM engine has features of sales, marketing and service management for customers. Therefore, ontologies from the persistence layer can be used to improve the performance. Recommendation engine uses Artificial Immune System to recommend points of interests to the tourists. Opinion mining engine uses ontologies and natural language processing. It is used to improve the quality of recommendations. Persistence layer stores all the relevant data using semantic technology and Geographic Information System(GIS). Validation of results is done by comparing POST-VIA 360 system and suggestions given by group of experts in the field who served as guides or tourism experts in the selected area. Experiments are carried out on sample of tourists visiting specific area. Precision, recall and f-measure are used as accuracy metric for comparison and this approach shows only a slight improvement in the accuracy. Better representation of data or inclusion of more data may improve the accuracy. Also inclusion of more contextual information may improve the accuracy of the system.

An Artificial Bee Colony (ABC) is a meta-heuristic algorithm. It is inspired by the behavior of honey bees in search of food source. There are two categories of bees namely employed and unemployed bees. Onlooker bees and Scout bees are types of the unemployed bees. Initially, all scout bees find food source locations. After that food sources are exploited by employed and onlooker bees. Once those food sources are exploited, the employed bees which are exploiting those resources will become scout bees to explore new food sources. Location of food source indicates possible solution and fitness of the solution is calculated based on quantity of food source. So the algorithm has employed bees, onlooker bees and scout bees as three main phases. The algorithm continues till required constraint like number of iterations is satisfied. This is especially used to solve optimization problems [91].

An improved k-means clustering algorithm using ABC is proposed in [92]. Users with similar interests are grouped in same cluster based on distance from centroid of clusters formed in k-means algorithm. After clustering, an estimate of rating for an unrated movie by a particular user is made. ABC algorithm is

Table 5
Context Aware Recommendation Systems using PSO.

Reference	Main feature	Data source	Application domain	Contexts used	Context type	Future scope
[80]	• Reduced RMSE	AIST context-aware food preference dataset	Food	Degree of hunger, gender, demographics, food type	Dynamic partially observable	Investigation can be using algorithms other than CF along with the PSO
[81]	• Reduced RMSE	AIST context-aware food preference dataset, data related to movies generated by survey	Food, Movie	Degree of hunger, gender, demographics, food type for food data & time, companion, location for movie	Dynamic partially observable	Comparison with more techniques like MF instead of standard CF can lead to more insight
[83]	• Improvement in prediction accuracy	Movielens 1M dataset	Movie	Genre of the movie	Static fully observable	Other contextual attributes like time available in the data can be used to improve accuracy
[84]	• PSO is used to get more accuracy than only CF	Yahoo! Movie dataset	Movie	Direction, action, story and visual effects	Static partially observable	Considering the contexts like companion, location in the same setup
[85]	• PSO based ANN for getting accuracy	Yahoo! Movie dataset	Movie	Direction, action, story and visual effects	Static partially observable	Comparison with model in [84] may give more insight
[86]	• Better recall value	Last.fm dataset	Music	Time, features of music	Static fully observable	Testing with dynamic and partially or fully observable contextual conditions like demographics, mood can be investigated
[87]	• Use of PSO to get contribution of contexts	LDOS-CoMoDa, Incar-Music	Movie, Music	day type, location, end emotions, dominant emotions, mood, physical state for movie & driving style, landscape, mood, roadtype, traffic condition, weather, time etc. for music	Dynamic partially observable	Real time testing with implicit capture of mood

applied on the clusters formed in k-means to evaluate the fitness function and explore new user groups with similarity. This may require reorganizing clusters of users. Hence, this process helps in improving results obtained from k-means in each iteration. Experiments were performed on MovieLens dataset and accuracy was measured using precision and recall as metric. This approach shows slight improvement in performance compared to selected baseline algorithms. Slight reduction in MAE was also observed compared to selected baseline algorithms. Use of many contextual attributes may further improve the performance of this technique.

Another category of algorithms called bat algorithm is also a topic of research in recommender systems. Bat algorithm is a kind of heuristic algorithm which is inspired by behavior of bats in darkness. Echolocation is a property associated with microbats to find their paths, prey and obstacles in complete darkness. They can even identify different types of insects accurately. The assumption for bat algorithm is bats fly with a random velocity at a particular position with some frequency of emitted sound wave and adjust the rate of emission in the range of [0,1] based on distance from the target object. It is also assumed that the loudness of emitted sound wave varies from large positive value to a predetermined constant minimum value as per the distance of a microbat from the prey. After initializing bat population, velocities, frequency, rate of emission and loudness, the algorithm generates new solutions by adjusting frequency and updating velocities. It exploits best solution if random number generated is less than rate of emission. It generates new solutions by random flying. If random number generated is less than current loudness and objective function at a location is than the objective function at target location, then accept the new solution. Increase the rate of emission and decrease the loudness as the bat approaches the prey i.e. target location. By this process, the algorithm tries to find optimal solution to a given problem [93].

Yadav et al. [94] propose incorporation of bat algorithm in CF to improve the accuracy of the system. CF based systems do not consider the importance of different features of items while predicting ratings for the users in the form of weights . The proposed method uses the bat algorithm along with CF to determine these weights so that better cluster of similar users can be formed. A subset of Jester dataset was used to evaluate the performance of this technique against ABC. On the selected subset, bat algorithm perform better than ABC as observed by MAE, precision, recall and F1 score values. Still rigorous evaluation with large dataset can decide actual accuracy in the prediction. More contextual information can be useful in improving the accuracy of the system.

Comparatively a recent algorithm called Artificial Algae Algorithm(AAA) is also a heuristic approach inspired by photosynthetic behavior of algae. AAA has three main phases. The first phase is an evolutionary process. In this phase, under sufficient light and nutrition, algal colony undergoes mitosis and produces two new algal cells in stipulated time. The growth kinetics of this process is calculated using Monod model. If algal colony cannot get sufficient light and nutrition, it survives for a while and then dies. The second phase is adaptation phase. If algal colony cannot grow in environmental conditions, it tries to adapt itself to environment by making changes in dominant species of the colony. Adaptation parameter decides the occurrence of adaptation at specific time. The last phase is helical movement. As friction surface of algal cell increases, frequency of helical movement increases by increasing local search ability. The movement is proportional to energy of cell and energy depends on nutritional intake at specific time. So if cell is close to water surface it has more energy and has more ability to move inside water. If friction surface is less, their movement distance is longer. AAA captures mathematical model of this movement in this phase. Experimental results show comparable results to other bio-inspired algorithms and in certain situations show better results than them. But this algorithm requires more parameters compared to few bio-inspired algorithms like ACO [95].

An AAA based recommendation system is proposed in [96]. In the first phase, for every user, a cluster of similar users is formed using a similarity measure proposed by the authors. Fuzzy c-mean algorithm is used to form cluster. Number of co-rated items in the for each user in the cluster is also calculated. In the next phase, if two users belong to same algal colony, then number of their co-rated items is recalculated. Even similarity between users is further computed and refined. The process is repeated for all users in the cluster. Experimental evaluation was done on MovieLens 100 K, MovieLens 1M, Jester and Epinion datasets. Accuracy in prediction of ratings was measured using MAE, precision and recall. Accuracy of proposed technique is compared with CF with Pearson correlation coefficient, weighted Pearson correlation coefficient, scalable Pearson correlation coefficient and multi-level CF. Results show reduction in MAE using the proposed approach over selected baseline algorithms. Precision is high only for MovieLens 1M dataset whereas recall value is slightly better only for Jester dataset. Use of more contextual attributes like location, mood etc. may improve the accuracy of the system. After inclusion of many attributes and parameters for AAA system, scalability and computation cost is a big challenge for this system. Selection of parameters for AAA to fine tune the above results can also be experimented. Table 6 gives comparison of different bio-inspired approaches discussed in 3.5.

4. Statistical computing techniques in Context Aware Recommendation Systems

Statistical computing techniques are variety of statistical tools which can be applied to data to learn from them. Using these techniques help in understanding the nature, distribution and characteristic of data which can be used in different learning tasks. These techniques are broadly classified as supervised and unsupervised techniques. Supervised techniques deals with building a statistical model to predict the output from one or more input fields. In unsupervised technique, there are no supervised outputs. So they are used to understand nature and relationships among data [17,18]. In this section, we will take a brief review of statistical computing techniques used in Context Aware Recommendation Systems.

4.1. Matrix Factorization (MF)

MF is a fundamental technique in linear algebra which is used for factorizing or decomposing large matrix into smaller matrices of specific properties. This technique is widely used in recommendation system literature due to nature of data such systems handle. A typical recommendation system deals with user–item rating matrix as primary data and tries to predict ratings of unrated items for the users. Since a real life recommendation system deals with large users and large items, this matrix becomes very huge. Therefore, MF techniques are used to reduce the dimensionality of data for faster calculation without losing on important information. These techniques support many supervised and unsupervised learning methods used in recommendation systems [97]. This section takes a brief overview of usage of MF techniques used in Context Aware Recommendation Systems.

Liu et al. [98] proposed incorporation of time and social network behavior in MF models. Interests of users keep on changing over a period of time. Hence, time is one of the important contexts. Standard MF models used in recommendation systems do not consider time as a factor. Therefore, time dependent data weighting and time aware models are proposed in [98]. In time dependent data weighting, more weight is assigned to a recent rating than the older one as recent rating depicts current inclination of a user. A temporal relevance factor is introduced in the standard neighborhood model used in CF. Since the relevance of the item decreases over a period of time, temporal relevance factor is modeled as exponentially decaying function. For considering temporal dynamics, time-aware modeling based on discrete time modeling approach is used. The entire time frame is divided into discrete time intervals and ratings are divided into different sets based on time interval in which they are obtained. Two models are proposed for time-aware recommendation system. One of them is tensor factorization where a rating matrix has three dimensions namely user, item and time interval. This matrix is used in MF model. The second approach is sequential MF in which for each time interval, we have a separate user, item pair which is used in rating prediction. It is observed that sequential MF performs better than other MF models used as baselines. For considering users of similar interests, social network aware models are used. One of them is collective MF model. In this model, if two users are connected in social network graph, then their rating matrices are jointly factorized. The other approach is Network Regularized Matrix Factorization (NRMF) model. If two users belong to same socio-demographic domains or have similar behavior, there are more chances that they have strong link on social network graph. Therefore, weights are given for each link on social network graph and those weights are incorporated in MF model. Experimental evaluation shows that NRMF model performs better than Standard MF model and collective MF model. Here, temporal and social MF models are considered separately. Hence, there is a scope to evaluate performance of both the models together. If more contextual attributes are added, incorporation of them in MF model is also a challenge.

An approach to give movie recommendations based on mood of the user is proposed in [99]. There is an attempt to introduce movie mood as context in the conventional CF based approach. A new similarity measure called mood-specific movie similarity is introduced in this technique. A mood similarity matrix is calculated based on specified mood tag. This measure is different than mood-based similarity which gives general similarity between movies based on their mood properties. A joint MF method is proposed using mood specific similarity measure in standard MF. Experimental evaluation was done on MoviePilot mood track dataset using popularity based recommendation, Random Walk with Restart, Matrix Factorization, Joint MF with movie-based similarity and the proposed approach. Mean Average Precision(MAP) and Area Under the Curve(AUC) are taken as evaluation metric. The proposed approach performs slightly better than other mentioned techniques. Inclusion of other movie specific contexts in the proposed model can be further investigated.

Baltrunas et al. [100] suggested a MF approach called CAMF for Context Aware Recommendation Systems. There are three aspects of the model. First one is more generic model which assumes that every contextual situation will have same impact on rating irrespective of the item. This model is called CAMF-C and has only one parameter for each contextual situation which is usually the value of contextual attribute. In the second aspect, there is one parameter for every contextual situation and every item. This leads to improved accuracy at the cost of increasing complexity of the system. This model is called CAMF-CI. The third aspect is

Table 6
Context Aware Recommendation Systems using miscellaneous bio-inspired algorithms.

Reference	Main feature	Data source	Application domain	Contexts used	Context type	Future scope
[89]	• Improvement in prediction accuracy	Movielens 100K dataset	Movie	Item and user features	Static fully observable	Can be improved by using real time contextual attributes like time available in Movielens 10M dataset
[90]	• Slight improvement in prediction accuracy	Data gathered using POST-VIA 360 system about tourist attractions in Spain	Tourism	Location, situation	Static fully observable	Use of more contextual attributes like companion can be investigated
[92]	• Use of ABC to improve performance of k-means	Movielens 1M dataset	Movie	Features of the movie	Static fully observable	Use of contextual attributes like time available in Movielens 10M dataset can be explored
[94]	• Performance of bat algorithm is better than ABC algorithm on subset of Jester dataset	Jester-1 dataset	Joke	User history	Static fully observable	Real time context inclusion like mood of the user while listening joke can be explored
[96]	• Reduction in MAE in prediction of ratings	Movielens 100K, Movielense 1M, Jester	Movie, joke	Features of the items	Static fully observable	Use of contextual attributes like mood, companion etc. can be incorporated to further improve accuracy

between the two extremes which has one model parameter for each contextual condition and item group say per genre of music. A parameter which decide deviation of ratings from standard MF is called baseline. In standard MF, baseline and interaction between context and items is added to form CAMF. Experimental evaluation on Tourism and music datasets considering MAE in prediction as metric shows slight improvement in accuracy compared to standard MF. There is still a scope to add dependencies among different contexts and test the effectiveness of the model.

A new MF based algorithm is proposed for Context Aware Recommendation Systems in [101]. Standard MF does not consider contextual information into consideration. Tensor Factorization can consider contextual information but the computational cost is high. So by establishing fuzzy mapping between context attributes and latent factors used in MF, Fang et al. have proposed new approach which considers contextual attributes and has less computational cost. Tags for movies can be treated as contextual attribute and SVD is used as MF method. Therefore, this technique was named as Tag SVD(TSVD). Even release time of the movie can be taken as context. Hence, technique incorporating tag and release time is called Time Tag SVD(TTSVD). Experimental evaluation on MovieLens 1M dataset using MAE, RMSE and half life utility as metric shows improvement in accuracy compared to selected baseline algorithms. Dependencies in contextual attributes and their impact on user preferences may also be incorporated to further improve accuracy of the model.

Shi et al. [102] extended their work on mood-specific movie similarity described in [99] to include plot-keywords in the model. Plot-keywords of the movie also give indication of similarity between movies as two movies having similar plot may have similar impact on viewer's mind. So movie plot can also be treated as contextual information. Therefore, in their proposed approach in [99], Shi et al. added plot keyword based movie similarity as a new similarity feature along with mood specific movie similarity. A joint MF is calculated to find recommendations. Latent factors of users and movies are used here also for prediction. Experimental evaluation on MoviePilot mood track dataset with MAP as metric indicates better performance of proposed method compared to popularity based recommendations, random walk with restart, standard MF, joint MF with mood based similarity and joint MF with mood specific similarity. But incorporation of more contextual attributes may lead to more similarity measures hence increasing computation overhead unless executed in parallel which may have to be tested.

Zheng et al. [103] proposed Sparse Linear Method(SLIM) based MF approach which is different than context aware MF approaches. In SLIM model, rating of the user on an item is given by aggregation of ratings on other items rated by that user. This approach is called SLIM-I as it uses aggregation between items. For aggregation between users, similar approach called SLIM-U is used. Incorporation of contextual attributes in SLIM model is called Contextual SLIM(CSLIM) which is proposed in [103]. Usually, contextual ratings given by user are sparse. Therefore, non-contextual ratings given by user are also considered. Aggregated contextual rating deviation matrix is used while computing predicted rating using CSLIM. Like standard SLIM, CSLIM also has variants namely CSLIM-I, CSLIM-U, CSLIM-I-CI, CSLIM-I-CU, CSLIM-U-CI, CSLIM-U-CU, CSLIM-I-C, CSLIM-U-C. Experimental evaluation was done on food, restaurant and music datasets. MAP and precision were used to measure accuracy. CSLIM models perform at par and sometimes better than context aware MF and Tensor Factorization methods. Further experiments may be performed to measure accuracy in different datasets with varied context density.

A model incorporating Convolutional Neural Network (CNN) in MF is proposed in [104]. To tackle data sparsity problem faced by recommendation systems, this takes domain related documents in consideration. Along with latent factors of users and items, contexts are also added using CNN. Hence, this approach is called convolutional MF. CNN has embedding layer, convolutional layer, pooling layer and output layer. Output is document latent vector. Parameters of document latent vectors model, user latent vector model and item latent vector model are optimized using maximum posteriori estimation method. Experimental was done on MovieLens and Amazon datasets. For MovieLens dataset, plot summary and other document related data were captured from IMDB dataset. Experiments were carried out on different levels of sparsity and proposed method performs around 3% better than probabilistic MF, collaborative topic regression and collaborative deep learning. Use of more contextual attributes (like time, companion in case of movies) may further improve the performance of the system. Table 7 gives comparison of MF techniques used in Context Aware Recommendation Systems.

Table 7
Context Aware Recommendation Systems using MF.

Reference	Main feature	Data source	Application domain	Contexts used	Context type	Future scope
[98]	• Consideration of time and social aspects in MF	CAMRA 2010 challenge dataset	Movie	Time, Social behavior	Dynamic partially observable	Time and social behavior models are considered separately. Investigating the combined effect is needed.
[99]	• Use of mood-specific similarity measure improving accuracy	MoviePilot mood track dataset	Movie	Movie mood	Dynamic unobservable	Use of latent knowledge of other contexts like companion, genre etc. can be experimented to improve accuracy
[100]	• Use of context aware MF improving accuracy of standard MF	MovieAT dataset and synthetic dataset generated from Yahoo webscope movie data	Movie	Time, social, opening weekend	Static partially observable	Considering dependency among context attribute can improve accuracy
[101]	• Use of movie and time tag using SVD as MF method	Movielens 1M	Movie	movie tag, release time	Static fully observable	Dependency among contextual attributes and their impact on performance can be explored
[102]	• Use of plot keyword and movie mood similarity measure in MF	MoviePilot mood track dataset	Movie	movie mood, plot keywords	Dynamic partially observable	Use of other relevant contexts like companion can be explored
[103]	• Use of sparse linear method for contextual MF	Incarmusic dataset for music, Food data collected from survey in [105], Restaurant data collected from system developed in [34]	Music, food, restaurant	driving style, landscape, mood, roadtype, traffic condition, weather, time etc. for music; hunger, temperature and fatigue for food; day of the week and location for restaurant	Dynamic partially observable for music, dynamic partially observable for food and static fully observable for restaurant	These datasets are sparse in nature so use of effective pre-processing technique can be investigated
[104]	• Use of CNN in MF improving accuracy	Movielens 1M, Movielens 10M, Amazon instant video dataset	Movie	words in reviews, movie plots etc.	Dynamic partially observable	Use of contexts like location can further improve accuracy

4.2. Tensor Factorization(TF)

A tensor is a multidimensional array or matrix with higher dimensions [106,107]. Since inclusion of contexts can lead to multidimensional problem space, TF can be used to predict user preferences in Context Aware Recommendation Systems. TF is usually a higher order extension of MF [106].

Maroulis et al. [108] propose context aware TF model for location based area of interest recommendation to mobile users. The initial phase of the model uses location information to form a tensor. The next phase of the model uses Stochastic Gradient Descent for computing latent factors for the users, areas of interest and context. Using this knowledge, a new tensor is formed to provide recommendations.

A probabilistic TF model is proposed in Context Aware Recommendation Systems in [109]. This model allows consideration of all the context simultaneously while modeling the system which is not possible with probabilistic MF. This model uses PARAFAC [106,109] tensor decomposition technique. Experimental evaluation shows improvement in accuracy of recommendations compared to probabilistic MF.

Luan et al. [110] propose a partition based collaborative TF approach. collaborative TF also considers correlation between two points of interest based on location. If the tensor is big and sparse then the degree of correlation between entities suffer. To overcome this, a tensor is partitioned into sub-tensors and then the collaborative TF is applied. Experimental evaluation shows good accuracy of results compared to MF with time-slicing and collaborative TF.

A travel recommendation system by using emotions and user behavior is proposed in [107]. This approach uses contextual pre-filtering and TF in the recommendation process. Based on context in consideration, pre-filtering technique filters out relevant results. Using those results TF is applied emotions and behavior. The results are compared with TF using emotions and TF using behavior. Experimental evaluation shows TF using emotions and behavior gives better accuracy than the other two approaches. Table 8 gives comparison of TF techniques used in Context Aware Recommendation Systems.

4.3. Bayesian theory and learning

Bayes theorem is a widely used statistical method for supervised learning because it has a prediction accuracy comparable to other state of the art techniques. Its a probabilistic model which learns conditional probability from training data. Park et al. [111] propose recommendation approach using fuzzy systems and Bayesian network considering utility theory. Context information is obtained from sensors and various sources on the internet. Preprocessing of this information is done using fuzzy systems which generate fuzzy membership vectors. Bayesian network is a probabilistic model which is used here to infer context from available information in fuzzy membership vectors. After contexts are inferred, final score is computed based on preference of the user in context using utility theory. Recommendations are given based on these scores. Experiments were carried out on 322 music creations collected from the internet and satisfaction level

Table 8
Context Aware Recommendation Systems using TF.

Reference	Main feature	Data source	Application domain	Contexts used	Context type	Future scope
[108]	• Use of context aware TF with Stochastic Gradient Descent to learn latent factors	Foursquare dataset of New York and Tokyo	Mobile location based Point of interest recommendation	check-in location, social	Static partially observable	Use of factors like weather can be explored
[109]	• Use of probabilistic TF which takes all contexts into consideration in system modeling	Epinions dataset, Movielens 20M dataset	E-commerce, Movies	number of reviews, user's average rating, average rating of the item for Epinions dataset & tags and time in Movielens	Static fully observable	Use of contexts like location, companion can be explored in this setting for movies
[110]	• Use of partition based collaborative TF to improve accuracy in sparse data conditions	Data collected from Weibo & DianPing point of interest websites	Mobile location based Point of interest recommendation	check-in time, location, point of interest	Static fully observable	Comparative study with context aware TF in [108] to get more insight
[107]	• Use of emotions and behavior using TF along with pre-filtering to improve accuracy	Location is collected from Wikipedia, ratings from TripAdvisor, emotions from synthetically generated dataset using CARSkit library	Tourism	location, emotion	Dynamic partially observable	Testing of the model by considering emotions in the real-time environment

of the users by recommended items was measured. Proposed approach works better than only Bayesian network. Further testing may also be done using metric like MAE or RMSE. Inclusion of dynamic contexts may also be tested in proposed experimental setting.

Location based recommendation approach is proposed for mobile devices using Bayesian network in [112]. Contextual data are collected by various means like user profile, sensors in mobile devices, websites etc. The contextual attributes are preprocessed and are given as input to Bayesian network. Bayesian networks create conditional probability tables using expectation maximization strategy and finally the results are displayed using a map interface on mobile devices. Experimental evaluation involved recommending restaurants based on class, price and mood which are inferred by Bayesian network. Experiment was conducted on four users in a part of Korea and probability distributions of restaurant class, price and mood of the user as inferred by Bayesian networks are presented. They are found to be aligned to user preferences. But still there is a need to exhaustively test the system with large number of users.

Pomerantz et al. [113] propose context aware recommendation system for movies where certain contexts depend on other contexts and can be inferred. The approach of considering each user as different user in different contexts will not work as user's ratings in different contexts can be related. Each time a new context is added , there would be no ratings available in that context. Hence, hierarchical Bayesian model is used to solve this problem. It has the capacity to predict rating in a context even if small number of ratings are available. Weights are estimated for each context. Even noise is estimated for every user using expectation maximization technique. PCA is used for reducing dimensionality. Experiments were carried out on the Recommendz movie dataset. Performance of four models namely nearest neighbors, separate users, hierarchical Bayesian and hybrid model combining k-Nearest Neighbor and expectation maximization are compared. Experimental evaluation shows that hybrid model performs well. Still investigation on proper modeling and fusing contexts with content based systems can be carried out.

Bayesian classification based model is proposed for video recommendation for mobile devices in [114]. Due to various resource restrictions of mobile devices, selection of appropriate content from a large data source becomes a problem. Therefore, recommendation of relevant content considering user's context is utmost important. In the proposed architecture, user's profile information is stored on recommendation server. Recommendation server maintains profile database having profiles of all the users of the system. Actual videos are retrieved from video streaming sites and appropriate recommendations are given based on user's context and preferences. User feedback is explicitly taken and contexts like time, location and mood are considered for the recommendation purpose. Bayesian classifier is used to find probability of liking a video given a specified contextual information and video content. Further improvement can be possible by using context relaxation strategy especially when less preferences are available in a given contextual scenario. Experimental evaluation of the proposed architecture on real time datasets must be done to find accuracy of the system.

Gupta et al. [115] propose context aware recommendation system model based on Bayes theorem and SVD. Since all contextual attributes are not of the same type, prediction of their behavior based on covariance is very difficult. All contextual attributes may not play significant role in preference prediction. Therefore, computation cost of preference prediction can be reduced by selecting only significant attributes. Naive Bayesian classification is used to find significant attributes. On the selected attributes with high significance, SVD is used to find attributes associated with high rating of entity. Since SVD is a MF technique, it helps here in dimensionality reduction. Simulated results on limited number of users and limited destinations for traveling application show that this technique helps in aligning recommendations to user preferences. But testing in real time environment may help in exploring advantages of this approach thereby further improving accuracy of the system.

Probabilistic model for context aware recommendation systems using Bayes theorem and BPSO is proposed in [116]. A hierarchical representation of the users and items is assumed in the model and effect of adding contextual attributes is investigated. Top level of hierarchy represents a generic level context whereas the bottommost level represents most specific contextual situation. Accuracy is compromised at the generic context level and specific contexts may lead to sparsity issue. The trade-off between the two is handled in the proposed approach by adding contexts at intermediate nodes of the hierarchy. The proposed approach uses Bayesian Flexible Mixture Model (BFMM) [117] to incorporate contexts in user and item latent classes. Parameter estimation is done using Gibbs sampling and Minka's fixed-point iteration. Rating prediction is done using marginal probabilities and the fitness value for optimization is also computed. The recommendation set should have accuracy as well as diversity to know different interests of the user. Trade-off between accuracy and diversity is taken care of by using BPSO. Initially, the algorithm tries to achieve desired level of accuracy. Once desired level of accuracy is achieved, contextual factors are optimized to provide diversity in the recommendation set. Experimental evaluation was done on LDOS-CoMoDa movie dataset. Four variations of the BFMM approach are proposed in [116] namely BFMM-CR, BFMM-CA, BFMM-CU and BFMM-CI. Out of these variations, BFMM-CR is the main proposed model which takes only relevant contexts into consideration. BFMM-CA takes all contexts into account whereas BFMM-CU and BFMM-CI considers only contexts related to only user classes and only item classes respectively. The performance comparison of these models with MF, BFMM and context aware CF indicates that BFMM-CR is most appropriate to handle accuracy and diversity trade-off. Still scalability and accuracy of the model in real time recommendation systems must be evaluated.

Yuan et al. [118] propose a Bayesian approach for Context Aware Recommendation Systems which is extension of their work in [119]. It is a non parametric approach where manual tuning of parameter is not required. Twitter messages give contextual information like who is the user, location of the user, where the user has traveled and activities user has done. These are actually part of behavioral aspects of the user. In the proposed approach, behavioral patterns of the user are learned in the offline mode. Recommendation prediction by incorporating contextual attributes is done in the online mode. The major contribution of this approach lies in offline model as the four behavioral aspects and their relations must be modeled appropriately. In the model proposed in [119], same parameter values are assumed for all users whereas in the current approach, the parameters are learned. For every tweet of a user on a specified day, a personal region and time is drawn. A topic is also drawn depending on user's preferences and a sampled region. The location and each word is drawn based on location and word distribution in a given region. The proposed approach uses Chinese restaurant process which is a stochastic non parametric Bayesian approach. Therefore, it effectively clusters data and takes decision on required number of clusters. To obtain samples of hidden variable assignment, Gibbs sampling is used. This framework has lot of intended applications like requirement aware location recommendations, activity prediction, location prediction, tweet recommendation etc. Experiments were carried out on WW dataset and USA dataset for location prediction using methods like KL model, mean coordinates, popular location, topic+region model, hierarchical graphical model, spatial topic model, who+ where+ when+ what model, enhanced who+ where+ when+ what model and enhanced who+ where+ when+ what model without time factor. Considering accuracy of prediction and average error distance, enhanced who+ where+ when+ what model outperforms all the other models. Use of this framework for adding and modeling other relevant contexts like mood can also be investigated. Table 9 gives comparison of Bayesian theory and learning techniques used in Context Aware Recommendation Systems.

4.4. K- Nearest Neighbor algorithm (kNN)

KNN is most frequently used approach in Collaborative Filtering mechanism employed in recommendation systems. For every user, a group of similar users is formed using similarity measure like cosine similarity. A weighted average of preferences of k most similar users is used to predict the preference of a user for particular item [120]. This section gives review of a literature where kNN has a pivotal role in the design and performance improvement in Context Aware Recommendation Systems.

A contextual pre-filtering approach using item splitting technique is proposed in [26]. A cohort of ratings for an item is split into number of subsets based on value of contextual attributes. The values in the subsets are assigned to new dummy items. The split is performed if ratings in two contextual situations are different. Experimental evaluation was done on Yahoo! webscope and MovieLens 1M dataset. Gender and age of the users are taken as contextual attributes. Use of item splitting is experimented with kNN, MF and non personalized recommendation involving average of item ratings. For kNN, Pearson correlation coefficient was used for finding similarity. Optimal value of k=30 was derived based for kNN. For MovieLens dataset, performance of KNN is reduced by item splitting because target items have small number of ratings and KNN uses all users that have rated target item. Item splitting helped MF to improve accuracy to certain extent. But incorporating other contextual attributes like time, location etc. in real time data may help to improve the performance of the system with item splitting.

Campos et al. [121] propose a time biased recommendation approach using kNN. Time is an important context while giving recommendation as recency of preferences has an impact on user choices. Dataset considered for experimental evaluation include movie ratings in Christmas week 2009 and in Oscar week 2010 to investigate importance of time in recommendation. In the proposed time-biased kNN approach, once a group of similar users is formed, most recent ratings of neighbors are used to predict rating of the target user. The proposed approach is compared with approaches like normal kNN, an ad-hoc and time periodic biased kNN. In normal kNN, the value of k is considered 3. User similarities are computed using Pearson correlation Coefficient. In ad-hoc strategy, preference of user to an item is predicted using movies received by the user and based on similarity of these movies with the target movie. Time periodic biased kNN is an approach where data from immediate last month and previous year data for same month are used in recommendation process. The accuracy of time-biased kNN is better than other selected baseline algorithms. Also inclusion of previous information in time periodic biased kNN has no contribution in predicting user preferences. Still exhaustive comparative study with established techniques like MF can be done to improve the design and accuracy of the system.

Chen et al. [122] propose web service recommendation technique using kNN. Web service discovery and recommendation is critical for users as there are many services available for a specific task. But QoS parameters play a vital role in deciding usefulness of recommended web services. Many of the times, QoS parameters like response time depend on region where the service and users are located. Therefore, a RegionKNN method is proposed which considers region wise QoS parameters. First, region-sensitive services are recognized and then modified kNN is used to predict QoS of web services. Finally, a set of services with best QoS are recommended to the user. The technique has region model building step which is done offline. Region is a cluster of users in nearby locality having similar Round Trip Time profiles. Because a user normally gives less number of QoS values, the data are sparse. Hence, to improve accuracy, region aggregation

Table 9
Context Aware Recommendation Systems using Bayesian theory and learning.

Reference	Main feature	Data source	Application domain	Contexts used	Context type	Future scope
[111]	• Use of fuzzy systems with Bayesian network leading to contextual inference	Data collected by web scraping music website	Music	Time, day, weather, noise	Static fully observable	Inclusion of contexts like mood can be tested
[112]	• Use of various contextual attributes to infer preference using Bayesian network	Data collected from survey	Restaurant	location, season, time, mood, class, price, user profile, weather, temperature	Dynamic partially observable	Implicit ways of capturing mood can be investigated
[113]	• Use of hierarchical Bayesian model for rating prediction	Recommendz dataset	Movie	time, mood, companion etc.	Dynamic partially observable	Investigation of appropriate context inclusion technique
[114]	• Use of Bayesian classifier to incorporate context in recommendation for mobile devices	Data collected by the proposed system	Video	time, location, mood	Dynamic partially observable	mood can be implicitly captured
[115]	• Use of naive Bayesian classification and SVD for dimensionality reduction	Data through simulation	Tourism	continent, companion, age, motive	Static fully observable	Testing in real time scenario
[116]	• Trade-off between accuracy and diversity is maintained	LDOS-CoMoDa dataset	Movie	day type, location, end emotions, dominant emotions, mood, physical state	Dynamic unobservable	Inclusion of mood, emotions implicitly
[118]	• Automatic parameter tunning approach	WW dataset, USA dataset	User mobility	user, Geo-tags, time, activity	Dynamic partially observable	Privacy issues in context capture can be investigated

algorithm is used which is based on bottom-up hierarchical clustering. Neighbor selection is an important step in predicting QoS values as similarity between user and region center found using average of QoS parameters in the region. WSRec dataset is used for experimental evaluation and MAE is used as evaluation metric. RegionKNN shows less value for MAE compared to baselines like user based algorithm using PCC, item based algorithm using PCC etc. RegionKNN performs better in sparse data and is also scalable. Inclusion of other contextual situations like time dependent behavior of QoS along with region can be further investigated.

A group recommendation approach for improving accuracy of recommendations is proposed in [123]. In cyber–physical–social systems, recommendations require analyzing multidimensional big data. Social data can give idea of emotional status of the users. So the proposed system integrates mobile, and social data to provide effective recommendations. In this approach, groups are formed based on similarity of user behavior. Emotional offset is calculated using sentiment analysis of reviews and included in ratings of the users. The behavior of users is modeled using spacetime-user-behavior model. Using Tucker decomposition, dimensionality reduction of user behavior data is performed. The multidimensional data is represented as a tensor. Groups of similar users are formed using kNN. MF techniques are used for generating group preferences. Data for conducting experiment was taken from Yelp dataset. Experimental evaluation shows that in the group discovery procedure, when the value of k is small, recall is low and precision is high. Whereas, when the value of k is large, recall is high and precision is low. However, for k=5, the values of recall, precision and F1 score are high. Therefore, the value of k in kNN is taken as 5. Even RMSE of proposed approach is less than item-CF, user-CF and MF technique. Further attempts can be made to make the system more efficient using bio-inspired learning techniques. Table 10 gives overview of the techniques which use kNN in Context Aware Recommendation Systems.

4.5. Support Vector Machine (SVM)

SVM is a supervised learning method. It is used to classify data by learning about the best line called hyperplane which separates them. There is a kernel version of SVM which can be used to classify data that cannot be separated by a line [124]. Cortes et al. also talk about support vector networks which can be used to solve two group classification problems in [124]. This section gives a brief overview of literature where SVM is used in Context Aware Recommendation Systems.

Oku et al. [125] propose context-aware recommendation system for restaurants using SVM. A Context-aware SVM(C-SVM) is used which considers different contextual attributes in classification process done by SVM. A Context-aware SVM using CF(C-SVM-CF) is also proposed in [125] which computes similarity between users considering their preferences in different contextual situations. Experiments were carried out on Yahoo!Gourmet dataset showing accuracy of C-SVM is better than SVM and satisfaction level of users to recommended restaurants is better with C-SVM-CF compared to C-SVM, SVM and random methods. This work can be further extended by including more contextual attributes.

Kahng et al. propose a ranking based approach for giving recommendations under different contextual conditions in [126]. This work involves usage of five types of features which are used for ranking. The types of features are:

1. Popularity based
2. User based
3. Usage of contextual variables
4. Usage of user and any one contextual variable
5. Usage of all components

Table 10
Context Aware Recommendation Systems using kNN.

Reference	Main feature	Data source	Application domain	Contexts used	Context type	Future scope
[26]	• Experimentation of item-splitting technique with kNN and MF	Yahoo webscope movies, Movielens 1M	Movie	gender, age	Static fully observable	Other contexts like time can be used from Movielens 10M dataset to improve accuracy
[121]	• Use of kNN along with time biasing	Weekly filmtipset dataset	Movie	time	Static fully observable	Contexts like companion, location can also be explored in the system
[122]	• Location based web service recommendation approach	WSRec dataset	Web Service	location	Static fully observable	Time can also be included to further investigate accuracy
[123]	• Group recommendation approach using kNN	Yelp dataset	Business recommendation	time, location	Static fully observable	Use of other MF techniques can be investigated

To retrieve related items based on user query, Latent Dirichlet Allocation based model is used. After the retrieval of items is performed, Ranking SVM is used to rank the items in the result. Experimental evaluation on the data from *Bugs* as music streaming service and location based service called Foursquare show better performance of Ranking SVM as compared to popularity based, User only CF and reduction techniques using normalized discounted cumulative gain as a metric. Incorporation of context inference techniques and detailed experimentation with other baseline algorithms may help in further improving efficiency of the system.

SVM based approach for context-aware recommendation system is suggested in [127]. The main purpose of using SVM is to classify features appropriately. User preferences are categorized as context independent and context dependent. They are given as input to SVM which generates recommendations. Experiment was performed on TripAdvisor dataset and SVM based approach was found performing better. This method can be incorporated in CF approaches. Other forms of SVM may be experimented to further improve accuracy of the system.

A movie recommendation system using SVM and IPSO is proposed in [128]. It uses movie's content information like genre, title etc. along with the user information like gender, occupation, demographics etc. By using this information, it forms user–movie feature vector. Classification of movies is done using SVM and basic recommendation list is obtained. After that, SVM regression model is trained based on this list and ratings of the movies are predicted . Final recommendation list is obtained by using basic recommendation list and predicted ratings. Optimization of SVM parameters is done using IPSO. Experimental evaluation on MovieLens 1M dataset shows better accuracy of classification using IPSO than PSO, GA and GS techniques. IPSO and GS are giving almost similar classification accuracy but IPSO is evaluated to be stable compared to GS. The proposed method gives better prediction accuracy compared to SVM direct regression, user-based CF, item-based CF, BPNN and multiple linear regression. Use of environmental contextual factors in this method may improve accuracy of the system. Table 11 gives comparison of techniques based on SVM in Context Aware Recommendation Systems.

4.6. Hidden Markov Model (HMM)

Markov chain is a kind of random process which is described by states and the probability of being in a state depends only on the previous state. A HMM is a dynamic statistical model with hidden Markov chain and every state produces random observation out of available observations [129]. The HMM described in [129] has states as latent contexts and preferred items by the user as observations. The task of the model is to estimate the transition probabilities between the states and the probability distribution of items at all hidden states. In case of dynamic contextual conditions, where user preferences keep on changing based on changes in the contextual conditions, HMM can be useful to predict user preferences [129]. In situations when data is sparse, HMMs are useful as they make use of latent contexts [129, 130].

A Context Aware Recommendation System using proposed in [130] which Hierarchical HMM. This model has two levels of hidden variables. First level uses positive feedback sequence of users to train the hidden variables that represent latent context of users over time. Common patterns in contextual states are represented by the hidden variables in the second level based on the hidden variables in the first level. Experimental evaluation of this model with baseline techniques used like HMM, User based kNN, most popular selection, random selection etc. shows better accuracy in terms of precision, recall and F-measure.

Li et al. [131] proposed a mixture HMM which uses HMM along with CF for user preference prediction. The experimental evaluation was done on music sequence prediction dataset from Kaggle prediction competition. The dataset has artists preferred by users on online streaming service. The model gives more weight to artist which is preferred more. The accuracy of this model is better compared to only CF and HMM. Table 12 gives comparison of HMM models in Context Aware Recommendation Systems.

4.7. Latent Dirichlet Allocation (LDA)

LDA is a probabilistic model which captures implicit topic structure from a collection of documents [132]. Every document is made up of specific topic distribution and every topic consists of probability distribution of words. Therefore, it is a three-level hierarchical Bayesian model. The hierarchy has a word layer, a topic layer and a document layer. Using these layers and observed documents, it tries to build latent topic structure. In case of Context Aware Recommendation Systems, if contexts are static unobservable or dynamic partially observable, the latent knowledge can be built using LDA. Following attempts are made in this direction. Using the latent knowledge, cold start problem can be handled to a certain extent.

Yuan et al. [133] proposed a context aware LDA model for TV program recommendation using data collected on campus via IPTV provider system. Time of the day and live/VoD option are considered as contextual attributes for the system. In the beginning, a user-oriented LDA is used on user-program matrix document term matrix to find preferences of individual users. These preferences along with contextual attributes are used to build context aware LDA model. Parameter estimation is done using Gibbs sampling technique [133] to avoid local optima.

Table 11
Context Aware Recommendation Systems using Support Vector Machine.

Reference	Main feature	Data source	Application domain	Contexts used	Context type	Future scope
[125]	• Use of Context-aware SVM and Context-aware SVM with CF to improve accuracy and user satisfaction	Yahoo! Gourmet dataset	Restaurant	time, schedule, partner, weather, temperature	Static partially observable	Use of other techniques used for classification like ANN can be investigated.
[126]	• Use of Ranking SVM to rank recommended results	Data collected from Bugs music streaming service and Foursqaure location based service	Music	date, time of the day, weather, location	Static fully observable	contextual inference of attributes like mood can be explored
[127]	• Use of SVM to classify context dependent and context independent feature vectors	Tripadvisor dataset	Tourism	room, cleanliness, value, service, location, check-in, business	Static fully observable	Use of non-linear SVM models can be explored
[128]	• Use of SVM and IPSO to classify movies	Movielens 1M	Movie	user demographics	Static fully observable	Use of time context may help in improving accuracy

Table 12
Context Aware Recommendation Systems using HMM.

Reference	Main feature	Data source	Application domain	Contexts used	Context type	Future scope
[130]	• Use of Hierarchical HMM to predict user preferences in dynamic context situations	Last.fm dataset, Netflix dataset	Music, Movie	time, user activity	Dynamic partially observable	Use of other contexts like location, companion can be tested
[131]	• Use of mixture HMM to predict music sequence	Music sequence prediction dataset from Kaggle	Music	User activity	Dynamic partially observable	Exploring use of other contexts like mood, time of the day

Table 13
Context Aware Recommendation Systems using LDA.

Reference	Main feature	Data source	Application domain	Contexts used	Context type	Future scope
[133]	• Use of context aware LDA to incorporate context for improving prediction accuracy	Data collected by IPTV service provider from students on campus in UK	TV program recommendation	time of the day, live/VoD environment	Dynamic partially observable	Performance comparison with other LDA models can be done to get more insight
[132]	• Use of seasonal layer in LDA to incorporate seasonal influence in tourism related documents	Tourism documents from Wikitravel and TravelChinaGuide	Tourism	Season	Dynamic partially observable	Model to test influence of companion, weather can be experimented

Results show better accuracy of this model over other models like random selection, most popular selection.

A tourist recommendation system using LDA is proposed in [132]. The documents related to tourist attraction usually contain patterns influenced by season. Therefore, [132] introduces an additional season layer with seasonal features and the model is named as STLDA. To learn model parameters, Gibbs sampling technique is used. Experimental evaluation shows better accuracy of STLDA over LDA. Table 13 gives comparison of LDA techniques in Context Aware Recommendation Systems.

4.8. Multi- Armed Bandit (MAB) algorithms

A stochastic MAB is a classical problem for understanding exploration and exploitation trade-off. It is a part of reinforcement learning. This set of problems is faced by automated agent when it tries to explore its environment while exploiting the best knowledge it has learned. This problem is inspired by slot machines in casino. Each arm of a slot machine has specific probability distribution with associated mean and variance values. The gambler using a slot machine has to find an arm with highest expected value as well as obtain as much rewards as possible. In the entire process, total regret of choosing a wrong arm must be as small as possible [134]. In case of recommendation systems, the number of items recommended say k can be treated as k arms of slot machine in MAB problem. The recommendation system has to find relevant items for the user by learning about the user preferences with minimum regret caused using the best knowledge it has gained for the user. It may be effective in learning new avenues for recommendations by still exploiting the best choice for the user. There are many algorithms designed to solve MAB problem including ϵ-greedy, softmax, Upper Confidence Bound(UCB) etc. This section gives an overview of literature where MAB algorithms are used in Context-aware Recommendation Systems.

Bouneffouf et al. [65] propose a contextual bandit algorithm called contextual ϵ-greedy for mobile recommendation systems. The content on mobile devices rapidly evolve and keeps on changing. Many mobile recommendation systems will not consider this aspect while dealing with exploration and exploitation trade-off. So [65] uses contextual ϵ-greedy to tackle this issue. A user model consists of a user's situation and corresponding preferences of the user. A User's preferences are modeled using certain characteristics like total number of click on a document, time spent reading the document etc. Context is represented by an ontology. A set of certain High Level Critical contextual Situations(HLCS) where only exploitation is preferred is identified. In every iteration of the algorithm, system calculates similarity between current situation of the user and previously recorded situations. Corresponding to identified situation, a set of documents is recommended to the user. The system observes user's behavior and gives rewards based on user's actions. Finally it selects the document with highest reward. The system also decides the document selection strategy based on observed rewards. In the proposed strategy, in HLCS, the system only exploits best item instead of performing exploration. In other situations, it may go for exploration depending on ϵ value. Experimental evaluation based on diary study shows that compared to other ϵ-greedy strategies, the proposed strategy has higher average value of CTR on documents. Inclusion of more contextual values be tested for the effect on scalability of the system especially in the mobile environment.

Many solutions to contextual bandit problem has specific parameter for controlled exploration. But in many online systems with large dataset, estimating the value of the parameter is a difficult task. Few strategies involve usage of probability matching techniques based on Bayesian learning models. But improper apriori distribution may compromise balance in exploration and exploitation phases. Therefore, in [135], a non-Bayesian probability matching using online bootstrap is proposed. This strategy is also a contextual bandit strategy. Experimental evaluation on Yahoo! Today news dataset and KDD cup 2012 online advertising dataset with user features as context to recommend items(i.e.to pull the arm of the slot machine under certain context) is performed. Results show slightly better values of average CTR compared to baselines like ϵ-greedy, only exploit, LinUCB, Thompson sampling in cold start as well as without cold start situations. Experimental evaluation also shows that this strategy is easily scalable which is the requirement for online real time recommendation systems. Exploring the impact of other contextual factors like time, location etc. can be the future research direction. Even contextual inferences may be tried out in this setting.

In real time recommendation systems, user preferences evolve over a period of time. Apart from that, they dynamically change over a period of time. To include this factor in recommendations, Zeng et al. [136] propose time varying MAB strategy. They propose time varying UCB and Time Varying Thompson sampling methods. Experimental evaluation on Yahoo! today news and KDD cup 2012 online advertising datasets with user and item features as context is performed. Results show comparable values of average CTR compared to baselines like random arm selection, ϵ-greedy, general UCB, Thompson sampling and bootstrap method proposed in [135]. Exploring correlation between two arms in time varying setting can be further research issue. Incorporation of environmental contextual factors may also be evaluated in this setup.

Commonly used recommendation methods like content based filtering and CF usually learn a model on a static data. In real time recommendation systems, recommendations are given based on dynamic data. Bandit algorithms are suitable for such scenario. These methods usually do not take preferences of similar users into consideration. An attempt is made to include CF strategy in contextual bandit setting in [137]. Co-clustering technique is used to cluster items and users along with contextual bandit algorithm. Experimental evaluation on Yahoo! news, Telefonica and Avazu datasets with different UCB techniques as baseline show comparable results with proposed technique. But all experiments conducted are without taking features into consideration due to inadequate data. Inclusion of features and more contextual information may further improve CTR value which is used as accuracy measure.

Zhang et al. [138] propose MAB algorithm called Con-CNAME (Contexts and Chosen Number of Arms with Minimal Estimation). It incorporates feedback given by the users and prior information of contexts. Unlike many contextual algorithms, Con-CNAME maintains exploration and process which includes searching for new items in the dataset in adaptable fashion. In the absence of contextual information, Con-CNAME behaves like context-free CNAME algorithm proposed in [139]. This algorithm can also be used in asynchronous manner which helps in improving CTR values. Experimental evaluation was done on Yahoo! Front Page Today dataset depicting CTR values on articles and contextual attributes like user's age, gender etc. Con-CNAME gives better CTR values compared to algorithms like random recommendation, most click and Linear Bayes algorithm. It gives comparable performance to algorithms like contextual click, naïve III algorithm etc. Inclusion of environmental contextual factors and different types of feedbacks may be experimented with for further improving accuracy and CTR values.

A combinatorial MAB framework in contextual setting called Contextual Combinatorial Multi Armed Bandit(CCMAB) is proposed in [66]. It takes advantages of both contextual and combinatorial bandits because in each iteration, a set of arms is selected based on contextual information. In certain real time recommendation applications, number of arms getting selected in each iteration may change due to dynamic nature of data. Such situation is called volatile arm selection in bandit setting. In specific situations, the rewards obtained cannot be taken as summation of all the rewards in each iteration but taken as rewards with diminishing returns due to redundancy in selected arms. This property is called submodular rewards. CCMAB is volatile and submodular in nature which makes it suitable candidate for real time dynamic context-aware recommendation systems. In this algorithm, arms are clustered using contextual situations assuming arms in same cluster will get similar rewards in similar situations. Experimental evaluation was done on Yelp dataset having businesses, users and reviews considering contexts like location, number of fans, number of votes and number of years the user has an elite status etc. CCMAB has more cumulative rewards compared to algorithms like K-LinUCB, UCB, CC-MAB-NS and random algorithms. Only Oracle algorithm has better rewards than CCMAB. Even Oracle and CCMAB converge early compared to other algorithms in consideration. Dynamic clustering of arms may be experimented in CCMAB setting. Even the effect of incorporation of other contextual factors on scalability of the system should be tested. Table 14 gives comparison of MAB algorithms used in Context Aware Recommendation Systems.

5. Discussion and future scope

The goal of recommendation systems is to align their recommendations to evolving user preferences. This implies recommendation systems have to learn those preferences. The interaction between the user, system and the environment help in shaping of those preferences forming a set of contextual attributes. As stated in [19], the features of the user, item and environment has their own contribution in forming those contexts. They help in understanding the situation in which the preferences of the users are formed.

Table 14
Context Aware Recommendation Systems using MAB algorithms.

Reference	Main feature	Data source	Application domain	Contexts used	Context type	Future scope
[65]	• Use of contextual ϵ-greedy approach to handle dynamic nature of content evolution	Survey conducted for experimentation	event recommendation	time, location, social	Dynamic partially observable	Use of other MAB algorithms like UCB can be explored
[135]	• Use of non-Bayesian parameter free contextual bandit for recommendation	Yahoo! today news dataset, KDD cup 2012 online advertising data	News, advertising	User habits	Dynamic unobservable	Use of known features can be experimented to infer context and reduce no.of iterations
[136]	• Use of time varying contextual bandit strategy to incorporate dynamic changes in user preferences over time	KDD cup 2012 online advertising data	Advertising	User habits	Dynamic unobservable	Finding correlation between arms in time varying set up can be experimented
[137]	• Use of CF along with Contextual Bandit	Yahoo! today news dataset, Telefonica dataset	News, advertising	timestamp, device type, connection type, user click pattern etc.	Dynamic partially observable	Context correlation can be experimented to reduce no. of iterations
[138]	• Use of minimal estimation technique with contextual information	Yahoo! today news dataset	News	User features and habits	Dynamic unobservable	Context correlation can be experimented to reduce no. of iterations
[66]	• Use of combinatorial and contextual MAB which is volatile and submodular	Yelp dataset	Business recommendation	Dynamic partially observable	Location, no. of fans, no. of votes, no. of years a user is elite	Dynamic clustering and use of real time contexts can be explored

5.1. Addressing the cold start problem

As seen from the survey, the problem of cold start is also important for Context Aware Recommendation Systems as they too encounter new users or new items about which insufficient information is available at the beginning. The key to effective handling of the cold start problem lies in the fact that how quickly the system knows about the new user or new item and adapts itself to the changes in the user preferences [135]. Table 15 gives the list of references covered in the survey that addresses the cold start issue.

By analyzing Table 15, Context Aware Recommendation Systems try to handle cold start by capturing and incorporating as much contextual information as possible in the recommendation model. Even the type of context also plays a role in which the recommendation system handles the cold start problem. As seen from the table, if the context is static and observable, the system knows everything about the context and hence utilize the contextual information fully to know about the new user or item. In Table 15, ANN, GA and ACO are taking the advantage of this type of context to tackle cold start. Whereas, when the context is dynamic and not fully observable, the recommendation system tries to learn about the context from the following two ways.

1. Learn about the other contexts from whatever contexts are known to the system by context inference used by (SVM and Bayesian models in Table 15) or by using latent topical models like LDA.
2. Learn about the preferences of the user by exploring about him/her through continuous feedback during the interaction under different contexts in real time used by MAB technique as seen from Table 15.

5.2. Addressing the data sparsity problem

Apart from cold start, data sparsity can also be an issue with Context Aware Recommendation systems. Context may act as a filter which narrows down the preferences available. Therefore, the data available with the system is sparse [80,87]. Even, there is a possibility that user preferences under different contexts are not available with the system. There are few bio-inspired computational intelligence techniques and few statistical computing techniques that try addressing this issue. Table 16 describes different techniques used in the survey that address the data sparsity issue.

As seen from Table 16, different techniques try to handle data sparsity based on the type of context used in the system. In case of static fully observable contexts, the system tries to use them effectively to predict the unknown user preferences as observed in [52,64,75,89,92,101,110,122]. On the other hand, if the context is dynamic and not fully observable, the system tries to learn about the context using the existing information like context similarity, use of latent knowledge, estimation of contextual rating etc. as seen from the remaining entries of Table 16.

In case, if the inclusion of contextual attributes add more constraints and hence cause data sparsity problem due to unavailability of the rating, then a context relaxation technique is suggested in [114]. Another technique to address this issue is to find the importance or weights of different contexts and use them in the technique used for recommendation to learn about user preferences as proposed in [81,87].

5.3. Addressing the scalability problem

The third issue is scalability of the system. As the number of users, items and contexts increase, more computation power is required. Because of addition of contexts, the system may be working in multi-dimensional space which requires it to be scalable. Table 17 describes the scalability issue addressed by the literature surveyed in the manuscript. If the number of users, items and contexts increase, then the recommendation systems go for parallel programming framework where the algorithm or technique used can be executed in parallel on cluster of machines improving scalability of the system as seen from Table 17. Majority techniques in Table 17 use such framework irrespective of the type of context used.

Table 15
List of references addressing cold start.

Technique used	Reference	Contexts used	Context type	Reason to handle cold start
ANN	[51]	Time	Static fully observable	Use of a collaborative method in which rating records of similar users in similar context are treated as part of the training set.
	[57]	location, time, user mobility history	Dynamic partially observable	Use of case based reasoning with ANN to take user feedback along with context into consideration.
GA	[64]	Location	Static fully observable	Use of hub average inference method to get location wise popularity ranking for new user.
	[69]	Time, location, demographic information	Static fully observable	Use of demographic contexts to find user similarity along with application of GA.
ACO	[75]	Features of the items, time	Static fully observable	Use of ACO with fuzzy c means as heuristic function and training ANN.
Bayesian theory and learning	[113]	time, mood, companion etc.	Dynamic partially observable	Use of hierarchical Bayesian model to relate the preference weights of different users.
SVM	[128]	user demographics	Static fully observable	Use of demographic context information to help SVM model to learn user preferences to build primary preference list which is subsequently modified by the model.
LDA	[133]	time of the day, live/VoD environment	Dynamic partially observable	LDA takes advantage of latent contextual factors to recommend new items.
	[132]	Season	Dynamic partially observable	LDA is a probabilistic model which takes latent seasonal contextual documents into consideration to make recommendation about new items.
MAB	[135]	User habits	Dynamic unobservable	In case of cold start when no data is available about new user or new item, the model tries to explore the user–item space and as data is getting collected during the process, it makes use of the learned preference in contextual setting.
	[136]	User habits	Dynamic unobservable	Initially when the user or item is new, the model tries to explore and get data from the available contexts and when the data is collected it uses it to personalize recommendations.
	[137]	timestamp, device type, connection type, user click pattern etc.	Dynamic partially observable	Makes use of contexts with CF to get a collaborative knowledge of the problem space and use MAB to get the advantage of user interaction to learn the preferences in real time.
	[138]	User features and habits	Dynamic unobservable	It uses user feedback and contexts both to keep the exploration process running with exploitation process helping to address cold start.

Use of more contextual information may lead to increase in number of dimensions of the problem space. Therefore, some systems use dimensionality reduction techniques to achieve accuracy with lesser contextual attributes. [83,122,137] use clustering technique to reduce the dimensionality. Another approach used by [81,98] is contextual weighting where the preferred contexts are having more importance than others and are used in the preference prediction process. As observed from Table 17, systems with static contexts prefer to use clustering for dimensionality reduction. Whereas systems with dynamic contexts generally use context weighting approaches because contextual preferences keep on changing over a period of time in case of dynamic contexts.

5.4. Datasets used in Context Aware Recommendation Systems

In order to build and evaluate the performance of Context Aware Recommendation Systems, datasets with all the relevant contextual attributes play a vital role. Table 18 describes some publicly available datasets used in different domains that are referred in the manuscript. It is evident after analyzing the Table 18 that most of the datasets have static fully observable content as major role of the datasets is in offline evaluation of the recommendation system. The LDOS-CoMoDa dataset has unobservable contexts like emotions, mood etc. after watching movie. This dataset has such attributes collected from the survey where the respondents have answered such dynamic contexts explicitly [140]. A real time testing of such a system may require gathering these contextual information implicitly. The KDD cup 2012 dataset [141] has user behavior information collected by clicks and impressions made by user while selecting advertisements. So this dataset has real time dynamic unobservable data. Usually, a synthetic data is used to test the efficiency of the model before the system goes live. A tool to generate synthetic data and evaluate performance of the model is given in [142].

5.5. Performance evaluation in Context Aware Recommendation Systems

The performance of the Context Aware Recommendation System is measured in terms of error in predicting user preferences. These measures are offline performance measures like MAE, F1 score etc. There are online evaluation metric like CTR which are helpful in measuring real time performance of the system.

Table 16
List of references addressing data sparsity.

Technique used	Reference	Contexts used	Context type	Reason to handle data sparsity
ANN	[52]	Application context of movie and notebook	Static fully observable	Creation of utility function for the users and users give importance of each attribute and context.
	[55]	Location, time, weather, social media sentiments	Dynamic partially observable	Model tries to learn user preferences by contexts like social media and also tries to give weights to important contexts which can be used to predict user preference using ANN.
	[58]	user music sequences	Dynamic partially observable	Use of ANN embedding to which can gather item relationships and contexts.
GA	[64]	Location	Static fully observable	Use of location context along with GA to optimize weights to deal with sparsity.
ACO	[75]	Features of the items, time	Static fully observable	Model tries to make use of time as a context to learn about user preferences.
PSO	[81]	Degree of hunger, food type etc. for food data & time, companion, location for movie	Dynamic partially observable	Use of differential context weighting along with PSO
	[87]	day type, location, emotions etc. for movie & driving style, landscape, mood etc. for music	Dynamic partially observable	PSO is used to compute the contribution of different contexts.
AIS	[89]	Item and user features	Static fully observable	Calculation of affinity between users using AIS and available contextual information
ABC	[92]	Features of the movie	Static fully observable	Use of iterative procedure using ABC to group similar users and predict their preferences.
MF	[98]	Time, Social behavior	Dynamic partially observable	Combined knowledge of ratings and sparse social behavior in MF.
	[99]	Movie mood	Dynamic unobservable	Use of contexts like movie mood and movie similarity along with joint MF.
	[101]	movie tag, release time	Static fully observable	Use number of iterations of MF with available contexts.
	[102]	movie mood, plot keywords	Dynamic partially observable	Use of contextual information with joint MF to handle sparse rating matrix.
	[103]	driving style, mood, roadtype, traffic etc. for music; hunger, temperature, fatigue for food; day of the week, location for restaurant	Dynamic partially observable for music, dynamic partially observable for food and static fully observable for restaurant	Use of contextual rating in training the model & non contextual rating to estimate ranking score.
	[104]	words in reviews, movie plots etc.	Dynamic partially observable	Balances importance of ratings and document description using available contextual information
TF	[110]	check-in time, location, point of interest	Static fully observable	A TF model simultaneously factorizing tensor with many contexts and features using correlation between them.
Bayesian theory and learning	[114]	time, location, mood	Dynamic partially observable	Context relaxation using Bayes theorem.
	[116]	day type, location, emotions etc.	Dynamic unobservable	Available contexts are related to users and items.
KNN	[122]	location	Static fully observable	Use of region aggregation and similarity using location.
	[123]	time, location	Static fully observable	Use of Tucker decomposition to get dense approximation of original tensor.
HMM	[130]	time, user activity	Dynamic partially observable	Use of hierarchical HMM to adapt to changes in user preferences by context changes using the latent contexts.

There are certain other measures like coverage, diversity, user satisfaction etc. discussed earlier in Section 2.

Table 19 describes different evaluation measures used throughout in the literature referred in the manuscript. Majority of

Table 17
List of references addressing scalability.

Technique used	Reference	Contexts used	Context type	Reason to handle scalability
ANN	[56]	Location, time, reviews	Static partially observable	Only model parameters need to be stored in memory.
GA	[62]	Youtube metadata information	Static fully observable	Use of parallel pipelines and proportional model using contexts.
	[64]	Location	Static fully observable	Use of decentralized cloud based solution
ACO	[71]	Time, Location, Activity type	Static partially observable	Can be operated in parallel computing framework.
	[72]	Time, Location, Activity type	Static partially observable	Can be operated in parallel computing framework.
	[73]	Time, music type, artist	Static fully observable	Can be operated in parallel computing framework.
PSO	[80]	Degree of hunger, gender, demographics, food type	Dynamic partially observable	Optimum context relaxation set is identified using PSO avoiding exhaustive search.
	[81]	Degree of hunger, gender, demographics, food type for food data & time, companion, location for movie	Dynamic partially observable	Contexts are weighted and selected and weights are selected using PSO.
	[83]	Genre of the movie	Static fully observable	Use of fuzzy c-means and PSO to cluster data reduces high dimension data and improves scalability.
MF	[98]	Time, Social behavior	Dynamic partially observable	Use of time and social network contexts with weights can reduce the data to be stored and MF can be executed in parallel computing framework.
	[99]	Movie mood	Dynamic unobservable	Use of joint MF which can be executed in parallel computing framework.
	[102]	movie mood, plot keywords	Dynamic partially observable	Model can be executed in parallel computing framework.
KNN	[122]	location	Static fully observable	Users are clustered into regions improving scalability.
MAB	[135]	User habits	Dynamic unobservable	Use of samples of different sizes with bootstrap improving scalability as seen from experiments.
	[137]	timestamp, device type, connection type, user click pattern etc.	Dynamic partially observable	Use of dynamic clustering to reduce data size and can be performed in parallel over large data.

the evaluation measures used are offline. MAB technique uses online evaluation as it deals with real time recommendations. User satisfaction is used as evaluation measure in [111,112,125]. Only [67] uses coverage as a measure whereas [133] uses novelty and diversity as a measure. To judge the performance of the system from various perspectives, different measures should be used. Therefore, development of performance measures and using them along with existing offline and online measures can also be a future research direction.

5.6. Discussion on bio-inspired computing techniques

As seen from the survey ANN was used in variety of domains. It incorporates static contexts very well to deal with cold start and data sparsity problems to certain extent as observed from the survey. Genetic algorithm usually uses context inference to deal with cold start and learn about the user preferences. It also tries to optimize context weights and understand user preferences in case of data sparsity [64]. ACO and PSO are usually used for finding optimal recommendation sequence or path as shown in [71,73,80]. They usually deal with scalability using context weighting approach as indicated in [71,73,80]. AIS, ABC, BA and AAA are recently used developed and used techniques in the research of Context Aware Recommendation Systems. They try to optimize the recommendation list accurately and quickly as seen in [90,92,96]. More experimentation in their capacity to handle the cold start, sparsity and scalability challenges can be a future research direction. Even there are systems proposed using more than one bio-inspired algorithms as observed in [68]. Using a hybrid model which takes benefit of characteristics of more than one such method can also be investigated.

5.7. Discussion on statistical computing techniques

As seen from the survey, MF methods work in two dimensional problem space. So, multi-dimensional problem space due to contexts is converted into two dimensional space. MF tries to deal with sparsity by usually by using context similarity as seen from [102]. MF techniques are also scalable [156]. TF uses contextual information in multi-dimensional space itself. Therefore, they do not require conversion to two dimensional space. Even TF techniques are scalable as indicated in [157]. Bayesian methods use context weighting and relaxation to deal with cold start and sparsity issue [122]. KNN uses clustering for dimensionality reduction and hence can be scaled well [122]. SVM is based on classification and uses contextual information well do deal with cold start [128]. HMM and LDA use latent knowledge to deal with cold start and data sparsity [130,133]. Therefore,

Table 18
Publicly available datasets in Context Aware Recommendation Systems referred in this manuscript.

Application domain	Dataset name	Source	Contexts used	Type of context	Reference
Movie	Movielens	[143,144]	features of the users, movies, time	Static fully observable	[59], [75], [83], [89], [92], [96], [101], [104], [109], [26], [128]
	CAMRa challenge dataset 2011	[145]	Time	Static fully observable	[53]
	LDOS-CoMoDa dataset	[140]	Time, day type, season, location, weather, social, end emotion, dominant emotion, mood, physical, decision, interaction	Dynamic unobservable	[67], [87], [116]
	Netflix dataset	[146]	Time	Static fully observable	[130]
Music	InCarMusic dataset	[147]	Driving style, landscape, mood, natural phenomena, road type, sleepiness, traffic conditions, weather	Dynamic partially obsrvable	[87],[103]
	Last.fm dataset	[148],[149]	Time, features of music	Static fully observable	[86], [130]
Venue recommendation	Gowalla dataset	[150]	location, check-in time	Static fully observable	[64]
	Foursquare dataset	[151–153]	check-in location, social	Static partially observable	[108], [126]
Restaurant	Chicago restaurant dataset	[154]	Location, type of cuisine, whether expensive	Static fully observable	[68]
Business recommendation	Yelp academic dataset	[155]	Location, time, reviews	Static partially observable	[56], [123], [66]
Advertising	KDD cup 2012 dataset	[141]	User behavior	Dynamic unobservable	[135], [136]

HMM and LDA are useful in dealing with partially observable and unobservable contexts. MAB strategy is a promising research area in real time online recommendation systems. Most of the study on these strategies is either done on simulation or on a dataset with lesser contextual attributes. So, involving more relevant contextual attributes and real time testing of them can also be experimented with. Use of bio-inspired algorithms in such online systems and their comparison with bandit algorithms can also be explored. Also a model taking advantages of one or more statistical learning techniques to solve multiple issues in Context Aware Recommendation Systems must be explored.

5.8. Privacy issue

Another important issue faced by Context Aware Recommendation Systems and which is rarely addressed in the literature is privacy. Collection of contextual attributes and their usage, collection of implicit feedback requires proper privacy considerations [158]. Incorporation of this constraint in the Context Aware Recommendation Systems without affecting accuracy of the system is also an area of research to be considered. As these privacy constraints may lead to data sparsity problem, use of latent topical models along with explicit feedback and existing models can be area of further research [158].

5.9. Future of model development in Context Aware Recommendation System

It may not be possible to have a single technique addressing all the issue at the same time. Despite few attempts made to use a hybrid model, development of a recommendation framework addressing all the issues is still an open area of research. As seen from the survey, bio-inspired algorithms are comparatively a new field of research. There are very recent bio-inspired algorithms like AAA Systems, bat Algorithm etc. which can be explored more in the area of Context Aware Recommendation systems. Even proper exploration of different parameters and use of relevant contextual situations using them in algorithms like genetic algorithms, ACO etc. can be carried out. There are few time-tested statistical computing techniques like MF, kNN etc. which are extensively used in recommendation systems literature. Development of a model considering advantages of both the approaches can be a good research direction in Context Aware Recommendation Systems.

As presented in the survey, there are techniques to include context in the recommendation system model. Effective selection of the context inclusion strategy suitable for the technique used is also a key factor in deciding accuracy of the system. Apart from that, the way context information is represented and stored may also influence the performance of the system. Selection of the contextual parameters which are actually impacting user preferences and filtering out irrelevant attributes is also a challenge. In case of insufficient contextual information, learning and predicting such contextual attributes based on available knowledge is also a challenge. Development of such context inferencing approaches can also be a probable research direction where the system will infer contexts and learn them over a period of time. In short, context pre-processing is one of the key steps in the development of the models for Context Aware Recommendation Systems.

Feedback of the users about recommended items is an important factor for recommendation systems to learn user preferences. Feedback can be explicit in the form of ratings, reviews etc. Feedback can also be implicit by collecting user history in the form of logs collected or by monitoring CTR, number of visits to view the item etc. Explicit feedback is important for the recommendation systems to learn but many times users are reluctant to give

Table 19
Analysis of references by the performance evaluation measures.

Technique used	Reference	Evaluation type	Evaluation method
ANN	[51]	Offline	Average quality score
	[52]	Other	User satisfaction, trust, usefulness
	[53]	Offline	MAP, AUC
	[54]	Online	Accuracy based on number of interactions
	[56]	Other	Confusion matrix evaluation
	[57]	Other	User satisfaction
	[58]	Offline	Precision, recall, f-measure, hit rate
	[59]	Offline	Precision, recall, f-measure, NDCG
GA	[61]	Other	Responsiveness
	[62]	Other	User interests matched/ iteration
	[63]	Offline	MAE
	[64]	Offline	Precision, recall, f-measure
	[67]	Offline	MAE
		Other	Coverage
	[68]	Offline	Precision
	[69]	Offline	MAE
ACO	[71]	Other	Responsiveness
	[73]	Other	Average saturation of the solutions
	[74]	Offline	Average precision rate
	[75]	Offline	MAE
	[76]	Other	Number of iterations to get optimal solution
PSO	[80]	Offline	RMSE
	[81]	Offline	RMSE
	[83]	Offline	MAE
	[84]	Offline	Precision, recall, f-measure
	[85]	Offline	RMSE, MAE, NDCG
	[86]	Offline	Recall
	[87]	Offline	MAE, RMSE, precision, recall, f-measure
AIS	[89]	Offline	MAE, precision, recall
	[90]	Offline	Precision, recall, f-measure
ABC	[92]	Offline	MAE
BA	[94]	Offline	MAE, precision, recall, f-measure
AAA	[96]	Offline	MAE, precision, recall
MF	[98]	Offline	MAP, AUC, precision
	[99]	Offline	MAP, AUC, precision
	[100]	Offline	MAE
	[101]	Offline	MAE, RMSE
	[102]	Offline	Precision, MAP
	[103]	Offline	Precision, MAP
	[104]	Offline	RMSE
TF	[108]	Offline	Precision, recall
	[109]	Offline	MAE, RMSE. precision
	[110]	Offline	MAE, RMSE
	[107]	Offline	Precision, MAP
Bayesian theory and learning	[111]	Other	User satisfaction
	[112]	Other	User satisfaction
	[113]	Offline	MAE, f-measure
	[115]	Offline	Precision, recall, f-measure
	[116]	Offline	RMSE
	[118]	Offline	Hit rate
KNN	[26]	Offline	MAE
	[121]	Offline	MAP, precision, AUC, NDCG
	[122]	Offline	MAE
	[123]	Offline	Precision, recall, f-measure, RMSE
SVM	[125]	Other	User satisfaction
	[126]	Offline	NDCG
	[127]	Offline	Hit rate
	[128]	Offline	MAE
HMM	[130]	Offline	Precision, recall, f-measure
	[131]	Offline	MAE

(*continued on next page*)

such feedback. In such situations, collecting implicit feedback and properly incorporating it in the Context Aware Recommendation Systems is also important. Many recommendation systems surveyed so far rely heavily on the explicit feedback. There can also be bias introduced in explicit feedback. Development of models for Context Aware Recommendation Systems with balanced explicit and implicit feedback is also an avenue open for the research to improve the accuracy of the system.

Finally, development of interests, choices of the users is also a part of consumer psychology and behavior sciences. Developing a context aware recommendation model considering theories from such fields can also benefit in improving accuracy of the system.

Table 19 (*continued*).

Technique used	Reference	Evaluation type	Evaluation method
LDA	[133]	Offline Other	NDCG, recall Diversity, novelty
	[132]	Offline	Perplexity
MAB	[65]	Offline Online	Precision CTR
	[135]	Online	CTR
	[136]	Online	CTR
	[137]	Online	CTR
	[138]	Online	CTR
	[66]	Online	Cumulative rewards(e.g. CTR)

6. Conclusion

In this survey, we presented the state of the art techniques used in the area of Context Aware Recommendation Systems. We classified them into bio-inspired computing techniques and statistical computing techniques. The survey also focused on the ability of these systems to deal with commonly encountered challenges like cold start, data sparsity and scalability. It also described meaning of the term context, known classifications of context and techniques to incorporate context in recommendation systems. Finally it highlighted the challenges faced by the Context Aware Recommendation Systems and it tries to give insight into some of the future research directions in Context Aware Recommendation Systems. As presented in the survey, there is a need to develop inclusive models using more than one technique along with proper context utilization and appropriate feedback mechanisms with no compromise with privacy to handle all the problems faced by these systems.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] R. East, K. Hammond, W. Lomax, H. Robinson, What is the effect of a recommendation? Mark. Rev. 5 (2) (2005) 145–157.
[2] A. Costa, F. Roda, Recommender systems by means of information retrieval, in: Proceedings of the International Conference on Web Intelligence, Mining and Semantics, ACM, 2011, p. 57.
[3] F. Ricci, L. Rokach, B. Shapira, Recommender systems: Introduction and challenges, in: Recommender Systems Handbook, Springer, 2015, pp. 1–34.
[4] G. Adomavicius, A. Tuzhilin, 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) (2005) 734–749.
[5] J. Leskovec, A. Rajaraman, J.D. Ullman, Mining of Massive Datasets, Cambridge university press, 2014, pp. 307–326.
[6] M.J. Pazzani, D. Billsus, Content-based recommendation systems, in: The Adaptive Web, Springer, 2007, pp. 325–341.
[7] J.S. Breese, D. Heckerman, C. Kadie, Empirical analysis of predictive algorithms for collaborative filtering, in: Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence, Morgan Kaufmann Publishers Inc., 1998, pp. 43–52.
[8] X. Su, T.M. Khoshgoftaar, A survey of collaborative filtering techniques, Adv. Artif. Intell. 2009 (2009) 1–19, article 421425.
[9] R. Thomadsen, R.P. Rooderkerk, O. Amir, N. Arora, B. Bollinger, K. Hansen, L. John, W. Liu, A. Sela, V. Singh, et al., How context affects choice, Cust. Needs Solut. 5 (1–2) (2018) 3–14.
[10] G. Adomavicius, A. Tuzhilin, Context-aware recommender systems, in: Recommender Systems Handbook, Springer, 2011, pp. 217–253.
[11] J.-y. Hong, E.-h. Suh, S.-J. Kim, Context-aware systems: A literature review and classification, Expert Syst. Appl. 36 (4) (2009) 8509–8522.
[12] K. Haruna, M. Akmar Ismail, S. Suhendroyono, D. Damiasih, A.C. Pierewan, H. Chiroma, T. Herawan, Context-aware recommender system: A review of recent developmental process and future research direction, Appl. Sci. 7 (12) (2017) 1211.
[13] N.M. Villegas, C. Sánchez, J. Díaz-Cely, G. Tamura, Characterizing context-aware recommender systems: A systematic literature review, Knowl.-Based Syst. 140 (2018) 173–200.
[14] S. Seyednezhad, K.N. Cozart, J.A. Bowllan, A.O. Smith, A review on recommendation systems: Context-aware to social-based, 2018, arXiv preprint arXiv:1811.11866.
[15] G. Adomavicius, B. Mobasher, F. Ricci, A. Tuzhilin, Context-aware recommender systems, AI Mag. 32 (3) (2011) 67–80.
[16] A.K. Kar, Bio inspired computing–A review of algorithms and scope of applications, Expert Syst. Appl. 59 (2016) 20–32.
[17] G. James, D. Witten, T. Hastie, R. Tibshirani, An Introduction to Statistical Learning, 112, Springer, 2013.
[18] J. Friedman, T. Hastie, R. Tibshirani, The Elements of Statistical Learning, 1, Springer series in statistics New York, NY, USA:, 2001.
[19] M. Bazire, P. Brézillon, Understanding context before using it, in: International and Interdisciplinary Conference on Modeling and using Context, Springer, 2005, pp. 29–40.
[20] M. Van Setten, S. Pokraev, J. Koolwaaij, Context-aware recommendations in the mobile tourist application COMPASS, in: International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems, Springer, 2004, pp. 235–244.
[21] A. Majid, L. Chen, G. Chen, H.T. Mirza, I. Hussain, J. Woodward, A context-aware personalized travel recommendation system based on geotagged social media data mining, Int. J. Geogr. Inf. Sci. 27 (4) (2013) 662–684.
[22] C. Mettouris, G.A. Papadopoulos, Ubiquitous recommender systems, Computing 96 (3) (2014) 223–257.
[23] M. Amoretti, L. Belli, F. Zanichelli, UTravel: Smart mobility with a novel user profiling and recommendation approach, Pervasive Mob. Comput. 38 (2017) 474–489.
[24] A.V. Smirnov, A.M. Kashevnik, A. Ponomarev, Context-based infomobility system for cultural heritage recommendation: Tourist Assistant—TAIS, Pers. Ubiquitous Comput. 21 (2) (2017) 297–311.
[25] Z.V. Ferdousi, D. Colazzo, E. Negre, Correlation-based pre-filtering for context-aware recommendation, in: 2018 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops), IEEE, 2018, pp. 89–94.
[26] L. Baltrunas, F. Ricci, Context-based splitting of item ratings in collaborative filtering, in: Proceedings of the Third ACM Conference on Recommender Systems, ACM, 2009, pp. 245–248.
[27] L. Baltrunas, F. Ricci, Experimental evaluation of context-dependent collaborative filtering using item splitting, User Model. User-Adapt. Interact. 24 (1–2) (2014) 7–34.
[28] M. Singh, H. Sahu, N. Sharma, A personalized context-aware recommender system based on user-item preferences, in: Data Management, Analytics and Innovation, Springer, 2019, pp. 357–374.
[29] M.J. Barranco, J.M. Noguera, J. Castro, L. Martínez, A context-aware mobile recommender system based on location and trajectory, in: Management Intelligent Systems, Springer, 2012, pp. 153–162.
[30] V. Codina, F. Ricci, L. Ceccaroni, Semantically-enhanced pre-filtering for context-aware recommender systems, in: Proceedings of the 3rd Workshop on Context-Awareness in Retrieval and Recommendation, ACM, 2013, pp. 15–18.
[31] V. Codina, F. Ricci, L. Ceccaroni, Exploiting the semantic similarity of contextual situations for pre-filtering recommendation, in: International Conference on User Modeling, Adaptation, and Personalization, Springer, 2013, pp. 165–177.
[32] V. Codina, F. Ricci, L. Ceccaroni, Distributional semantic pre-filtering in context-aware recommender systems, User Model. User-Adapt. Interact. 26 (1) (2016) 1–32.
[33] N. Hariri, B. Mobasher, R. Burke, Context-aware music recommendation based on latenttopic sequential patterns, in: Proceedings of the Sixth ACM Conference on Recommender Systems, ACM, 2012, pp. 131–138.
[34] X. Ramirez-Garcia, M. García-Valdez, Post-filtering for a restaurant context-aware recommender system, in: Recent Advances on Hybrid Approaches for Designing Intelligent Systems, Springer, 2014, pp. 695–707.

[35] P.G. Campos, I. Fernández-Tobías, I. Cantador, F. Díez, Context-aware movie recommendations: An empirical comparison of pre-filtering, post-filtering and contextual modeling approaches, in: International Conference on Electronic Commerce and Web Technologies, Springer, 2013, pp. 137–149.
[36] K. Haruna, M.A. Ismail, D. Damiasih, H. Chiroma, T. Herawan, Comprehensive survey on comparisons across contextual pre-filtering, contextual post-filtering and contextual modelling approaches, Telkomnika 15 (4) (2017) 1865–1875.
[37] J. Bobadilla, F. Ortega, A. Hernando, A. Gutiérrez, Recommender systems survey, Knowl.-Based Syst. 46 (2013) 109–132.
[38] B. Lika, K. Kolomvatsos, S. Hadjiefthymiades, Facing the cold start problem in recommender systems, Expert Syst. Appl. 41 (4) (2014) 2065–2073.
[39] Z. Huang, H. Chen, D. Zeng, Applying associative retrieval techniques to alleviate the sparsity problem in collaborative filtering, ACM Trans. Inf. Syst. 22 (1) (2004) 116–142.
[40] H. Wang, Z. Wang, W. Zhang, Quantitative analysis of matthew effect and sparsity problem of recommender systems, in: 2018 IEEE 3rd International Conference on Cloud Computing and Big Data Analysis (ICCCBDA), IEEE, 2018, pp. 78–82.
[41] F. Isinkaye, Y. Folajimi, B. Ojokoh, Recommendation systems: Principles, methods and evaluation, Egypt. Inform. J. 16 (3) (2015) 261–273.
[42] G. Shani, A. Gunawardana, Evaluating recommendation systems, in: Recommender Systems Handbook, Springer, 2011, pp. 257–297.
[43] Z.D. Champiri, A. Asemi, S.S.S. Binti, Meta-analysis of evaluation methods and metrics used in context-aware scholarly recommender systems, Knowl. Inf. Syst. 61 (2) (2019) 1147–1178.
[44] M. Deshpande, G. Karypis, Item-based top-n recommendation algorithms, ACM Trans. Inf. Syst. 22 (1) (2004) 143–177.
[45] H.M. Wallach, I. Murray, R. Salakhutdinov, D. Mimno, Evaluation methods for topic models, in: Proceedings of the 26th Annual International Conference on Machine Learning, 2009, pp. 1105–1112.
[46] C.-H. Tsai, H.-W. Liu, T. Ku, W.-F. Chien, Personal recommendation engine of user behavior pattern and analysis on social networks, in: 2015 International Conference on Computational Science and Computational Intelligence (CSCI), IEEE, 2015, pp. 404–409.
[47] J. Bhavithra, A. Saradha, Personalized web page recommendation using case-based clustering and weighted association rule mining, Cluster Comput. 22 (3) (2019) 6991–7002.
[48] C. Helmers, P. Krishnan, M. Patnam, Attention and saliency on the internet: Evidence from an online recommendation system, J. Econ. Behav. Organ. 161 (2019) 216–242.
[49] A. Darwish, Bio-inspired computing: Algorithms review, deep analysis, and the scope of applications, Fut. Comput. Inf. J. 3 (2) (2018) 231–246.
[50] S.J. Russell, P. Norvig, Artificial Intelligence - A Modern Approach, second ed., in: Prentice Hall Series in Artificial Intelligence, Prentice Hall, 2003.
[51] N.-H. Liu, S.-J. Hsieh, C.-F. Tsai, An intelligent music playlist generator based on the time parameter with artificial neural networks, Expert Syst. Appl. 37 (4) (2010) 2815–2825.
[52] S.-L. Huang, Designing utility-based recommender systems for e-commerce: Evaluation of preference-elicitation methods, Electron. Commer. Res. Appl. 10 (4) (2011) 398–407.
[53] C. Biancalana, F. Gasparetti, A. Micarelli, A. Miola, G. Sansonetti, Context-aware movie recommendation based on signal processing and machine learning, in: Proceedings of the 2nd Challenge on Context-Aware Movie Recommendation, ACM, 2011, pp. 5–10.
[54] M. Krstic, M. Bjelica, Context-aware personalized program guide based on neural network, IEEE Trans. Consum. Electron. 58 (4) (2012) 1301–1306.
[55] K. Meehan, T. Lunney, K. Curran, A. McCaughey, Context-aware intelligent recommendation system for tourism, in: Pervasive Computing and Communications Workshops (PERCOM Workshops), 2013 IEEE International Conference on, IEEE, 2013, pp. 328–331.
[56] T.K. Paradarami, N.D. Bastian, J.L. Wightman, A hybrid recommender system using artificial neural networks, Expert Syst. Appl. 83 (2017) 300–313.
[57] Z. Bahramian, R. Ali Abbaspour, C. Claramunt, A cold start context-aware recommender system for tour planning using artificial neural network and case based reasoning, Mob. Inf. Syst. 2017 (1) (2017).
[58] D. Wang, S. Deng, G. Xu, Sequence-based context-aware music recommendation, Inf. Retr. J. 21 (2–3) (2018) 230–252.
[59] L. Rakkappan, V. Rajan, Context-aware sequential recommendations withstacked recurrent neural networks, in: The World Wide Web Conference, 2019, pp. 3172–3178.
[60] B. Coppin, Artificial intelligence illuminated, Jones & Bartlett Learning, 2004.
[61] A. Ciaramella, M.G. Cimino, B. Lazzerini, F. Marcelloni, Using context history to personalize a resource recommender via a genetic algorithm, in: Intelligent Systems Design and Applications (ISDA), 2010 10th International Conference on, IEEE, 2010, pp. 965–970.
[62] K.S. Gopalan, S. Nathan, B.T. CH, A.B. Channa, P. Saraf, A context aware personalized media recommendation system: An adaptive evolutionary algorithm approach, in: Bio-Inspired Computing: Theories and Applications (BIC-TA), 2011 Sixth International Conference on, IEEE, 2011, pp. 45–50.
[63] T.H. Dao, S.R. Jeong, H. Ahn, A novel recommendation model of location-based advertising: Context-aware collaborative filtering using GA approach, Expert Syst. Appl. 39 (3) (2012) 3731–3739.
[64] R. Irfan, O. Khalid, M.U.S. Khan, C. Chira, R. Ranjan, F. Zhang, S.U. Khan, B. Veeravalli, K. Li, A.Y. Zomaya, MobiContext: A context-aware cloud-based venue recommendation framework, IEEE Trans. Cloud Comput. 5 (4) (2017) 712–724.
[65] D. Bouneffouf, A. Bouzeghoub, A.L. Gançarski, A contextual-bandit algorithm for mobile context-aware recommender system, in: International Conference on Neural Information Processing, Springer, 2012, pp. 324–331.
[66] L. Chen, J. Xu, Z. Lu, Contextual combinatorial multi-armed bandits with volatile arms and submodular reward, in: Advances in Neural Information Processing Systems, 2018, pp. 3250–3259.
[67] M. Wasid, R. Ali, Context similarity measurement based on genetic algorithm for improved recommendations, in: Applications of Soft Computing for the Web, Springer, 2017, pp. 11–29.
[68] A. Kumar, N. Sachdeva, A. Garg, Analysis of GA optimized ANN for proactive context aware recommender system, in: International Conference on Health Information Science, Springer, 2017, pp. 92–102.
[69] S. Linda, K. Bharadwaj, A genetic algorithm approach to context-aware recommendations based on spatio-temporal aspects, in: Integrated Intelligent Computing, Communication and Security, Springer, 2019, pp. 59–70.
[70] M. Dorigo, G. Di Caro, Ant colony optimization: A new meta-heuristic, in: Evolutionary Computation, 1999. CEC 99. Proceedings of the 1999 Congress on, Vol. 2, IEEE, 1999, pp. 1470–1477.
[71] J.A. Mocholi, J. Jaen, K. Krynicki, A. Catala, A. Picón, A. Cadenas, Learning semantically-annotated routes for context-aware recommendations on map navigation systems, Appl. Soft Comput. 12 (9) (2012) 3088–3098.
[72] J.A. Mocholi, J. Jaen, K. Krynicki, A. Catala, TSACO: Extending a context-aware recommendation system with Allen temporal operators, in: International Conference on Ubiquitous Computing and Ambient Intelligence, Springer, 2012, pp. 253–260.
[73] J.A. Mocholi, V. Martinez, J. Jaen, A. Catala, A multicriteria ant colony algorithm for generating music playlists, Expert Syst. Appl. 39 (3) (2012) 2270–2278.
[74] V. Viswanathan, I. Krishnamurthi, Finding relevant semantic association paths using semantic ant colony optimization algorithm, Soft Comput. 19 (1) (2015) 251–260.
[75] H. Hasija, An effective approach of feature selection for recommender systems using fuzzy C means clustering along with ant colony optimization and neural networks, in: 2017 8th International Conference on Computing, Communication and Networking Technologies (ICCCNT), IEEE, 2017, pp. 1–7.
[76] M. Rastegarmoghadam, K. Ziarati, Improved modeling of intelligent tutoring systems using ant colony optimization, Educ. Inf. Technol. 22 (3) (2017) 1067–1087.
[77] R. Li, Adaptive learning model based on ant colony algorithm, Int. J. Emerg. Technol. Learn 14 (01) (2019) 49–57.
[78] H. Parvin, P. Moradi, S. Esmaeili, TCFACO: Trust-aware collaborative filtering method based on ant colony optimization, Expert Syst. Appl. 118 (2019) 152–168.
[79] J. Kennedy, Particle swarm optimization, in: Encyclopedia of Machine Learning, Springer, 2011, pp. 760–766.
[80] Y. Zheng, R. Burke, B. Mobasher, Optimal feature selection for context-aware recommendation using differential relaxation, in: In ACM RecSys' 12, Proceedings of the 4th International Workshop on Context-Aware Recommender Systems (CARS 2012), ACM, 2012.
[81] Y. Zheng, R. Burke, B. Mobasher, Recommendation with differential context weighting, in: International Conference on User Modeling, Adaptation, and Personalization, Springer, 2013, pp. 152–164.
[82] S. Ujjin, P.J. Bentley, Particle swarm optimization recommender system, in: Swarm Intelligence Symposium, 2003. SIS'03. Proceedings of the 2003 IEEE, IEEE, 2003, pp. 124–131.
[83] R. Katarya, O.P. Verma, A collaborative recommender system enhanced with particle swarm optimization technique, Multimedia Tools Appl. 75 (15) (2016) 9225–9239.
[84] P. Choudhary, V. Kant, P. Dwivedi, A particle swarm optimization approach to multi criteria recommender system utilizing effective similarity measures, in: Proceedings of the 9th International Conference on Machine Learning and Computing, ACM, 2017, pp. 81–85.
[85] M. Hamada, M. Hassan, Artificial neural networks and particle swarm optimization algorithms for preference prediction in multi-criteria recommender systems, in: Informatics, Vol. 5, Multidisciplinary Digital Publishing Institute, 2018, p. 25.

[86] Rahul Katarya, Efficient music recommender system using context graph and particle swarm, Multimedia Tools Appl. 77 (2) (2018) 2673–2687.
[87] P. Jain, V.S. Dixit, Recommendations with context aware framework using particle swarm optimization and unsupervised learning, J. Intell. Fuzzy Systems 36 (5) (2019) 4479–4490.
[88] D. DasGupta, An overview of artificial immune systems and their applications, in: Artificial Immune Systems and their Applications, Springer, 1993, pp. 3–21.
[89] M.-H. Chen, C.-H. Teng, P.-C. Chang, Applying artificial immune systems to collaborative filtering for movie recommendation, Adv. Eng. Inform. 29 (4) (2015) 830–839.
[90] R. Colomo-Palacios, F.J. García-Peñalvo, V. Stantchev, S. Misra, Towards a social and context-aware mobile recommendation system for tourism, Pervasive Mob. Comput. 38 (2017) 505–515.
[91] D. Karaboga, B. Basturk, On the performance of artificial bee colony (ABC) algorithm, Appl. Soft Comput. 8 (1) (2008) 687–697.
[92] R. Katarya, Movie recommender system with metaheuristic artificial bee, Neural Comput. Appl. (2018) 1–8.
[93] X.-S. Yang, A new metaheuristic bat-inspired algorithm, in: Nature Inspired Cooperative Strategies for Optimization (NICSO 2010), Springer, 2010, pp. 65–74.
[94] S. Yadav, S. Nagpal, et al., An improved collaborative filtering based recommender system using bat algorithm, Procedia Comput. Sci. 132 (2018) 1795–1803.
[95] S.A. Uymaz, G. Tezel, E. Yel, Artificial algae algorithm (AAA) for nonlinear global optimization, Appl. Soft Comput. 31 (2015) 153–171.
[96] R. Katarya, O.P. Verma, Effectual recommendations using artificial algae algorithm and fuzzy c-mean, Swarm Evol. Comput. 36 (2017) 52–61.
[97] Y. Koren, R. Bell, C. Volinsky, Matrix factorization techniques for recommender systems, Computer 42 (8) (2009) 30–37.
[98] N.N. Liu, B. Cao, M. Zhao, Q. Yang, Adapting neighborhood and matrix factorization models for context aware recommendation, in: Proceedings of the Workshop on Context-Aware Movie Recommendation, ACM, 2010, pp. 7–13.
[99] Y. Shi, M. Larson, A. Hanjalic, Mining mood-specific movie similarity with matrix factorization for context-aware recommendation, in: Proceedings of the Workshop on Context-Aware Movie Recommendation, ACM, 2010, pp. 34–40.
[100] L. Baltrunas, B. Ludwig, F. Ricci, Matrix factorization techniques for context aware recommendation, in: Proceedings of the Fifth ACM Conference on Recommender Systems, ACM, 2011, pp. 301–304.
[101] Y. Fang, Y. Guo, A context-aware matrix factorization recommender algorithm, in: Software Engineering and Service Science (ICSESS), 2013 4th IEEE International Conference on, IEEE, 2013, pp. 914–918.
[102] Y. Shi, M. Larson, A. Hanjalic, Mining contextual movie similarity with matrix factorization for context-aware recommendation, ACM Trans. Intell. Syst. Technol. (TIST) 4 (1) (2013) 16.
[103] Y. Zheng, B. Mobasher, R. Burke, CSLIM: Contextual SLIM recommendation algorithms, in: Proceedings of the 8th ACM Conference on Recommender Systems, ACM, 2014, pp. 301–304.
[104] D. Kim, C. Park, J. Oh, S. Lee, H. Yu, Convolutional matrix factorization for document context-aware recommendation, in: Proceedings of the 10th ACM Conference on Recommender Systems, ACM, 2016, pp. 233–240.
[105] C. Ono, Y. Takishima, Y. Motomura, H. Asoh, Context-aware preference model based on a study of difference between real and supposed situation data, in: International Conference on User Modeling, Adaptation, and Personalization, Springer, 2009, pp. 102–113.
[106] T.G. Kolda, B.W. Bader, Tensor decompositions and applications, SIAM Rev. 51 (3) (2009) 455–500.
[107] P. Ishanka, T. Yukawa, T. Yukawa, An analysis of emotion and user behavior for context-aware recommendation systems using pre-filtering and tensor factorization techniques, Glob. J. Comput. Sci. Technol. 18 (1) (2018).
[108] S. Maroulis, I. Boutsis, V. Kalogeraki, Context-aware point of interest recommendation using tensor factorization, in: 2016 IEEE International Conference on Big Data (Big Data), IEEE, 2016, pp. 963–968.
[109] C. Zheng, E. Haihong, M. Song, J. Song, CMPTF: Contextual modeling probabilistic tensor factorization for recommender systems, Neurocomputing 205 (2016) 141–151.
[110] W. Luan, G. Liu, C. Jiang, L. Qi, Partition-based collaborative tensor factorization for POI recommendation, IEEE/CAA J. Autom. Sinica 4 (3) (2017) 437–446.
[111] H.-S. Park, J.-O. Yoo, S.-B. Cho, A context-aware music recommendation system using fuzzy Bayesian networks with utility theory, in: International Conference on Fuzzy Systems and Knowledge Discovery, Springer, 2006, pp. 970–979.
[112] M.-H. Park, J.-H. Hong, S.-B. Cho, Location-based recommendation system using Bayesian user's preference model in mobile devices, in: International Conference on Ubiquitous Intelligence and Computing, Springer, 2007, pp. 1130–1139.
[113] D. Pomerantz, G. Dudek, Context dependent movie recommendations using a hierarchical Bayesian model, in: Canadian Conference on Artificial Intelligence, Springer, 2009, pp. 98–109.
[114] T. De Pessemier, T. Deryckere, L. Martens, Extending the Bayesian classifier to a context-aware recommender system for mobile devices, in: Internet and Web Applications and Services (ICIW), 2010 Fifth International Conference on, IEEE, 2010, pp. 242–247.
[115] R. Gupta, A. Jain, S. Rana, S. Singh, Contextual information based recommender system using singular value decomposition, in: Advances in Computing, Communications and Informatics (ICACCI), 2013 International Conference on, IEEE, 2013, pp. 2084–2089.
[116] P. Sitkrongwong, S. Maneeroj, P. Samatthiyadikun, A. Takasu, Bayesian probabilistic model for context-aware recommendations, in: Proceedings of the 17th International Conference on Information Integration and Web-Based Applications & Services, ACM, 2015, pp. 1–10, Article 22.
[117] P. Samatthiyadikun, A. Takasu, S. Maneeroj, Multicriteria collaborative filtering by Bayesian model-based user profiling, in: 2012 IEEE 13th International Conference on Information Reuse & Integration (IRI), IEEE, 2012, pp. 124–131.
[118] Q. Yuan, G. Cong, K. Zhao, Z. Ma, A. Sun, Who, where, when, and what: A nonparametric bayesian approach to context-aware recommendation and search for twitter users, ACM Trans. Inf. Syst. 33 (1) (2015) 2.
[119] Q. Yuan, G. Cong, Z. Ma, A. Sun, N.M. Thalmann, Who, where, when and what: Discover spatio-temporal topics for twitter users, in: Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2013, pp. 605–613.
[120] M. Grčar, B. Fortuna, D. Mladenič, M. Grobelnik, Knn versus SVM in the collaborative filtering framework, in: Data Science and Classification, Springer, 2006, pp. 251–260.
[121] P.G. Campos, A. Bellogín, F. Díez, J.E. Chavarriaga, Simple time-biased KNN-based recommendations, in: Proceedings of the Workshop on Context-Aware Movie Recommendation, ACM, 2010, pp. 20–23.
[122] X. Chen, X. Liu, Z. Huang, H. Sun, Regionknn: A scalable hybrid collaborative filtering algorithm for personalized web service recommendation, in: Web Services (ICWS), 2010 IEEE International Conference on, IEEE, 2010, pp. 9–16.
[123] Y. Zhang, GroRec: A group-centric intelligent recommender system integrating social, mobile and big data technologies, IEEE Trans. Serv. Comput. 9 (5) (2016) 786–795.
[124] C. Cortes, V. Vapnik, Support-vector networks, Mach. Learn. 20 (3) (1995) 273–297.
[125] K. Oku, S. Nakajima, J. Miyazaki, S. Uemura, Context-aware SVM for context-dependent information recommendation, in: Proceedings of the 7th International Conference on Mobile Data Management, IEEE Computer Society, 2006, p. 109.
[126] M. Kahng, S. Lee, S.-g. Lee, Ranking in context-aware recommender systems, in: Proceedings of the 20th International Conference Companion on World Wide Web, ACM, 2011, pp. 65–66.
[127] A.A. Kothari, W.D. Patel, A novel approach towards context based recommendations using support vector machine methodology, Procedia Comput. Sci. 57 (2015) 1171–1178.
[128] X. Wang, F. Luo, Y. Qian, G. Ranzi, A personalized electronic movie recommendation system based on support vector machine and improved particle swarm optimization, PLoS One 11 (11) (2016) e0165868.
[129] N. Sahoo, P.V. Singh, T. Mukhopadhyay, A hidden Markov model for collaborative filtering, Mis Q. (2012) 1329–1356.
[130] M.H. Aghdam, Context-aware recommender systems using hierarchical hidden Markov model, Physica A 518 (2019) 89–98.
[131] T. Li, M. Choi, K. Fu, L. Lin, Music sequence prediction with mixture hidden markov models, in: 2019 IEEE International Conference on Big Data (Big Data), IEEE, 2019, pp. 6128–6132.
[132] C. Huang, Q. Wang, D. Yang, F. Xu, Topic mining of tourist attractions based on a seasonal context aware LDA model, Intell. Data Anal. 22 (2) (2018) 383–405.
[133] J. Yuan, F. Sivrikaya, F. Hopfgartner, A. Lommatzsch, M. Mu, Context-aware LDA: Balancing relevance and diversity in TV content recommenders, in: 2nd Workshop on Recommendation Systems for Television and Online Video, 2015, http://eprints.gla.ac.uk/108455/.
[134] V. Kuleshov, D. Precup, Algorithms for multi-armed bandit problems, 2014, arXiv preprint arXiv:1402.6028.
[135] L. Tang, Y. Jiang, L. Li, C. Zeng, T. Li, Personalized recommendation via parameter-free contextual bandits, in: Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval, ACM, 2015, pp. 323–332.
[136] C. Zeng, Q. Wang, S. Mokhtari, T. Li, Online context-aware recommendation with time varying multi-armed bandit, in: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2016, pp. 2025–2034.
[137] S. Li, A. Karatzoglou, C. Gentile, Collaborative filtering bandits, in: Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval, ACM, 2016, pp. 539–548.

[138] X. Zhang, Q. Zhou, T. He, B. Liang, Con-CNAME: A contextual multi-armed bandit algorithm for personalized recommendations, in: International Conference on Artificial Neural Networks, Springer, 2018, pp. 326–336.
[139] Q. Zhou, X. Zhang, J. Xu, B. Liang, Large-scale bandit approaches for recommender systems, in: International Conference on Neural Information Processing, Springer, 2017, pp. 811–821.
[140] LDOS-CoMoDa dataset, 2019, https://www.lucami.org/en/research/ldos-comoda-dataset/. (Accessed 20 July 2019).
[141] KDD Cup 2012 track 2 dataset, 2019, https://www.kaggle.com/c/kddcup2012-track2/data. (Accessed 23 February 2019).
[142] Y. Zheng, B. Mobasher, R. Burke, CARSKit: A java-based context-aware recommendation engine, in: Proceedings of the 2015 IEEE International Conference on Data Mining Workshop (ICDMW), in: ICDMW '15, IEEE Computer Society, USA, 2015, pp. 1668–1671, http://dx.doi.org/10.1109/ICDMW.2015.222.
[143] F. Harper, J.A. Konstan, The movielens datasets: History and context, ACM Trans. Inter. Intell. Syst. 5 (4) (2015) 1–19.
[144] Movielens dataset, 2019, https://grouplens.org/datasets/movielens/. (Accessed 25 April 2019).
[145] CAMRa 2011 challenge dataset, 2011, http://2011.recsyschallenge.com/dataset. (Accessed 07 September 2011).
[146] Netflix prize dataset, 2019, https://www.kaggle.com/netflix-inc/netflix-prize-data. (Accessed 23 December 2019).
[147] L. Baltrunas, M. Kaminskas, B. Ludwig, O. Moling, F. Ricci, A. Aydin, K.-H. Lüke, R. Schwaiger, Incarmusic: Context-aware music recommendations in a car, in: E-Commerce and Web Technologies, Springer, 2011, pp. 89–100.
[148] T. Bertin-Mahieux, D.P. Ellis, B. Whitman, P. Lamere, The million song dataset, in: Proceedings of the 12th International Conference on Music Information Retrieval (ISMIR 2011), 2011, pp. 591–596.
[149] Last.fm dataset, 2016, http://millionsongdataset.com/. (Accessed 15 June 2016).
[150] Gowalla dataset, 2018, https://snap.stanford.edu/data/loc-gowalla.html. (Accessed 05 March 2018).
[151] D. Yang, D. Zhang, Z. Yu, Z. Yu, Fine-grained preference-aware location search leveraging crowdsourced digital footprints from LBSNs, in: Proceedings of the 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing, ACM, 2013, pp. 479–488.
[152] D. Yang, D. Zhang, Z. Yu, Z. Wang, A sentiment-enhanced personalized location recommendation system, in: Proceedings of the 24th ACM Conference on Hypertext and Social Media, ACM, 2013, pp. 119–128.
[153] Foursquare dataset, 2019, https://sites.google.com/site/yangdingqi/home/foursquare-dataset). (Accessed 25 March 2019).
[154] D. Dua, C. Graff, UCI Machine Learning Repository, University of California, Irvine, School of Information and Computer Sciences", 2017, http://archive.ics.uci.edu/ml. (Accessed 20 September 2019).
[155] Yelp dataset, 2019, https://www.yelp.com/dataset. (Accessed 18 January 2019).
[156] G. Takács, I. Pilászy, B. Németh, D. Tikk, Scalable collaborative filtering approaches for large recommender systems, J. Mach. Learn. Res. 10 (Mar) (2009) 623–656.
[157] K. Shin, L. Sael, U. Kang, Fully scalable methods for distributed tensor factorization, IEEE Trans. Knowl. Data Eng. 29 (1) (2016) 100–113.
[158] M. Unger, A. Bar, B. Shapira, L. Rokach, Towards latent context-aware recommendation systems, Knowl.-Based Syst. 104 (2016) 165–178.