IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received October 4, 2019, accepted October 31, 2019, date of publication November 21, 2019, date of current version December 6, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2954861

Multi-Criteria Review-Based Recommender System–The State of the Art

SUMAIA MOHAMMED AL-GHURIBI [1,2] AND SHAHRUL AZMAN MOHD NOAH [1]
[1] Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia, 43600 Bangi, Malaysia
[2] Department of Computer Science, Faculty of Applied Sciences, Taiz University, Taiz, Yemen

Corresponding author: Shahrul Azman Mohd Noah (shahrul@ukm.edu.my)

This work was supported in part by the Organization for Women in Science for the Developing World (OWSD), and in part by the Universiti Kebangsaan Malaysia through the Monetary Assistance under Grant DCP-2017-002/3.

ABSTRACT In recent times, the recommender systems (RSs) have considerable importance in academia, commercial activities, and industry. They are widely used in various domains such as shopping (Amazon), music (Pandora), movies (Netflix), travel (TripAdvisor), restaurant (Yelp), people (Facebook), and articles (TED). Most of the RSs approaches rely on a single-criterion rating (overall rating) as a primary source for the recommendation process. However, the overall rating is not enough to gain high accuracy of recommendations because the overall rating cannot express fine-grained analysis behind the user's behavior. To solve this problem, multi-criteria recommender systems (MCRSs) have been developed to improve the accuracy of the RS performance. Additionally, a new source of information represented by the user-generated reviews is incorporated in the recommendation process because of the rich and numerous information included (i.e. review elements) related to the whole item or to a certain feature of the item or the user's preferences. The valuable review elements are extracted using either text mining or sentiment analysis. MCRSs benefit from the review elements of the user-generated reviews in building their criteria forming multi-criteria review based recommender systems. The review elements improve the accuracy of the RS performance and mitigate most of the RS's problems such as the cold start and sparsity. In this review, we focused on the multi-criteria review-based recommender system and explained the user reviews elements in detail and how these can be integrated into the RSs to help develop their criteria to enhance the RSs performance. Finally, based on the survey, we presented four future trends based on this type of RSs to support researchers who wish to pursue studies in this area.

INDEX TERMS Recommender system, multi-criteria recommender system, user-generated reviews, review elements, sentiment analysis, text mining, multi-criteria review-based recommender system, recommender system accuracy.

I. INTRODUCTION

At present, there is a vast flow of information on the Web, and it continues to grow exponentially while providing users or customers with various resources pertaining to services such as products, hotels, and restaurants. Despite the benefits of such data, the vast flow of information causes challenges for users to deal with and choose from a huge number of options made available to them. This causes an information overload problem [1] and makes the decision-making process more complex. In this case, it is important to filter the information to a limited amount based on the current user/customer preferences in order to assist them in making the correct decision [2]. Such a filtering process is typically done by RSs, which are developed to solve the information overload problem by providing personalized suggestions of services (i.e. items) to specific customers according to their preferences [3].

RS has been proven to be significantly crucial in many fields and is widely used by various domains such as shopping (Amazon), music (Pandora), movies (Netflix), travel (TripAdvisor), restaurant (Yelp), people (Facebook) and articles (TED). There are many definitions of RSs including:

a. *A tool* to mine items and/or collect users' opinions to help users in their search process and suggests items related to their preferences [2], [4], [5].

The associate editor coordinating the review of this manuscript and approving it for publication was Biju Issac.

b. *A program* or software for content filtering that attempts to reduce the information overload problem, where users encountered a flood of data on the Web, by recommending personalized items to users depending on the items' information and/or users' preferences [6]–[8].

c. *A system* to manage information overload problem by collecting information, guiding users in a personalized way and providing individualized recommendations as output when there are many possible alternatives to choose from [9].

The problem of RSs can be identified as a way to assist users/customers to discover relevant items to suit their needs and most likely to their preferences [10]. Generally, a model of RS consists of two sets and a utility function, in which *Users* set contains all the users and *Items* set contains all the items that can be recommended to the users. The utility function calculates the suitability of a recommendation to a user $u \in Users$ an item $i \in Items$, which is declared as R: $Users \times Items \rightarrow R_0$, where R_0 is equal to either a real number or a positive integer within a specific range [11].

Typically, RS works through three phases [11]–[13] as follows:

a. ***Modeling Phase:*** This phase is focused on preparing the data that will be used in the next two phases. There are three cases for that, the first is building *a rating matrix* that contains the users as rows, items as columns and the value of each matrix's cell is the rating done by a user for a certain item. Second, building a *user profile* which is mostly a vector for each user that explains his preferences of an item as a whole or on some aspects of the item. Third, building an *item profile* that contains the features of a specific item.

a. ***Prediction Phase:*** This phase aims to predict the rating or score of unseen/unknown items for a specific user through a utility function depending on the extracted information during the modeling phase.

a. ***Recommendation Phase:*** This phase is an extension of the prediction phase where various approaches are applied to support the user's decision by filtering the most suitable items. It recommends/proposes new items to the user (i.e. a set of top-N items with the highest-predicted ratings) that is most likely to be interesting to him.

Figure 1 shows the three main phases of RS.

There are three main recommendation approaches which are content-based, collaborative-based and hybrid. Classical approaches rely on the users' ratings as the main source of input of the recommendation. Relying on a single-criterion (i.e. overall) rating for a recommendation is insufficient to give an accurate recommendation because the overall ratings cannot express fine-grained analysis behind the users' behaviors since it only expresses the coarse-grained analysis. It cannot be determined why the user choose such ratings. Thus, it is difficult to know the exact user's preferences. As a result, multiple-criteria decision analysis is combined with RS to form a multi-criteria recommender system (MCRS), in which the recommendation is based on multiple criteria, and not just on a single criterion.

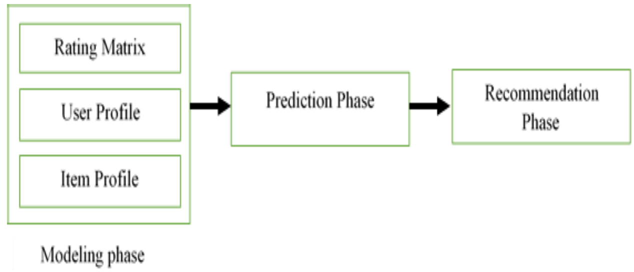


FIGURE 1. The phases of recommender system.

Besides the primary source (i.e. numeric rating) of the recommendation input, the user-generated reviews are also used as an alternative source because of the valuable and rich information they contained. The rich information from the reviews can be extracted as elements such as topics, features, overall score, and context, through analyzing the reviews using sentiment analysis or text analytics approaches. In this survey, we emphasized on MCRS especially in a multi-criteria review-based RS because of its effective role in enhancing the accuracy of the RS performance.

In the following content, the state-of-the-art is organized as follows: the RS approaches are described in Section 2, then the multi-criteria recommender system is explained in Section 3. After that, in Section 4, the user-generated reviews and the valuable elements that can be extracted from them are discussed. This is followed by Section 5 which is the main section because it contains the most recent researches in the multi-criteria review based recommender system approaches. Finally, the discussion and the forthcoming trends in MCRS are presented in Section 6 followed by the conclusion in Section 7.

II. RECOMMENDER SYSTEM APPROACHES

Approaches to the recommendation are usually categorized into four categories which include content-based, collaborative filtering, knowledge-based and hybrid.

A. CONTENT-BASED APPROACH

A content-based (CB) approach mines the appropriate recommendations for a user based on his recent behaviors according to what the user liked, bought or watched [14]. It generates the user profile from previously selected items by characterizing the user according to the item features and recommends items to the user based on the items that have similar features to the items that the user liked before [15]. It characterizes each user without having to compare his preferences to other users. Put differently, it does not use the information about other users' preferences or the similarities with other users [15], [16]. The process of CB approach can be summarized into the following steps [2], [17], [18]:

a. Item representation: The information source of the item description is used to extract the item's characteristics (i.e. features) to produce the structured item's representation.
b. Learning the user profile: A user profile is generated from the previous user's behaviors (i.e. explicit and implicit feedback) such as like/dislike of an item; assign a score to an item (rating) or writing a textual opinion about an item (comment).
c. Recommendations' generation: A list of items is recommended to the user by comparing the item's features with the user's profile and the items that are most likely to be interesting to the user are added to the list (i.e. top-ranked items).

This type of approach has been implemented in many domains [9] especially in recommending items that contain textual information such as websites, news, and articles. It also recommends activities such as travel, tourism, e-commerce, and TVs [15]. This approach is preferred for moderate-sized items.

Some of the CB approach advantages are:

a. It can give an explanation for recommending specific items (i.e. present the logic behind their recommendations) through providing a list of content features. This, in turn, can strengthen the user's confidence about the RS that reflects his own preferences [16].
b. Since this approach relies on the content of each item, not the ratings of other users, it gives several advantages as follows [19]:
    - It offers a high level of personalization in the recommendations.
    - It is scalable in terms of the number of users.
    - It can make recommendations for users with peculiar interests.
    - It has high security from malicious item creation and allows users to prevent viral marketing.

On the other hand, CB approaches have some disadvantages such as:

- The vast size of the items is considered a major problem because when the recommendation is made, the content of every item has to be examined to discover items that are most likely relatable to the user's interest [19]. This task is error-prone and time-consuming [20].
- User profiles are built based on the static characteristics of the items. As a result, there is a high probability that different users have similar profiles even if they have various preferences among these items, just because they commented on the same items [9].
- The over-specialization problem occurs in this type of approaches because users do not receive diverse or new items because of the restriction in his profile regarding the description of similar items [20].
- Lack of serendipity. Overspecialization can also cause the issue of serendipity, whereby users are being recommended with familiar items.

B. COLLABORATIVE FILTERING APPROACH

The collaborative filtering approach (CF) is the most popular technique used in RSs [21]. It generates the recommendation for a user based on the similarities among users who have similar preferences/interests to him in the past. This approach is based on the following hypothesis: people who agreed with a user in the past will also agree in the future [16]. It identifies the new user-item association by determining the relationships between users and the interdependencies between items [21]. It uses the implicit knowledge of a community of users on used items to identify the relationships of those items to other users who have not used/seen those items within the community [15]. This can be represented as a user $\times$ items matrix in which each cell represents the user rating of a particular item.

The first CF framework for RS was developed by Resnick *et al.* [22] called GroupLens. It recommends articles to the Netnews clients using the rating server, named Better Bit Bureaus (BBB) which gathers users' rating to predict other user's scores on articles based on the heuristic model that clients who agreed to the rating of articles in the past they will probably agree in the future.

CF can be grouped into two classes memory-based and model-based [9]. The memory-based CF type is a heuristic algorithm that predicts the item's rating based on other users' ratings, and can be classified into two methods [10]: user-based and item-based, the former identifies a set of neighbors (i.e. like-minded users) for a target user using ratings then recommend a set of items that interest his neighbors. While the latter, recommends items to a target user that are similar (i.e. has shared features) interests in the items that a user purchased, viewed or liked before. There are two approaches that are most frequently used to identify the user/item similarity, the Pearson correlation approach and cosine-based approach [1].

On the other hand, the model-based CF type [23] predicts user's rating of unseen items by developing models using different representative techniques such as the clustering models, Bayesian networks and Markov decision process. A survey by Su and Khoshgoftaar [23], provided a comparison between the CF classes as shown in Table 1.

CF approaches possess many advantages compared to other approaches, some of the main advantages are:

- Serendipity where novel and unfamiliar items are recommended.
- Able to recommend more subtle items and can capture more nuances around items.
- Flexible and suitable for various domains.
- No need to analyze the items' contents.

Generally, the performance of the CF approach depends on the availability of sufficient user participation [16]. It performs satisfactorily only when there is adequate rating information [23]. Depending on the ratings exposed CF approach to the following issues [21], [24]:

TABLE 1. Comparison between CF Classes.

CF Class	Advantages	Shortcomings
Memory-based	• Easy to implement. • Easy to add new data incrementally. • The content of recommended items need not consider. • Co-rated items have been scaled well.	• Dependent on human ratings. • In sparse data, the recommendation performance is decreased. • New items/users cannot be recommended (i.e. cold start problem). • The large datasets have limited the scalability.
Model-based	• The scalability and sparsity problem are better addressed. • The prediction performance is improved. • An intuitive rationale is given for recommendations	• Building the model is expensive • A trade-off is done between scalability and prediction performance. • Useful information is lost through dimensionality reduction techniques.

a. Sparsity Problem: One of the major problems that complicate the personalized item ranking process is data sparsity because items cannot be reliably linked to users [25], causing a limitation in the recommendation's effectiveness and limited coverage of recommendation space [26].
This problem occurs due to the following issues [26], [27]:
- Insufficient or missing information of either the user or item or both in the dataset during the process of filling the ratings (user-item) matrix. The complexity of gathering the items' ratings.
- Expressing user's preferences about items as a rating is a complicated process.

b. Cold Start Problem: This problem happens in the case of new users who do not provide any ratings as yet or new items that have not been rated [28]. It can be considered as a particular case of the sparsity problem in which most of the cells of the item-user interaction matrix contain null values [29]. The CF approach is not able to generate accurate recommendations for new users or items without sufficient existing data on them [24].

c. Scalability: The number of users and items in a system grows rapidly. For example, the behavior of such a user per day may result in his stored data reaching the size of TBs in some popular websites [30]. Furthermore, the RS should respond in less than a second to keep users satisfied and to enable them to continuously engaged with the RS [30]. As a result, both large-scale datasets and responding time create a challenge in designing efficient RS and as a result, it demands colossal computing resources.

d. Rating bias: In the CF approach, recommendations are based on users' ratings, but these ratings cannot show users' preferences or their clear opinion on some criteria which makes it difficult in interpreting these ratings.

C. KNOWLEDGE-BASED APPROACH

This approach is applied in some cases when both content-based and collaborative-based approaches cannot work properly because no sufficient ratings are available for a specific item at hand which affects the recommendation process [31]. For instance, recommending items that are rarely purchased like cars, houses and financial services. This approach uses the user's knowledge of the item domain to recommend items that will best satisfy his requirements [32]. The main advantage and strength of this approach are that no-existence of the early-rater problems and cold start problems. While a corresponding drawback is that it requires knowledge engineering with all of its attendant difficulties to understand the item's domain satisfactorily [33].

D. HYBRID APPROACH

This approach aims to mitigate the weakness of both CF and CB and benefit from their strengths by integrating two or more recommendation components or algorithm's implementations in a single recommendation system to enhance RS accuracy and gain better performance [2], [34]. When the hybrid approach is generated through hybridizing two or more algorithms, two major points must be taken into account [2]: the first is the recommendation models that declare the required inputs and the determination on which the hybrid recommender will be based on. The second point is determining the strategy that will be used within the hybrid recommender [35].

Although hybrid approaches may overcome the limitation of both CB and CF approaches and enhance the prediction performance, it is expensive to implement, increases the complexity and needs external information that is mostly unavailable [23].

CONVENTIONAL SINGLE-RATING RECOMMENDATION PROBLEM

The aforementioned conventional approaches mostly rely on a single-criterion rating for generating predictions. This rating is considered as the overall satisfaction of a user for the item [2]. In other words, most of the RS approaches work in a two-dimensional space (Users and Items), and the RS uses the previous ratings made by a user to predict the utility function for the user of an item represented as a totally ordered set as $R : Users \times Items$ [36]. However, the overall rating is not enough to gain high-performance recommendations because the overall rating is only a numeric rating with a specific scale that cannot express a fine-grained analysis about the underlying rationale behind the users' rating. It expresses the coarse-grained rating only (i.e. overall rating cannot reflect the details of user preferences or interest toward each part of the item to understand users' opinions and analyze users' behaviors) [8], [37], [38].

For example, when a user gives a high rating about an item, it does not mean that the user likes the item as a whole. There is still a probability that he dislikes some specific features (i.e. aspects) of that item. Likewise, a low rating does not imply that the user dislikes everything about the item. Additionally, when the user puts the overall ratings, he places various emphases on various aspects and this has a significant effect on the final decision made by the user [21].

To overcome the shortage of using a single criterion (or an overall rating) in RS, multiple-criteria decision analysis is combined with RS to form a multi-criteria recommender system (MCRS) to develop the overall accuracy and performance of the RS [8], [36]. Thus, by adopting multi-criteria decision analysis, an item recommendation process is the decision process, a potential user is the decision-maker, the item attributes are the criteria and the items are the decision alternatives [2]. The following section presents a survey of various approaches used for supporting the multi-criteria recommendation.

III. MULTI-CRITERIA RECOMMENDER SYSTEM

Multiple-criteria decision-making or Multiple criteria decision analysis (MCDA) is a sub-discipline of operations research and management science. It aims to develop tools and methodologies to construct a convincing and reliable model for addressing complicated decision problems including multiple criteria goals or multi-alternatives [8].

The idea of combining MCDA with RS is to recommend items that meet users' personalized needs. In this case, personalization refers to "the ability to provide services and content that are tailored to users depending on the knowledge about their behaviors and preferences" [39]. As such, RSs will be able to comprehend how the user thinks and why the user likes an item and not only what the user likes [8].

Both single-criterion RS and MCRS have the same goal which is to identify items that are suitable and relevant to fit the user's preferences. The difference between them is that the MCRS has more detailed information about both the items and users that can be used to efficiently enhance the recommendation performance. Generally, the rating function in the MCRS is described as follows:

> R: Users $\times$ Items $\rightarrow R_0 \times R_1 \times .... \times R_{k;}$ Where R_0 is the overall rating and $R_1, R_2, \ldots, R_k$ is the rating values for each singular criterion.

As an illustration, consider a hotel RS meant to recommend a suitable hotel based on the needs and requirements of the target user. In the conventional single criteria RS, a user (U) provides one rating (overall rating) for the hotel (I) that he has visited, denoted R (U, I). Specifically, the RS calculates the predicted rating of the unvisited hotel based on other users' ratings that have similar preferences for the target user. The precise choice of the relevant users is crucial to gain an accurately predicted rating and high-performance recommendation. So, if two users (U1) and (U2) have rated

TABLE 2. Multi-Criteria hotel recommender system.

	H1	H2	H3	H4
U1	$5_{2,2,8,8}$	$7_{5,5,9,9}$	$7_{5,5,9,9}$	?
U2	$5_{8,8,2,2}$	$7_{9,9,5,5}$	$7_{9,9,5,5}$	$9_{8,9,10,8}$
U3	$6_{3,3,9,9}$	$7_{6,6,8,8}$	$6_{4,4,8,8}$	$5_{2,2,8,8}$

their overall satisfaction of the visited hotel 5 out of 10 as presented in Table 2, they will be considered as neighbors and the predicted rating of the user (U1) for the unvisited hotel (H4) is calculated using the ratings of the user (U2) and it will be 9 out of 10. On the other hand, in a multi-criteria rating, a user provides ratings for multiple features (i.e. attributes) of an item. For example, in a hotel RS with four criteria such as room, price, location, and cleanness, the users will provide ratings for these four criteria.

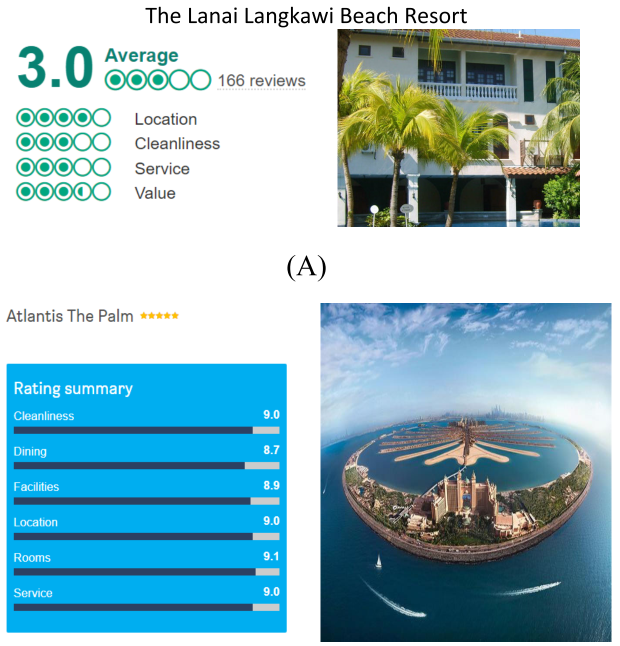
Suppose we have three users' ratings for the four features of the hotel (H2) plus an overall rating as illustrated in Table 2: U1($7_{overall}$, 5_{room}, 5_{price}, $9_{location}$, $9_{cleaness}$), U2($7_{overall}$, 9_{room}, 9_{price}, $5_{location}$, $5_{cleaness}$), and U3($7_{overall}$, 6_{room}, 6_{price}, $8_{location}$, $8_{cleaness}$). If we recommend based on a single-criterion rating only, all the three users are considered as neighbors because all of them has an overall rating of 7 out of 10 for the hotel H2. Considering the three users as neighbors, despite the difference in their preferences will affect the accuracy of the recommendation's performance. While, in the MCRS, when the choice of neighbors is based on the rating of each item's feature, users U1 and U2 are not neighbors because they chose different ratings for the hotel's features even if they have a similar overall rating, so the predicting rating for H4 for U1 will be 5. These additional details of users' preferences from the item's features help the RS to recommend more accurate items and enhance the RS performance in general.

MCRS becomes a significant trend in studying RS and it is successful in gaining the attention of both the industry and research [11]. The numerous research prove that by using MCRS, the recommendation's accuracy outperforms the single-rating RS [11], [40].

The item's criteria in MCRS are either explicitly represented or implicitly represented in the user-generated reviews. The next section will discuss these two types of item's criteria.

A. MULTI-CRITERIA RECOMMENDER SYSTEM USING EXPLICIT USER PREFERENCES

In this type, the user gives ratings to each of the item's features with or without the rating of the whole item. The user's preferences are known directly from the users' ratings on the items'



(B)

FIGURE 2. **Example of hotel rating from TripAdvisor.**

features (explicitly stated). As an example, Figure 2 shows two ratings for two hotels from TripAdvisor; each rating contains an overall rating and multiple criteria ratings, Hotel A contains 4 features/criteria (location, cleanliness, service, and value) while hotel B contains six features (cleanliness, dining, facilities, location, rooms, and service).

A considerable number of research applies this type of MCRS as illustrated in the works of [1], [8], [40]–[42]. Additionally, there are some researches that apply MCRS recently and the following are three of them:

- **Wasid and Ali [43]** proposed a multi-criteria RS using a clustering approach. The main idea of this approach is to find more similar neighbors of a user within the user's cluster in order to improve the recommendation set. To achieve that, initially the users' preferences are extracted from the multi-criteria ratings that they have given for items and the user cluster centers (C) are defined based on the extracted preferences. Then, the Euclidean distance is used to assign the closest C for each user and the Mahalanobis distance is used to compute the top-N neighbors for a user in the same cluster. After that, the predicting rating of an item for the user is computed based on similar neighbors who have been chosen from the same cluster. The approach is evaluated using Yahoo! Movies dataset and the users who have ratings of at least 20 movies are chosen, yielding to 484 users, 945 movies and 19,050 ratings. An experiment is done to compare the Mean Absolute Error (MAE) using clustering and without clustering. The result shows that their clustering method produces the best result with MAE equal to 2.175.
- **Zheng [44]** developed a utility-based multi-criteria recommender system in which the items are recommended to a user based on the utility function of each item for the user. The utility function is built using the multi-criteria ratings as the similarity between the vector of user evaluations and the vector of user expectations (i.e. the higher degree of over-expectations, the higher the similarity between the vectors of the expectation and the user evaluations). Three similarity measures are used to calculate the utility score (i.e. Pearson correlation, cosine similarity, and Euclidean distance). The user expectations are learned by three optimization learning-to-rank methods (i.e. Pointwise ranking, Pairwise Ranking, and Listwise Ranking). Evaluation of the proposed method was done using two datasets which are TripAdvisor and Yahoo! Movies [45]. TripAdvisor contains of 14,300 hotels, 1502 users (users with at least 10 ratings are chosen), and 22,130 ratings including seven criteria ratings (i.e. price, location, quality of rooms, cleanliness, convenience of the hotel, service experience of check-in, and particular business services). While Yahoo! Movies contains 2,162 users who have issued 62,739 ratings on 3,078 movies that have four criteria (i.e. story, direction, visual effects, and acting). The developed method is compared with four baselines: the matrix factorization, the linear aggregation model 40], the hybrid context model [46] and the criteria chain model [47]. The results outperform the baselines in terms of precision and NDCG and the Pearson correlation measure gives the best result and the Listwise ranking gives the most outstanding performance.
- **Tallapally et al. [48]** used a deep neural network technique called stacked autoencoders to solve the shortage of single rating RS through using multi-criteria RS. The conventional stacked autoencoders are extended to fit with the multi-criteria ratings by adding an extra layer which acts as an input layer to the autoencoders. The input which is the multiple criteria ratings is connected to the intermediate layer which is represented by the items. The intermediate layer is connected to N consecutive encoding layers where the latent representation for each item is encoded. The last encoding layer is connected to N consecutive decoding layers. The last layer is the output layer in which the items' overall rating is predicted. An experiment is conducted to evaluate the effectiveness of the proposed network on two datasets: the TripAdvisor and Yahoo! Movies (YM). For TripAdvisor users who rated at least five hotels and hotels that have rated by at least five users are chosen (5-5) leading to 3,550 hotels rated by 3,160 users by 19,374 ratings. Similarly, YM dataset forms three subset YM 5-5, YM 10-10 and YM 20-20. The proposed network result is compared with many baselines such as [1], [47] and [49]. The result outperforms all the compared baselines in terms of the following performance metrics MAE, F1, and both Good Items MAE

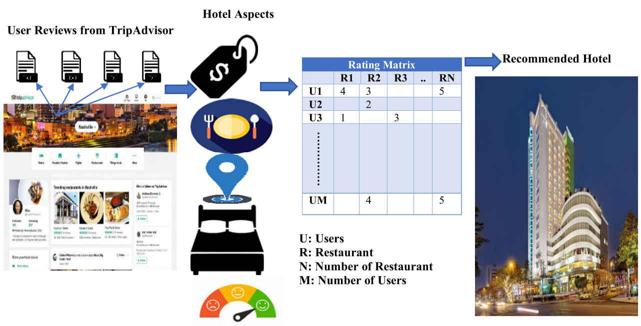


FIGURE 3. **Multi-Criteria RS with Users reviews.**

and Good Predicted Items MAE that was introduced by Cacheda *et al.* [50].

B. MULTI-CRITERIA RECOMMENDER SYSTEM USING IMPLICIT USER PREFERENCES

In the first type of MCRS, users should give ratings for each feature of the item regardless of whether he is interested in the features or otherwise. Unlike this type, users provide opinions *only on the item's feature that they are interested in* through writing comments (i.e. reviews) that express their feelings or opinions about their experiences with the items. This type of approach is claimed to be more accurate in determining the users' preferences because users will write exclusively about what they concerned with regarding the items. This, in turn, will enhance the accuracy of the RS, because *the more accurate the user's preferences are determined, the more accurate the recommendation provided to the user*.

In this type, the criteria of the RS process are implicitly represented and they need to be extracted from the valuable information of the user-generated reviews. Figure 3 shows an example of multi-criteria RS where the users' reviews are collected from TripAdvisor to extract the hotel's criteria (i.e. aspects) such as price, food, location, and bed using sentiment analysis methods. These aspects are used in building the rating matrix. Then recommend a hotel to a specific user based on the criteria that are mentioned in his reviews. The extracted valuable information can be summarized as the review elements. In the following section, we will explain the reviews, their benefits in RS and the review elements. Then, we will explore various research that have utilized this type of MCRS and explain how the review elements enhanced the recommendation process.

IV. USER-GENERATED REVIEWS

Recently, vast growth in e-commerce and social Websites have been observed and these Websites encourage users to incorporate their experience with each other. Therefore, there is a significant number of online comments (i.e. reviews) about various topics such as hotels, products, movies, restaurants, travel, and services and they continue to increase on a daily basis [15], [51]. These reviews are valuable resources for users because they help them in making decisions before consuming or buying a particular item. Such reviews may provide an overall overview of the items or specific comments on certain features of the items [7]. The reviews may also indicate users' preferences.

Many users are affected by the other customers' reviews because it is considered as trustworthy information compared to the vendor's information [52]. This, in turn, influences the buying behavior which also helps the vendors and companies to manage and improve their products and develop new ones based on the users' preferences which can be extracted from the written reviews [53], [54].

The user's review exhibits distinct characteristics: it is brief, prone to the occurrence of noise (i.e. misspelling, many hyperlinks and may include advertisement), written in the form of plain/textual text without a standard structure or fixed rules and may contain emoticons. The user writes them just to explain his usage experience with the item [15], [51]. Due to the previous characteristics of the reviews, most RS do not use them in generating recommendations because of the difficulties encountered by the machines to comprehend written natural language compared to other structured data sources [25].

A. ANALYSING USER-GENERATED REVIEWS

There are many fields involved in processing textual reviews and extracting the valuable information from the reviews such as natural language processing, text mining and opinion mining (or sentiment analysis). In this survey, we are more interested in the involvement of sentiment analysis with RS because the sentiment analysis field will help us in determining the user's preferences by analyzing the user's sentiment behind his reviews. Sentiment analysis is a discipline derived from artificial intelligence, information retrieval, and natural language processing. It focuses on predicting the positive or negative polarity of the given entity. Sentiment analysis usually works in three levels: document-level, sentence-level, and aspect-level.

Leung *et al.* [27] is the first researcher who indicated the potential advantages of integrating sentiment analysis field with the CF approaches to improve the accuracy of the RS performance through calculating an inferred rating from users' reviews when the explicit rating is not available. He developed a rating inference framework that consists of two parts; the first part is a rating inference which is responsible for calculating the inferred rating from user's reviews through extracting the opinion words (OWs) from the reviews and aggregating the sentiment polarity of such OWs to determine an inferred rating. While the second part is the recommendation process using the CF approach which recommends items to users based on the calculated inferred rating. An experiment is done to infer users' ratings using the MovieLens-100k dataset which contains 1477 movies, 1065 users (i.e. users with more than 10 reviews) and 30,000 reviews (i.e. reviews with user-specified ratings). The work of Leung *et al.* [27] is considered as a hypothesis because there is no evaluation for the RS performance after the inferred rating is calculated.

After Leung *et al.* [27], **Aciar *et al.*** [16] made the first attempt to use the user reviews in building RS through developing an ontology to convert the review content into a structured form that is used to provide recommendations. The ontology model is built manually with two main components of opinion qualities, which show the user's expertise regarding the product; and the product quality, which indicates the rating that the user made for the product features. Each review is considered as an ontology instance and it is automatically mapped onto the ontology through the mapping process. After all the reviews are mapped onto the ontology, the product's overall assessment (OA) score (i.e. the final score for the product based on each product feature's estimation) is determined through performing a set of computations. Using OA, this application gives a recommendation to the user about the product that has the highest OA based on the features that are mentioned in the user's request. For the application evaluation, the authors have yet to do an empirical test for measuring the performance of the proposed RS. The authors claim that their application overcomes the cold start problem in the CF techniques. It is beyond the scope of this paper to discuss various research that exploit users reviews in providing recommendations. Interested readers may refer to the review done by Adomavicius and Tuzhilin [10].

B. ADVANTAGES OF USING USER-GENERATED REVIEWS IN RECOMMENDER SYSTEMS

Although there are some difficulties in processing users' reviews, there are major advantages that RS can get benefit from them to enhance its performance especially the reviews that can be broadly accessed over the internet. The following are some of the reviews' advantages [37], [55], [56]:

a. Alleviate the data sparsity problem in the case of missing ratings. Reviews provide valuable and natural information about the user's interests which can be extracted and inferred.
b. Relieve cold start problem either for a new user or a new item. It can be considered as a special instance of the sparsity problem. There are three cases for causing this problem: the first is a user who enters the system for the first time (totally new), the second is a user who has not made many ratings (limited experience) and the third is a user with incomplete (i.e. partial) preferences. Similarly, for new items either totally new items are added to the system or items have no ratings. Reviews can solve this type of problem by providing information that is used to improve recommendation such as the work of **Wang *et al.* [57]**.
c. In the case of dense data, the reviews still provide a valuable and detailed information that can be used to enhance the recommendation accuracy such as: check the rating quality (compared both the user's star rating with the inferred rating from the review's text, or from review's helpfulness), derive users' aspects or context-dependent-aspect or context- independent-aspect preferences.
d. The reviews provide rich and useful information in some domains like tourism and travel, where it is difficult to express user's preferences as scalar ratings or collect numerical ratings for items.
e. Reviews help to construct both the user model and item model precisely because they contain much finer-grained sentiment trend for various features of a single item.

1. **Total Review Polarity Score:** yellow underline marks the sentiment words for calculating the total score.
2. **Term / Features:** green lines show either a term (most frequent words) or a feature (concept)
3. **Context:** blue lines
4. **Comparative words:** red lines
5. **Helpful:** brown lines.

FIGURE 4. Example of review elements.

C. REVIEW ELEMENTS

After agreeing on the review's usefulness on improving the RS performance, we can summarize the rich and valuable information (called elements) that can be extracted from the users' reviews as follows:

a. Total Review of Polarity Score

A user's overall opinion can be inferred from his written review about an item whether he or she likes it or not (positive or negative sentiment), this overall sentiment can be converted into implicit ratings. Implicit rating (also called virtual rating) is generated by aggregating the opinion words of the review (i.e. mostly the adjectives or adverb) and then calculating the sentiment polarity of each opinion word. For example, the opinion words in the review in Figure 4 are *newly, strategic, friendly, helpful, spacious, nice, clean, tidy* and *thumbs up*. The total review polarity score is the summation of all the polarities of the extracted opinion words, which is done using either machine learning methods or text mining methods. There is an implicit relationship between the user's rating and his expressed comment [15]. As a result, the implicit rating takes the role of the explicit rating (also called the actual rating) in case the explicit rating is not available such as in [58]. Additionally, in the case where the explicit rating is available, the implicit rating can be used either to enhance the actual rating [59] or be used for both ratings to enhance the performance of RSs further as illustrated in [21], [60].

b. Review Terms
Review terms are the words that are frequently used or that occurred in the reviews and extracting them is the easiest way for analyzing users' reviews. For example, the terms in the review in Figure 4 are *hotel, location, staff, room,* and *budget*. The Term Frequency-Inverse Document Frequency (TF-IDF), weight scheme is the most widely used statistical method for measuring the importance of terms. In this case, the items that are recommended to a user are based on his term-based profile. Researches that use this type of element prove its usefulness in improving the RS performance such as in [20], [61].

c. Review Feature/Aspect/Topic
Review aspect can be defined as a concept that depicts a topic of each item's domain and it is restricted to exist in every item; each aspect consists of a set of words (terms) (e.g., the following terms "attitude, service, waitress, waiter" correspond to the "Service" aspect). Aspects comprise of either noun or noun phrases that are common in the domain being analyzed and must be in every item. In contrast, the terms consist of nouns that most frequently occur in the reviews and it is not necessary that every term present themselves in all the items set [7]. Some researchers use the terminology of feature for aspect such as [62], [63], while others use topic such as [25] and all the terminologies (aspect, feature, and topic) have the same meaning. The identification of aspects is usually based on two approaches: heuristic-based and model-based [64]. The former approach identifies a set of manually-selected keywords (fix aspects) and then searches for other related terms by applying the clustering method [65] and relying on the calculation of the relationship between the aspect and the candidate's terms [66], [67]. While in the latter approach, the aspects are automatically extracted (denoted as learned aspect) and the most popular model that is applied is Latent Dirichlet Allocation (LDA) [68]. A comparison between the fix and learned aspects will be discussed in detail later. The review as a whole gives a coarse-grained opinion about the user's preferences, while the review-aspect gives a fine-grained opinion about the user's preferences. An aspect-based recommendation is claimed to enhance the performance of RSs due to its ability in determining the specific preferences of the user [9], [38], [69]. Besides the advantage of aspect-extraction in enhancing the accuracy of the RS, one point must be taken into consideration, which is the number of the extracted aspects, because the high number of the candidate aspects will negatively affect the RS's performance and lead to more sparse data. As a result, aspect selection is of importance that may influence the performance of the RSs [7]. In the example illustrated in Figure 4, three aspects have occurred which are *location, staff,* and *room*.

d. Review Context
Review context is the circumstance within which a user expresses his opinion about the item or some feature of the item. For example, in Figure 4 the context is *traveling for business*. Like aspects, review contexts are either pre-defined (fixed) contexts or learned contexts that are automatically extracted. It can be discovered through rule-based reasoning, keyword matching or using a classifier such as LDA-based classifier [9]. The review context proves its benefit in enhancing the recommendation performance by either combining it with the explicit rating to predict the user rating for an item in a specific context [64] or using it in the user modeling as proposed by Chen and Chen [70] through using context-dependent aspect preferences or context-independent aspect preferences.

e. Review Comparative Words
A user sometimes writes his opinion about an item by comparing it with other items in terms of some specific features. This type of element called comparative opinion where it identifies if item A is superior or inferior to item B in some shared aspect. The comparative words can be extracted either using graph relations or a set of linguistic rules and then use them in RSs in order to enhance the items' ranking quality such as in [55], [71], [72]. In the review illustrated in Figure 4, 'best' is a comparative word used to emphasize that the price of the hotel is the best compared to others.

f. Review Emoticons
When a user writes a review, he can reflect his mood using some symbolic representations of icons (faces) (e.g., smile, joy, sadness, distress faces). Most of the reviews contain icons, (41% of the reviews contain emoticons [73]), which make them available for use in the recommendation process in spite of the fact that it is harder to detect them compared to other review elements. Using these icons, we can infer if the user likes the item or not (overall rating) and then use this information for better item recommendation for the users as seen in [74]. Additionally, these icons can be aggregated with other review elements to enhance the RS performance such as in [73].

g. Review Helpfulness
For every user's review, readers can vote by clicking the helpful button for the review if they find it useful for them. These votes can be used in the RS to make a better predictions, especially for determining the rating's quality score such as in [37], [75]. In other words, the more votes were given for a review, the more the rating's quality score is assigned. For example in Figure 4, the review helpfulness is equal to two which means that two users get to benefit from this review.

D. ASPECTS TYPE

As mentioned in the previous section, there are two types of aspects, fix aspects and learned aspects. In the fix aspects type, experts define a fixed set of aspects manually such as food, price, atmosphere, and service for a restaurant domain. While in the learned aspects type, some methods are used to extract the aspects automatically from the users' reviews. Additionally, some researchers identity fix aspects at the beginning of their methods then search for other learned aspects that are related to the fixed aspects from the users' reviews such as in [63], [76]. Most of the researchers claim that the learned aspects give better recommendations compared to the fixed aspects [60], [69], [77].

Learned aspects are preferred than the fixed aspects due to the following reasons:

a. The number of fixed features (catalog features) are few. This, in turn, will restrict the range of estimating inter-item similarity at the recommendation time.
b. The static features in some domains are technical in nature; as a result, it is hard to know the significance of the feature similarities in practical terms. For example, a camera item in the product domain has the following features (resolution, sensor-type, and price) while picture quality and beautiful design are learned aspects that provide more details about the camera and make the item's similarity easier to find.
c. Learned aspects from the user's review show the user's preferences are more accurate compared to the static ones because the user will write in his reviews only the aspects that he or she is interested in which will make knowing user's preferences easier and more obvious.
d. Approaches that use static aspects sometimes fail to provide compatible recommendations about the user's preferences. For example, service and food are both fixed aspects for a specific restaurant, the user put a 5/5 rating for the restaurant's service and 2/5 for the food. When the RS gives a recommendation to this user, it will recommend restaurants that have a good service but in fact, the user does not care about the restaurant's service aspect and care only with the restaurant's food. Thus, the system is unable to propose a suitable recommendation for the user.
e. A high number of item's features produce better results in the recommendation scenarios because the item is much better described compared to using fixed features which are typically small in numbers.

V. MULTI-CRITERIA REVIEW-BASED RECOMMENDATION APPROACH

Multi-criteria review-based RS uses user reviews to extract the criteria that will be used in the recommendation process. These criteria are defined from the review elements explained in the previous section. These criteria can be used on their own or by combining with the actual users' ratings. The full cycle (stages) of the multi-criteria review-based RSs is summarized in Figure 5.

Multi-criteria review based RS is applied by many kinds of research; each research has an idea about combining the

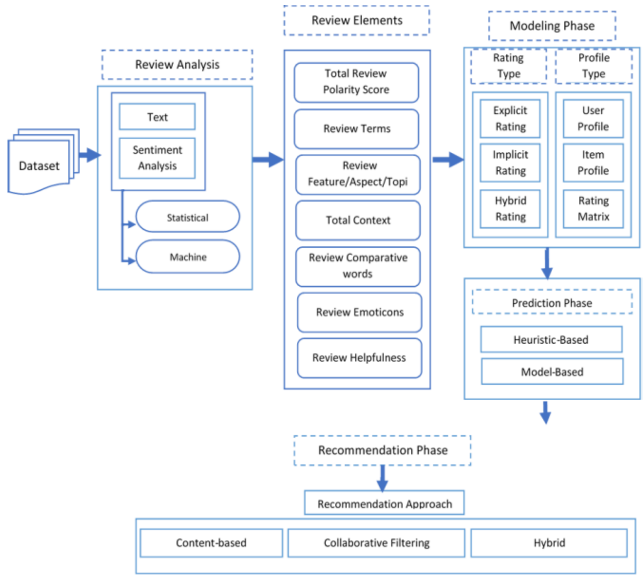


FIGURE 5. The full cycle of the multi-criteria review-based RSs.

elements of the user-generated review with the RS; some use just one review element while others combine more than one review element. Stated below are recent researches in multi-criteria review-based RS which are grouped based on the review elements as discussed in Section IV.

A. TOTAL REVIEW POLARITY SCORE

A user's general opinion can be inferred from his written review of an item. This overall sentiment can be converted into an implicit rating. Most of the works that use total review mainly involved the sentiment analysis approach whereby the total review polarity scores are generated by aggregating the score of all the opinion words in the reviews. However, there exist slight variations among the approaches that used total review polarity score that is mainly concerned with the variations of the selected opinion words within the reviews.

Pappas and Popescu-Belis [78] work focuses on addressing the problem of one-class CF. One-class CF problem referred to the CF approach that deals with nothing but positive explicit feedbacks.

The issue with the one-class CF problem is the identification of negative instances. Therefore, this approach extracts sentiment information from textual reviews, and it is integrated with the nearest neighbor model into a sentiment-aware nearest-neighbor model (SANN) by mapping the sentiment scores according to the user's ratings (likes or favorites). The proposed approach consists of two steps: firstly, the polarity of the user's reviews is calculated using a rule-based classifier [79]. Then the sum of the total polarities of each sentence is normalized. Secondly, the normal neighborhood model is extended by proposing a sentiment aware nearest neighbor approach using a mapping function (MF) to combine the user's ratings with the user polarities resulted from a rule-based classifier. Three

MF are defined, which are random mapping, fixed mapping and learned to map. To evaluate their application three real datasets are used which contain both user ratings and comments, namely Vimeo, TED, and Flicker, which are popular sources for videos, lectures, and images respectively. The proposed application is compared with five baseline models which are Top popular, Nearest Neighbors (NN), Singular Value Decomposition (SVD), Non-negative Matrix Factorization (NMF), and Sparse Non-negative Matrix Factorization (SNMF). Three performance measures are calculated which are the mean average precision (MAP), the mean average recall (MAR) and the mean average F-measure (MAF) using 5-fold cross-validation. The results show that the proposed approach outperforms all the baseline models which prove that there is an inherent relationship between user unary feedback (likes or favorites) and sentiment expressed in user comments.

García-Cumbreras *et al.* [15] approach exploits the pessimistic and optimistic behaviors among users of RSs. The idea is to classify users into two classes (Pessimist and Optimist) according to the average polarity of users' reviews then add the user's class as a new attribute to the CF algorithm. Five experiments are performed using RapidMiner to prove the effectiveness of the authors' idea as follows: the first experiment studies the relation between the user's rating and his reviews through calculating the user's rating from his reviews using SVM algorithm. The result shows an implicit relation between the user's rating and his reviews and it proves that the user's reviews provide valuable information that enhances RSs' performance. While in the second experiment, the rating prediction is calculated by feeding the rating from the users' reviews into the CF using the k Nearest Neighbor (kNN) algorithm for both user-based and item-based approach. The result of the user-based outperforms item-based approach. Thus, the authors try to enhance the rating prediction in a subsequent experiment by adding some characteristics for users by either using a rating behavior or sentiment analysis of the users' reviews. In the third experiment, the rating prediction is calculated using a new attribute called user classification. The classification is performed based on the average of all the movie ratings given by each user, whereby the pessimist class is for users with average ratings < 4 and optimist class is for users with average ratings > 6. The results show that using the user category in rating a prediction based on the rating only (not reviews) slightly reduce the error prediction values of the ratings. Finally, the last experiment is similar to the previous one except that the classification of the users is based on the average polarity of his reviews and not his ratings. The accuracy of the classification is 80% which proves that the user can be classified based on reviews only. Additionally, the rating predicting is calculated in this experiment by feeding the user class into the CF using kNN, and the results outperform the conventional CF algorithms which prove that users' reviews can enhance the RSs' performance. For performing the experiments, a new corpus is created from the Internet Movie Database (IMDb) using an automatic extraction program which retrieves the user rating and reviews for each movie.

TABLE 3. Summary for researches that use total review polarity score element.

Research	Main Contribution
Pappas & Popescu [78]	Deals with one-class CF problems.
García-Cumbreras et al. [15]	Takes into account the optimistic and pessimistic users' behaviors during ratings.
Zhang et al.[73]	Aggregates the opinion words with the emoticons to generate overall ratings.

Zhang *et al.* [73] proposes an algorithm to infer the overall rating (or virtual rating) from users' reviews by aggregating the sentiments of the opinion words with the emoticons that are also included in the reviews to mitigate the sparsity problem in RSs. The proposed algorithm consists of two main steps: the first step is a review sentiment classification using SElf-supervised, Lexicon and Corpus-based (SELC) model to derive the virtual rating, while the second step is item recommendation using user-based and item-based CF algorithms. The SELC model combines the unsupervised model with the semi-supervised model. Through it, the overall sentiment score of each review is calculated from the two sets which are the sentiment word element set and the emoticons set by aggregating the scores of the words and emoticons that occur in the target review. Experiments that compared among user-based, item-based and non-personalized popularity-based approaches use two datasets: Youku (a Chinese Website) that does not contain real ratings and Amazon.com (book section) that has real ratings. The results show that the user-based CF outperforms both the item-based CF and the non-personalized popularity-based approach in terms of precision. Experiment on top-N recommendation shows that the user-based CF that uses both real and virtual ratings performed the best in terms of precision. A unique feature of this approach is the combination of user textual reviews and emoticons, which exist in 41% of the users' reviews.

Table 3 summarizes the main contributions of the recommendation approaches that exploit total review polarity score elements.

B. REVIEW TERMS

Review terms are the words that frequently occur in a review. The use of review terms is mainly found in the works of D'Addio *et al.* [20], [60], and D'Addio and Manzato [7], [61], which primarily use users' reviews to produce item representation that is based on the overall sentiment regarding the items' features. The approach follows a four-step procedure: text pre-processing, feature extraction, item representation using sentiment analysis and recommendation.

The text preprocessing step aims to convert the unstructured user reviews into a structured form to extract features that can then be used to develop a vector-based representation for each item. The value of each vector's position represents the overall sentiment of a specific feature in all the reviews.

The feature extraction step is the main step and it is quite different from the four types of research conducted by D'Addio: At the beginning of their approach, the Transductive Learning for Automatic Term Extraction (TLATE) method proposed by Conrado *et al.* [80] is used for extract the features in the works of both D'Addio *et al.* [60] and D'Addio and Manzato [61]. Next, they develop two techniques for feature extraction term-based and aspect-based in the work of D'Addio and Manzato [7]. For the term-based technique, the candidate features are extracted if they are tagged as a singular or plural noun and their frequencies exceed the threshold value. While in the aspect-based technique the features are extracted after the process of stemming using porter algorithm [81], stop words removal and clustering. In the last work of D'Addio and Manzato [61], the feature extraction is made more precise through extracting terms and aspects using heuristic and machine learning.

In the item representation using the sentiment analysis step, the item vector is generated using the extracted feature as used in the previous step in which each position of the item vector is the score of a feature. The score is calculated using the Stanford CoreNLP proposed by Socher *et al.* [82]. In the work of D'Addio and Manzato [61], the score is calculated based on the feature popularity of all the users.

The last step is the recommendation step where item neighborhood-based CF is used. The produced items' vectors are used to discover the items' similarities instead of the items' rating vector and they are then fed into the item neighborhood-based CF model, and the items with the highest rating are recommended to the user.

An experiment is conducted to evaluate the proposed approach for each work; for the works of both D'Addio *et al.* [20] and D'Addio and Manzato [61], two databases are combined which are the MovieLens dataset and the Internet Movie Database (IMDb). The results show that the proposed approach has a better value in both prec@10 and MAP performance measures compared to the recommendations based only on structured metadata. For the work of D'Addio and Manzato [7], the proposed approach is tested on the MovieLens-100K database (ML-100k). The results show the term-based technique gives better accuracy compared to the aspect-based technique and the proposed approach in both techniques outperforms the baselines (the approaches that use structured metadata) in terms of Root Mean Square Error (RMSE). Finally, the proposed approach of D'Addio *et al.* [60] is tested on two databases which are the MovieLens-100K (ML-100k) and Movielens-2k (HetRec ML). The results outperform all the results obtained from the compared traditional structured metadata constructions used as baselines in term of RMSE. The feature extraction technique based on the terms using machine learning provides the best results since it gives a large set of features and this provides more details about the items.

C. REVIEW FEATURE/ASPECT/TOPIC

The review aspect can be defined as a concept that describes a topic for each item's domain, and it is restricted to exist in every item. Each aspect consists of a set of terms. Most approaches under this category employ algorithms for aspect extraction and subsequently identify terms and opinion words associated with each aspect. Sentiment analysis is then applied to identify the polarity of each aspect and scores or ratings that have been allocated to the aspects. Some of the works under this category are as follows.

An approach by **Musto *et al.* [69]** follows a two-step process: the first step is building a framework using a non-symmetric measure called Kullback-Leibler divergence to extract the aspects, and for each aspect, the sub-aspects are extracted using phrases and informativeness measures proposed by Tomokiyo and Hurst [83]. Subsequently, a sentiment score for each main aspect and its sub-aspects is assigned using two strategies: a model-based algorithm that utilizes deep learning method proposed by Socher *et al.* [82] and a lexicon-based algorithm proposed by Musto *et al.* [84] which is based on the AFINN wordlist created by Nielsen [85]. The second step used the extracted aspects to feed the multi-criteria user-based and item-based CF algorithms. The sentiment score that resulted from the first step is considered as a rating, and the similarity between two users (or items) is calculated using the multi-dimensional Euclidean distance [40]. Their experimental evaluation included three datasets Yelp, TripAdvisor and Amazon. The best performance is achieved from the user-based CF with 10 aspects except for the Amazon dataset with 50 aspects. The results of the proposed algorithm also outperform all the single-criterion recommendation algorithms and algorithms that are based on the matrix factorization in terms of mean average error (MAE).

Akhtar *et al.* [86] present a technique for analyzing hotel reviews and extracting valuable information from them to help service providers and customers. The technique is targeted at TripAdvisor website's users. Two types of information are crawled and extracted from the TripAdvisor: the review text and the metadata. Then, each review is classified into one of the predefined categories. These categories are aspects that frequently recur in the review data set. After that, the topic modeling technique Latent Dirichlet Allocation (LDA) is applied to reveal the hidden topics from the reviews. Finally, sentiment analysis is performed using SentiWordNet corpus to calculate the review's polarity by aggregating the positive and negative words in the review. The experiment is carried out for the Orchid Residency Hotel and 78 reviews are crawled. After implementing all the previous processes on the reviews, a summary for the reviews is given showing the most positive, negative and neutral reviews. However, no evaluation result is reported.

Bauman *et al.* [87] develop a recommendation method that recommends to a user the items with the most valuable aspects to enhance the user's experience with those items. The valuable aspects are identified using Sentiment Utility

Logistic Model method which consists of two parts, the first part is used for extracting aspect-sentiment pairs using opinion parser called double propagation proposed by Qiu *et al.* [88] for extracting aspects from user reviews and a sentiment lexicon created by Liu [89] to classify the aspect sentiment (i.e. positive, negative or neutral). The second part is used for predicting the overall rating of a review by combining all the sentiment values for all the extracted aspects in the user's review and identifies the influence of each aspect on the overall rating. After the aspect is identified and the overall ratings are estimated, users' and items' profiles are created and the recommendation process is completed as a classification problem (i.e. the rating is classified as 'like' if the estimated overall rating is 4 or 5, and 'dislike' for 1, 2 or 3). An experiment is done to evaluate the performance of the developed method on the Yelp dataset for the domains of a restaurant, hotel and beauty, and spa. The number of the extracted aspects for the three domains are 69, 42 and 45 respectively. The proposed method is compared with three baseline approaches as follows: the popular aspect approach, the most positive aspect approach and the most negative aspect approach. The results show the proposed method outperforms the baseline approaches in terms of the Precision@3 and Area Under Curve (AUC) [12].

Yang *et al.* [21] also proposed a similar approach whereby the technique consists of three main components, opinion mining, aspect weight computing, and overall rating inference. The opinion mining component is responsible for extracting the aspects and opinion words from users' reviews then it computes a rating for each extracted aspect. The aspect extraction is done using the double propagation method [88] which selects the relationship between the aspect terms and the opinion word of type Direct Dependency relationship described using dependency grammar created by Tesnière [90]. The aspect weight computing component uses a tensor factorization approach to compute the aspect weight which expresses the user's satisfaction about the aspect. The third component is the overall rating inference which uses the aspect rating (user opinion) of component one and aspect weight (user preferences) of component two to predict the overall rating for the item that is not rated by a user. Two datasets are used for the experiment evaluation which are the movies dataset collected from IMDb website and hotel dataset provided by Wang *et al.* [66] in which the user review is associated with a rating on seven fixed aspects. Two accuracy metrics MAE and RMSE are computed, and then the results are compared with two baseline models (MF that does not consider any text reviews and TF that extracts user's opinions not only aspects weights). The proposed framework's results outperform the baseline models with high accuracy for both datasets.

Dong *et al.* [77] develop an approach for CB that combines feature similarity and feature sentiment to recommend items with high priority that are similar and better than the items in the user's query. The approach consists of three steps, and the first step is extracting the product's features from user-generated reviews using shallow NLP and statistical methods proposed by Hu and Liu [91] and Justeson and Katz [92] respectively. The second step is identifying an opinion for each extracted feature using the opinion pattern method proposed by Moghaddam and Ester [93]. The third step is generating recommendations for the user depending on his query Q. This approach recommends items that are not only similar to Q but also have higher relative sentiment improvement by calculating the product's score. The top-N products with the highest score are recommended to the user. To evaluate the developed approach, data from Amazon.com is extracted for six product domains such as Phones, Tablets, and GPS, in which each product has at least ten reviews. Two measure qualities are used which are rating benefit metric and query product similarity. The former compares two sets of recommendations depending on their ratings, while the latter computes the average similarity between the query product and the given recommendations based on the extracted feature. The experiment results demonstrate significant benefits in the quality of the given recommendations of the developed approach compared to Amazon's recommendations.

Wang *et al.* [57] focus on solving new users' problems with partial preferences. New users usually relate to the cold start problem in RS. Thus, most RS will ask users to indicate their preferences in some aspects or attributes of the items. However, such preferences are usually incomplete due to the user's knowledge gap of the items. Thus, Wang *et al.* **[57]** use users' reviews at the aspect opinion levels of the items to predict the missing preferences. The approach extracts the feature opinions from users' reviews and maps it to the static item's attributes to predict the user's incomplete preferences [63], [65]. The sentiment polarity of each opinionated feature is calculated using SentiWordNet [94], and this is subsequently mapped to the static items' attributes. Incomplete preferences of the new users are then inferred by calculating the similarities between the new user and the like-minded reviewers' preferences. The recommendation is based on the new user's preferences for the top-N items. The proposed approach is evaluated on a dataset collected from Amazon, containing 57 users (full preferences are determined), 64 products (digital cameras), each product has eight static attributes and 4904 reviews. To simulate the missing preferences of a new user, the partial preferences are selected at random (i.e. 2, 4, or 6 of his attribute preferences). The proposed approach achieves better recommendations accuracy compared to the four baselines used during evaluations: random, PopRank, PartialRank, and HybridRank.

Musat *et al.* [25] develop a method called topic profile collaborative filtering (TPCF) to address the problems of data sparsity and non-personalized ranking methods. TPCF works as follows: a frequency-based technique is utilized to extract the topics and this is followed by grouping them based on their synonyms using Wordnet synsets. Then, for each extracted topic, the relevant opinion word and its polarity are identified through constructing a set of relations such as the work of DeMarneffe *et al.* [95] and using the

OpinionFinder proposed by Wilson *et al.* [96]. Finally, based on the extracted topics and the scores generated from the polarities of the opinion words, the profile of the user topic is created. To recommend a product to the user, a product's score is calculated using the generated user profile, and the highest product's score is recommended to the user. The method is evaluated using a dataset collected from the TripAdvisor's website and its result outperforms the baseline method; the non-personalized product ranking method in terms of MAE and Kendall's tau rank correlation coefficient [97]

The work of **Chen and Chen [70]** attempts to address the issue of context in order to enhance personalized recommendation. They suggest that people may possess distinct aspect-level preferences in various contexts. An algorithm for contextual recommendation by extracting the relationship between the weight of each user's preferences and the related context is proposed. Both user's preferences and contextual information are extracted from the user's reviews. Two types of preferences are detected from the user's reviews, context-independent preference, and context-dependent preference. Both preferences are then combined to generate accurate recommendations. The former preferences are not affected by context and reflect the individual user's requirements for items that do not change over time. It is learned from the user's overall ratings and aspect opinions. The latter refers to the aspect-level requirement under certain context for a user. Contextual opinions are extracted using a rule-based approach and keyword matching. They experiment with mutual information, chi-square statistics, and information gain measures when assigning weights for various aspects in different contexts. Finally, both context-independent and dependent preferences are combined to compute an item's matching score, and the highest top-N scores are recommended to the user. The proposed algorithm has been tested on two datasets, TripAdvisor and Yelp which are restaurant datasets. They compare their algorithm with some baseline methods such as Context Freer and Context Pre-filter; the result of the proposed algorithm outperforms all the compared methods in terms of the Hit Ratio and Mean Reciprocal Rank. Additionally, the chi-square statistic method generates the best results compared to the other two contextual weighting methods.

Jamroonsilp and Prompoon [55] present an approach for item ranking based on the analysis of the user's reviews. Five pre-defined aspects for software items are defined, and the software ranking is calculated by analyzing the users' comparative sentences from the user's reviews for each software aspect. It consists of three phases, gathering user reviews, analyzing the gathered reviews and calculating software ranking. In the first phase, users' reviews are collected from google custom search API for three software topics which include database management system, PHP web application framework, and content management system. While in the second phase, the quality term (aspect) mentioned in the user's review is classified as one of the five pre-defined aspects based on the classes given by Coallier [98] and Mairiza *et al.* [99]; in addition a score for the quality term is assigned using the lexicon created by Hu and Liu [91]. This is followed by the extraction of the comparative relation, and a polarity score is assigned for the relation of the two types of software and the compared quality term mentioned in the user's reviews. Finally, in the third phase, the overall software score is calculated based on all the quality aspects scores and the relation's score that are calculated in the previous phase. The approach is evaluated using the dataset that is collected in the first phase using Pearson's correlation coefficient and compared with a human expert and the work of Zhang *et al.* [71]. It achieves a high Pearson's correlation coefficient with value 0.935 which proves that the software ranking is statistically consistent with the human experts' rankings and better than Zhang's approach.

TABLE 4. Summary of Researches that use feature/aspect/topic Element.

Research	Main Contribution
Musto et al. [69]	Feeds the CF with the aspect and the sub-aspects.
Akhtar et al. [86]	Uses the aspects as categories and classifies users based on the categories.
Bauman et al. [87]	Identifies the most valuable aspects and recommends items to user based on them.
Yang et al. [21]	Integrates users' preferences and opinions on different aspects into the CF algorithm.
Dong et al. [77]	Combines feature similarity with feature sentiment to recommend the high prioritized items that are similar and better than the items in the user's query.
Wang et al. [57]	Solves the new user problem with partial preferences through using users' reviews at the aspect opinion levels of the items to predict the missing preferences.
Musat et al. [25]	Creates user's profile based on the extracted topics and using them to calculate the product's score to be used in the recommendation process.
Chen & Chen [70]	Combines between two review elements (aspect and context) to enhance recommendations.
Jamroonsilp & Prompoon [55]	Analyzes the users' comparative sentences from user reviews for each software aspects to calculate the software ranking.
Zhang et al. [37]	Extracts aspects from user reviews and assigns weights for them by using the helpfulness element to develop user and item model.

Zhang *et al.* [37] propose an approach that exploits the aspect-level sentiment of the users' reviews with the support of helpfulness reviews. The approach consists of four phases. The first phase is extracting aspects using a latent Dirichlet allocation model and the words with the highest conditional probability are chosen as aspects. The second phase is determining the sentiment orientation of each extracted aspect using the sentiment lexicon SentiWordNet. By using the extracted aspects and their sentiment orientation, the item model and user model are created in Phase 3. The item model is represented as a vector with the mentioned aspects that appear in the product's reviews with the support of the helpfulness reviews to give weight to the related aspects. The user model is represented as a vector with the aspects that frequently occur in the user's reviews. The last phase is the recommendation phase, a score for each user and a candidate item pair is calculated by multiplying both user's vector and item vector, and the items with the top k scores are recommended to the user. An experiment conducted on Yelp dataset (i.e. restaurant domain) evaluates the proposed approach. The approach is compared with two baseline methods (CF based on matrix factorization approach and popularity-based approach), and its result outperforms the two baseline methods in terms of mean reciprocal rank (MRP).

Table 4 provides a summary of all the approaches previously discussed that use aspects elements in the review-based recommendation.

Following Table 5 that provides a summary of all the 28 surveyed approaches categorized as multi-criteria review-based recommender system.

VI. DISCUSSION

The current state in RS research is concerned with the inability of such systems in providing an accurate recommendation to users. One of these inabilities relates to the systems focusing only on single criterion recommendation. To enhance the RS performance, there is a need to provide more information about the users and items and make the decision regarding the recommendation process based on multi-criteria or multi-alternatives that show users' preferences not only based on a single-criterion rating. The importance of such multi-criteria recommender systems is that they build their criteria from the users' reviews to enhance the accuracy of the RSs performance and is the primary reason for conducting this survey.

Relying only on the overall ratings in the recommendation process may result in inaccurate recommendations because these ratings cannot reflect the users' preferences accurately. This survey explores how the user-generated reviews can overcome such problems by utilizing them as an alternative and valuable source in the recommendation process through merging them with MCRS to define the recommendation criteria to enhance the accuracy of the RS's performance. Additionally, the elements that can be extracted from the user's reviews are discussed in detail in this survey, and the recent research by various researchers who extract the criteria from users' reviews which have been discussed and categorized based on the elements used. This is followed by a table that contains 28 recent studies in the field of multi-criteria review-based recommender systems. The published papers are collected from reliable resources such as ACM, Springer, and ScienceDirect and Figure 6 illustrates the statistics of the selected papers by the publisher while Figure 7 illustrates the statistics of the included papers according to the publication year. All the chosen papers are published in the last 8 years and the reference Aciar et al. [16] is added because their research is the first attempt to use user reviews in building RS.

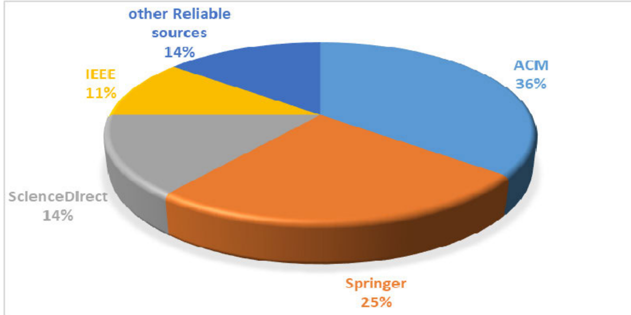


FIGURE 6. Statistics of the surveyed papers according to publishers.

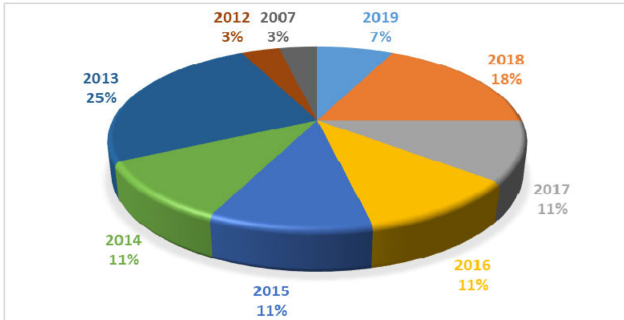


FIGURE 7. Percentage of the surveyed papers by publication year.

Additionally, the included papers have been cited by others which make them more reliable. Figure 8 shows the citation number for each reference collected from the Google Scholar.

Follows are some analyzing points that summarize Table 5:

- **RS approach**

Most of the research that used multi-criteria review-based RS type applied the CF approach because finding the user's preferences is the major reason for using this type of RS and users' reviews are considered valuable resources for establishing user preferences' criteria. On the other hand, CB does not use users' details which makes it less preferable compared to this type. So, CF is the most suitable approach for multi-criteria review-based RS type.

- **Review Analysis Method**

As mentioned previously, this review is focused on the sentiment analysis method for review processing. Based on Table 5 it is clear most of the researches utilize the statistical methods for analyzing the user reviews and extracting their elements. One of the reasons for using the statistical methods

TABLE 5. Multi-Criteria Review-Based recommender systems.

Research	Approach				Review Analysis Method Sentiment Analysis Technique			Review Elements	Rating Type		Profile Type			Recommendation using		Evaluation of Data Set		Size				Performance measure
	Content Based	Collaborative Filtering	Preference Based	Hybrid	Statistical	Machine Learning	Lexicon / Dictionary for Sentiment		Explicit	Implicit	Rating matrix	User	Item	Overall rating	Preferences rating	Data Set	Domain	Item	Users	Rating	Reviews	
(Rafailidis & Crestani 2019) [100]		•				•	Not used	Total Review polarity scores		•	•			•		Amazon	Beauty	For all the dataset Domains				MSE= 1.311
																	Health					1.203
																	Cell Phone					1.382
																	Clothing	N/A	N/A	10754316	823534	1.163
(Cheng et al. 2019) [101]		•				•	Not used	Aspect		•	•	•	•	•	•	Yelp	Local business	1416	7109	N/A	135015	RMSE = 0.9562
																Amazon	Beauty	12101	22363	N/A	198502	1.0613
																	CDs	64421	75258	N/A	1097597	0.9250
																	Phone	10429	27879	N/A	19439	1.1456
																	Clothing	23033	39387	N/A	278677	1.0118
																	Movies	50052	123960	N/A	1697533	1.0091
(Pappas and Popescu-Belis 2016) [78]		•				•	MPQA polarity lexicon [2]	Total Review polarity scores	•	•	•					TED	Talks	1203	4961	113241	35229	MAP = 6.10 MAR= 22.73 MAF = 9.63
																Vimeo	Videos	2000	7071	155207	32639	MAP = 5.48 MAR= 16.92 MAF = 8.28
																Flicker	Images	1994	9963	161398	304564	MAP= 15.26 MAR= 56.75 MAF = 24.05
(Musto et al. 2017) [69]		•			•		Stanford CoreNLP[3] & a lexicon based on AFINN wordlist [4]	Learned Aspect		•	•			•	•	Yelp	restaurant	45981	11537	N/A	229906	#asp=10 MAE= 0.8362
																TripAdvisor	hotel	536952	3945	N/A	796958	#asp=50 MAE= 0.7111
																Amazon	product	826773	50210	N/A	1324759	#asp=10 MAE= 0.6276
(Yang et al. 2016) [21]		•				•	Seed opinion lexicon [5]	Learned Aspect	•	•	•			•	•	IMDb	Movies	1507	879	N/A	41128	MAE= 1.204 RMSE=1.647
																[66] dataset	Hotel	1850	64215	N/A	81085	#asp =8 MAE= 1.134 RMSE=1.468
(Cheng et al. 2018) [38]		•			•		Not Used	Learned Aspects	•	•	•	•	•	•	•	19 DataSet from Amazon and Yelp	Product & restaurant	For Yelp only 63300	169257	1659678	N/A	K=5 RMSE= 1.155
(Wang et al. 2013) [57]			•		•		SentiWordNet [6]	Learned aspects		•		•			•	Amazon	Products [digital camera]	64	57	N/A	4909	Hit ratio = 0.2807 ± 0.0372 With given 2 attributes
(Musat et al. 2013) [25]		•			•		OpinionFinder [96]	Learned Topics		•		•			•	TripAdvisor	Hotel	216	59067	N/A	68049	reduce the MAE error by 8% when no of reviews = 15
(Zhang et al. 2013) [73]		•				•	HowNet Sentiment Dictionary (Chinese dictionary)	-Total Review polarity Scores - Emoticons		•	•			•		Youku	movies	1085	6450	N/A	108137	K=10 precision U-U =5.1% I-I =4.7%
									•	•						Amazon	book	1805	5502	N/A	318730	K=10 precision U-U =5.6% K=20 precision I-I =5.2%
(Bauman et al. 2017) [87]				•	•		sentiment lexicon [89]	Learned aspects		•		•	•	•	•	Yelp	restaurant	N/A	23209	N/A	602112	Precision@Top3 =0.818 AUC=0.707
																Yelp	Hotel	N/A	352	N/A	5669	Precision@Top3 =0.849 AUC=0.745
																Yelp	Beauty &Spa	N/A	349	N/A	5065	Precision@Top3 =0.862 AUC=0.663

TABLE 5. *(Continued.)* Multi-Criteria Review-Based recommender systems.

Research	Approach				Review Analysis Method Sentiment Analysis Technique			Review Elements	Rating Type		Profile Type			Recommendation using		Evaluation of Data Set						Performance measure
																		Size				
	Content Based	Collaborative Filtering	Preference Based	Hybrid	Statistical	Machine Learning	Lexicon / Dictionary for Sentiment		Explicit	Implicit	Rating matrix	User	Item	Overall rating	Preferences rating	Data Set	Domain	Item	Users	Rating	Reviews	
(Jamroonsilp & Prompoon 2013) [55]			•		•		SentiWordNet 6	-Fixed aspects -comparative words		•			•	•		Google	Software	105	N/A	N/A	3542	Pearson's correlation coefficient = 0.935
(D'Addi et al. 2017) [60]		•				•	Stanford CoreNLP [102]	-Learned Aspects Or -Terms	•	•			•	•		Movie Lens-100K & IMDb	Movies	1682	943	100000	15863	Terms with k=20 RMSE=0.9302 Prec@10 =0.1043 MAP =0.0658
					•																	RMSE=0.9310 Prec@10 =0.1041 MAP =0.0656
		•				•	Stanford CoreNLP [102]	-Learned Aspects Or -Terms	•	•			•	•		Movie Lens-2K (HetRec ML) & IMDb	Movies	10197	2113	855598	656031	RMSE=0.7964 Prec@10 =0.1047 MAP =0.0258
					•																	RMSE= 0.8025 Prec@10 =0.1057 MAP = 0.0256
(Zhang et al. 2015) [37]			•			•	SentiWordNet 6	-Learned aspects -helpfulness		•		•	•	•	•	Yelp	restaurant	N/A	30000	N/A	60000	MRP=0.627
(D'Addio & Manzato 2015) [7]		•			•		Stanford CoreNLP sentiment analysis tool	-Learned Aspects or -Terms		•			•	•		Movie Lens-100K & IMDb	Movies	1682	943	100000	15863	Using term CF RMSE= 0.93106
(Ebadi & Krzyzak 2016) [3]				•		•	Not Used Polarity score is predicted	learned Aspects The detected keywords list is manually refined	•	•	•			•	•	TripAdvisor	Hotel	4333	148429	N/A	878561	MSE=1.22 MAE=0.78 RMSE=1.1
(Wang & Chen 2012) [63]			•		•		SentiWordNet	Fix Feature heuristic		•		•		•	•	www.buzzillions.com	digital camera	186	3754	N/A	7485	H@10=0.229 H@20=0.245 H@30=0.409 Percentile=0.697
(Chen &Chen 2015) [64]			•		•		opinion lexicon[79]	-Fix Context -Fix Aspect heuristic		•	•			•	•	*Yelp*	Restaurant	11485	23152	N/A	237077	H@10 =0.1378 H@15= 0.1601 MRR@10=0.046 MRR@15= 0.062
																TripAdvisor	Hotel	11405	30039	N/A	357113	H@10 =0.1184 H@15= 0.1457 MRR@10=0.052 MRR@15= 0.059
(D'Addio et al. 2014) [20]		•				•	Stanford CoreNLP sentiment analysis tool	Terms		•			•	•		Movie Lens-100K & IMDb	Movies	1682	943	100000	15851	K=57 prec@10= 0,05991 MAP= 0,04764
(Dong et al. 2013) [62]			•		•		opinion pattern method [93]	Learned feature		•			•	•		Amazon	Electronic Product	1000	599	N/A	42482	Mean relative rank achieve a rank improvement of only about 4% (Jaccard) and 9% for Cosine
(D'Addio & Manzato 2014) [61]		•			•		Stanford CoreNLP sentiment analysis tool	Terms		•			•	•		Movie Lens-100K & IMDb	Movies	1682	943	100000	15851	Threshold 30 prec@10= 0.04623 MAP= 0.03684
(García-Cumbreras et al. 2013) [15]		•				•	WeFeelFine1 [103]	Total Review polarity scores		•	•			•		IMDb	Movies	2713	4112	N/A	80848	**RMSE**= 2.14 **MAE** = 1.63

TABLE 5. *(Continued.)* **Multi-Criteria Review-Based recommender systems.**

Research	Approach				Review Analysis Method Sentiment Analysis Technique			Review Elements	Rating Type		Profile Type			Recommendation using		Evaluation of Data Set						Performance measure
	Content Based	Collaborative Filtering	Preferenece Based	Hybrid	Statistical	Machine Learning	Lexicon / Dictionary for Sentiment		Explicit	Implicit	Rating matrix	User	Item	Overall rating	Preferences rating	Data Set	Domain	Size: Item	Size: Users	Size: Rating	Size: Reviews	
(Wang et al. 2018) [6]				•	•		lexicon is built for movies	Terms	•	•		•	•	•	•	Douban movie (https://movie.douban.com/)	Movies	12253	205754	N/A	617985 7	Precision= 0.782 F1= 0.771
(Siering et al. 2018) [76]			•		•		Harvard General Inquirer Lexicon [104]	Fix aspect + learned Aspect	•	•			•	•	•	Three random samplings of airline review airlinequality.com	Airline	195	N/A	N/A	3000	Using bag-of-words and SVM Accuracy= 80.80 F1= 80.91
(Sun et al. 2018) [51]		•			•		HowNet lexicon	Fix aspect		•		•		•	•	commercial website http://t.dianping.com/tianjin	restaurant	2903	9171	N/A	56011 3	Precision= 0.023 F1=0.025
																	Hotel	608	9563	N/A	31961	Precision= 0.012 F1=0.01
(Zheng *et al. 2018*) [105]		•			•		Chinese sentiment vocabulary [106]	Feature		•	•			•		Tongcheng, (Chinese tourist social network)	Tourism	5722	31289 6	N/A	98568 3	**MAE**= 1.041 **RMSE**= 0.913
(Aciar et al. 2007) [16]	•					•	Rule Induction Kit for Text	Fix feature		•		•			•	Digital Photography Review Web site	Camera	N/A	N/A	N/A	N/A	N/A
(Dong et al. 2013) [77]			•		•		Sentiment Lexicon [91]	Learned feature		•			•	•		Amazon	Tablets -26	166	N/A	N/A	17936	Rating benefit metric (range btw 15 and 21%) higher than Amazon
																Amazon	Phones -9	257	N/A	N/A	14860	
																Amazon	GPS-24	119	N/A	N/A	12115	
(Chen and Chen 2014) [70]			•		•		opinion lexicon[79]	-fix Context -Fix Aspects heuristic		•	•			•	•	TripAdvisor	restaurant	15315	6203	N/A	12193 2	Using CHI **H@10**=0.171 7 **MRR@10**= 0.0528
																Yelp	restaurant	10581	3969	N/A	12528 6	Using CHI **H@10=0**.1559 **MRR@10**= 0.0588

1 WeFeelFine is constantly collecting affective sentences from different blogs and other social media sources. These sentences are labeled with a ''feeling.'' Available at http://wefeelfine.org/api.html.
2 http://mpqa.cs.pitt.edu/.
3 a model-based sentiment analysis algorithm included in Stanford CoreNLP3 which uses deep learning techniques
4 AFINN wordlist contains about 2500 English words that have been manually tagged with a score that can range from positive (+5) to negative (-5)
5 http://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html.
6 SentiWordnet is a lexical resource which WordNet synset is associated with three numerical scores describing how objective, positive and negative the terms contained in the synset are.

PB is Preference-based product ranking approach which is a new approach can be added to the RSs approaches Chen [9] suggested. This type of approach is used when the item can be characterized as set of features/attributes (e.g., food, atmosphere, the price for a restaurant) while the user preferences on each of the item's attribute are calculated as a value or a weight that shows his interested to this attribute. The recommendation for an item is performed as a score (i.e. item ranking) which measured based on the weights that placed on each of the user's attributes [9]. PB considers as a part of CF because it uses users' vectors for calculating the predicting rating for the recommended list.

is there are no labeled datasets that suit the sentiment analysis processing. Multicriteria review-based RS deals with a considerable number of reviews and it is a challenging task to label a huge number of reviews and determine the sentiment words for each review.

- **Sentiment Lexicons**

Most of the researchers use existing lexicons, and the most prominent two are the SentiWordNet and Stanford-CoreNLP sentiment analysis tools. Accurately assigning sentiment polarity for each sentiment word is a difficult task and it is also significantly sensitive depending on the domains. Unfortunately, most of the lexicons are general and they are domain-independent lexicons. Thus, it may affect the accuracy of the sentiment analysis process. The work by Wang [6], however, used a domain-specific lexicon (movie domain) which is built based on his movie dataset and it shows high precision results compared to others.

- **Review Elements**

Surprisingly, only a few researchers have used more than one element but the most are only two. In spite of that, the research that implements two elements has enhanced the RS performance. Most of the approaches focus on aspect review element either the fixed or learned aspects. It can be deduced that extracting the aspects provides more information about the users. This will accordingly improve the process of recommending items to the users based on their preferences.

- **Rating Type**

We can observe that most of the researches use implicit ratings with the view that implicit ratings deliver more legitimate opinions of the users instead of the explicit ratings.

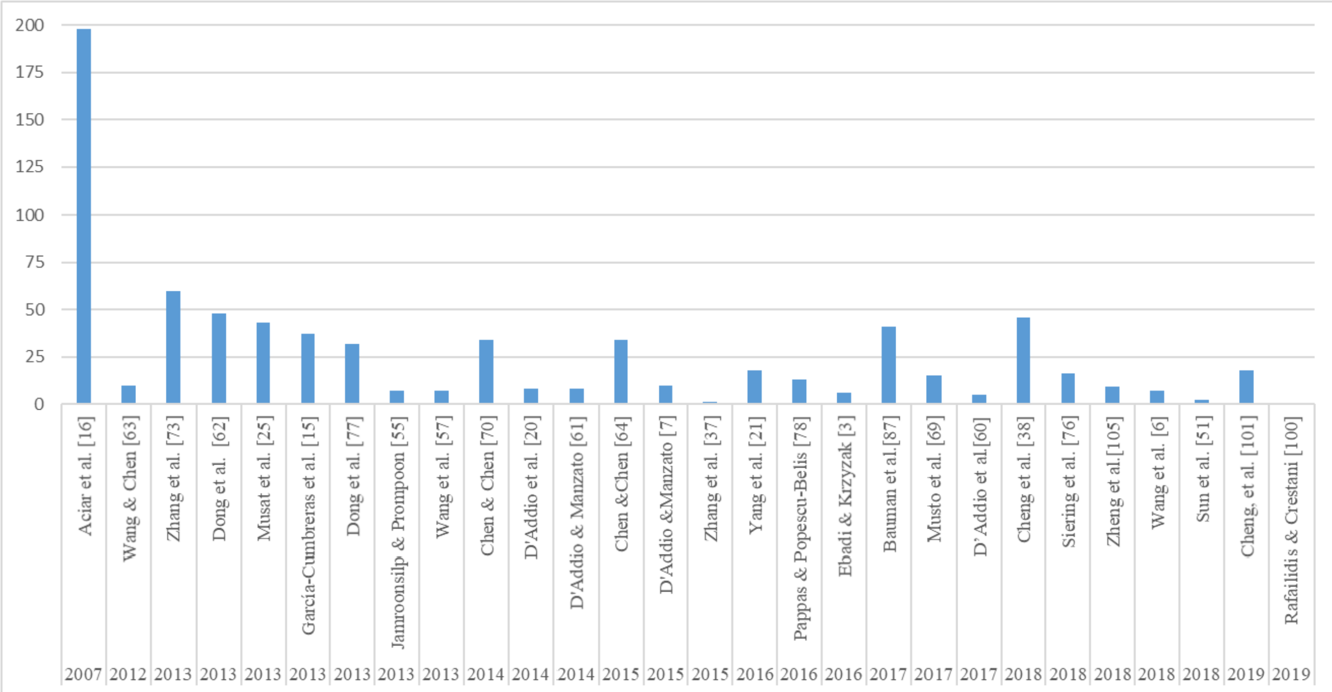


FIGURE 8. Number of citations for each reference included in table 5.

- **Profile Type**

The use of profile types largely depends on the type of RSs. The CF recommender systems mainly require a rating matrix, whereas the CB recommender systems exploit the user and item profiles. However, the distinction between these two types of recommender systems is becoming less crucial as some RSs exploit both types of profiles.

- **Recommendation using the Overall/Preferences rating**

Most of the researches obtain benefits from the user reviews element. Thus, they provide the recommendations based on the overall rating from the elements of the review or make it more precise based on specific user preferences.

- **Evaluation of Data Set**

The data sets used for evaluating the RSs are mainly from the Amazon, TripAdvisor, and Yelp which involve domains such as products, movies, hotels, and restaurants.

- **Performance Measures**

There are many measures to evaluate the performance of RS. This is summarized in Figure 9.

Error Metric is the performance measure that is used by most of the research then in comparison with Top-N metrics. Although there is remarkable progress in multi-criteria review-based recommender systems in recent years, further studies are needed.

One of the future trends that can be explored is combining various types of elements of the reviews [9]. Most of the current research works that exploit user reviews in multi-criteria recommendations only consider one or two

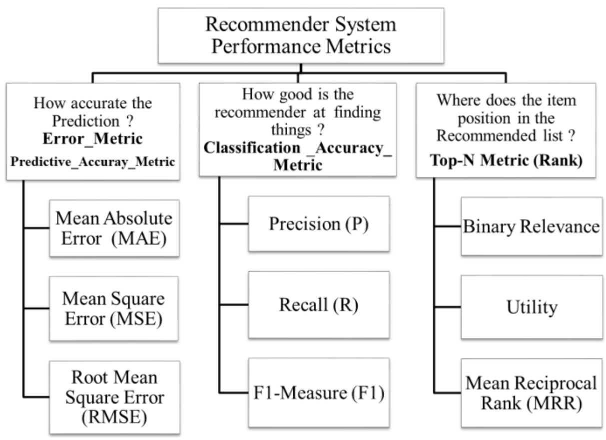


FIGURE 9. Recommender system performance metrics.

elements as shown in Table 6. All the studies that use two-element combinations from reviews prove that the accuracy of the RS is enhanced compared to other baselines. As a result, there is a need for developing multi-criteria RS that explores more element combinations to improve the accuracy of RS [9].

Another future trend is precise profiling for users and items from the extracted elements of the user's reviews. The recommendation process that generates recommendations based on user's preferences makes the RS a tool of personalization technologies in which the effectiveness of such technologies is based on the accuracy and completeness of the user's profiles [8]. As a result, precise profiling for the user and

TABLE 6. Multi-Criteria Review-Based Recommender Systems with the review elements that are used.

References	Elements						
	Total Review Polarity Score	Review Terms	Review Feature/Aspect / Topic	Review Context	Review Comparative Words	Review Emoticons	Review Helpfulness
Sun et al. [51]			•				
Siering et al. [76]			•				
Cheng et al. [38]			•				
Musto et al. [69]			•				
Akhtar et al. [86]			•				
Bauman et al. [87]			•				
D'Addio et al. [60]		•	•				
Yang et al. [21]			•				
Pappas & Popescu-Belis [78]	•						
Zhang et al. [37]			•				•
D'Addio & Manzato [7]		•	•				
Chen & Chen [70]			•	•			
D'Addio & Manzato [61]		•					
D'Addio et al. [20]		•					
García-Cumbreras et al. [15]	•						
Zhang et al. [73]	•						
Dong et al. [77]			•			•	
Wang et al. [57]			•				
Musat et al. [25]		•					
Jamroonsilp & Prompoon [55]			•		•		

items is important to gain accurate recommendations. Despite that, there is a lack of studies that aim to use and organize the review elements to develop or enhance user and item profiles [7]. Most of the available studies aim to use the "total review polarity score" element such as [78], [15] or use aspect (i.e. a single element) to develop user profile or/and item profile such as [38], [60], [87]. As a result, there is a need to use a combination of the review elements to develop both item and user profiles to enhance the accuracy of the recommendation performance.

Another point of the forthcoming trends is enhancing the evaluation process of the existing multi-criteria review-based RSs because it includes some limitations, especially with the type of baselines that are compared with. For example, the CB approach that use reviews to build a user's profile compared their approach with the profiles that are built from static item description [56] and not with other profiles that are developed using reviews and the approaches that use reviews to calculate item rank compared to their approach with popularity-based approach, not with the approaches that use standard preferences ranking [9], [55].

Finally, the abundant information gained from the users' reviews can be used to produce explanations for users during the recommendation process. This explanation will increase the user's trust in using the recommendation system because it declares and explains why these recommendations are recommended to him. On the other hand, there is a lack of research that works at this point.

VII. CONCLUSION

Currently, user-generated reviews are used to improve the accuracy of the RSs performance by using text analysis and sentiment analysis to transform the unstructured user reviews into a structured form that can be merged with RSs. Many elements can be extracted from the user's reviews then delivering them to the RSs approaches to solve the problem of inaccurate recommendations caused by relying only on the overall ratings in the recommendation process. This survey concerns the MCRSs that extract their criteria from users' reviews due to the apparent improvement implemented to the RSs performance when applying them. Users' reviews elements are discussed in detail. The approaches that implement these elements in the RSs are then explained and grouped based on the review elements used in developing their systems. After that, the most recent researches in multi-criteria review based recommender systems are presented in a table that explained the main points used. Finally, some of the future trends are discussed as challenges or open problems for this type of RSs.

We expect this survey will help researchers to gain more understanding about the multi-criteria review based recommender system and encourage them to explore the implicit values of the reviews and utilize them in future studies.

REFERENCES

[1] L. Liu, N. Mehandjiev, and D.-L. Xu, "Multi-criteria service recommendation based on user criteria preferences," in *Proc. 5th ACM Conf. Recommender Syst.*, 2011, pp. 77–84.
[2] F. Hdioud, B. Frikh, and B. Ouhbi, "Multi-criteria recommender systems based on multi-attribute decision making," in *Proc. Int. Conf. Inf. Integr. Web-Based Appl. Services*, 2013, p. 203.
[3] A. Ebadi and A. Krzyzak, "A hybrid multi-criteria hotel recommender system using explicit and implicit feedbacks," in *Proc. 18th Int. Conf. Appl. Sci. Inf. Syst. Technol. (ICASIST)*, Amsterdam, The Netherlands, 2016, pp. 1377–1385.
[4] N. R. Kermany and S. H. Alizadeh, "A hybrid multi-criteria recommender system using ontology and neuro-fuzzy techniques," *Electron. Commerce Res. Appl.*, vol. 21, pp. 50–64, Jan./Feb. 2017.
[5] F. Peña and R. Riffo, "Revisión, selección e implementación de un algoritmo de recomendación de material bibliográfico utilizando tecnología j2EE," Universidad del Bío-Bío, Consultado en, Concepción, Chile, Tech. Rep., 2008. [Online]. Available: http://repobib.ubiobio.cl/jspui/bitstream/123456789/2394/1/Riffo_Carrillo_Ricardo_Elias.pdf
[6] Y. Wang, M. Wang, and W. Xu, "A sentiment-enhanced hybrid recommender system for movie recommendation: A big data analytics framework," *Wireless Commun. Mobile Comput.*, vol. 2018, Mar. 2018, Art. no. 8263704.
[7] R. M. D'Addio and M. G. Manzato, "A sentiment-based item description approach for kNN collaborative filtering," in *Proc. 30th Annu. ACM Symp. Appl. Comput.*, 2015, pp. 1060–1065.
[8] K. Lakiotaki, N. F. Matsatsinis, and A. Tsoukias, "Multicriteria user modeling in recommender systems," *IEEE Intell. Syst.*, vol. 26, no. 2, pp. 64–76, Mar./Apr. 2011.
[9] L. Chen, G. Chen, and F. Wang, "Recommender systems based on user reviews: The state of the art," *User Model. User-Adapted Interact.*, vol. 25, pp. 99–154, 2015.
[10] G. Adomavicius and 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.*, vol. 17, no. 6, pp. 734–749, Jun. 2005.
[11] G. Adomavicius, N. Manouselis, and Y. Kwon, "Multi-criteria recommender systems," in *Recommender Systems Handbook*, F. Ricci, Ed. Boston, MA, USA: Springer, 2011, pp. 769–803.
[12] F. Ricci, L. Rokach, and B. Shapira, "Introduction to recommender systems handbook," in *Recommender Systems Handbook*. Boston, MA, USA: Springer, 2011, pp. 1–35.
[13] G. Adomavicius, R. Sankaranarayanan, S. Sen, and A. Tuzhilin, "Incorporating contextual information in recommender systems using a multidimensional approach," *ACM Trans. Inf. Syst.*, vol. 23, no. 1, pp. 103–145, 2005.
[14] M. Pazzani and D. Billsus, "Learning and revising user profiles: The identification of interesting Web sites," *Mach. Learn.*, vol. 27, no. 3, pp. 313–331, 1997.
[15] M. Á. García-Cumbreras, A. Montejo-Râez, and M. C. Díaz-Galiano, "Pessimists and optimists: Improving collaborative filtering through sentiment analysis," *Expert Syst. Appl.*, vol. 40, no. 17, pp. 6758–6765, 2013.
[16] S. Aciar, D. Zhang, S. Simoff, and J. Debenham, "Informed recommender: Basing recommendations on consumer product reviews," *IEEE Intell. Syst.*, vol. 22, no. 3, pp. 39–47, May/Jun. 2007.
[17] Y. Blanco-Fernandez, J. J. Pazos-arias, A. Gil-Solla, M. Ramos-Cabrer, and M. Lopez-Nores, "Providing entertainment by content-based filtering and semantic reasoning in intelligent recommender systems," *IEEE Trans. Consum. Electron.*, vol. 54, no. 2, pp. 727–735, May 2008.
[18] I. Cantador, M. Szomszor, H. Alani, M. Fernández, and P. Castells, "Enriching ontological user profiles with tagging history for multi-domain recommendations," in *Proc. CEUR Workshop*, 2008, pp. 1–16.
[19] M. J. Pazzani and D. Billsus, "Content-based recommendation systems," in *The Adaptive Web*. Berlin, Germany: Springer, 2007, pp. 325–341.
[20] R. D'Addio, M. Conrado, S. Resende, and M. Manzato, "Generating recommendations based on robust term extraction from users' reviews," in *Proc. 20th Brazilian Symp. Multimedia Web*, 2014, pp. 55–58.
[21] C. Yang, X. Yu, Y. Liu, Y. Nie, and Y. Wang, "Collaborative filtering with weighted opinion aspects," *Neurocomputing*, vol. 210, pp. 185–196, Oct. 2016.
[22] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl, "GroupLens: An open architecture for collaborative filtering of netnews," in *Proc. ACM Conf. Comput. Supported Cooperat. Work*, 1994, pp. 175–186.
[23] X. Su and T. M. Khoshgoftaar, "A survey of collaborative filtering techniques," *Adv. Artif. Intell.*, vol. 2009, Aug. 2009, Art. no. 421425.
[24] Y. Chen, C. Wu, M. Xie, and X. Guo, "Solving the sparsity problem in recommender systems using association retrieval," *J. Comput.*, vol. 6, no. 9, pp. 1896–1902, 2011.
[25] C. C. Musat, Y. Liang, and B. Faltings, "Recommendation using textual opinions," in *Proc. Int. Joint Conf. Artif. Intell.*, 2013, pp. 2684–2690.
[26] S. G. Esparza, M. P. O'Mahony, and B. Smyth, "Effective product recommendation using the real-time Web," in *Research and Development in Intelligent Systems XXVII*. London, U.K.: Springer, 2011, pp. 5–18.
[27] C. W. K. Leung, S. C. F. Chan, and F.-L. Chung, "Integrating collaborative filtering and sentiment analysis: A rating inference approach," in *Proc. ECAI Workshop Recommender Syst.*, 2006, pp. 62–66.
[28] A. I. Schein, A. Popescul, L. H. Ungar, and D. M. Pennock, "Methods and metrics for cold-start recommendations," in *Proc. 25th Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, 2002, pp. 253–260.
[29] Z. Huang, H. Chen, and D. Zeng, "Applying associative retrieval techniques to alleviate the sparsity problem in collaborative filtering," *ACM Trans. Inf. Syst.*, vol. 22, no. 1, pp. 116–142, 2015.
[30] Y. Xin, "Challenges in recommender systems: Scalability, privacy, and structured recommendations," Ph.D. dissertation, Dept. Elect. Eng. Comput. Sci., Massachusetts Inst. Technol., Cambridge, MA, USA, 2015.
[31] C. C. Aggarwal, *Recommender Systems*. Cham, Switzerland: Springer, 2016.
[32] R. Burke, "Knowledge-based recommender systems," in *Encyclopedia of Library and Information Sciences*, vol. 69. New York, NY, USA, 2000, pp. 175–186.
[33] T. Tran, "Combining collaborative filtering and knowledge-based approaches for better recommendation systems," *J. Bus. Technol.*, vol. 2, no. 2, pp. 17–24, 2007.
[34] V. Danilova and A. Ponomarev, "Hybrid recommender systems: The review of state-of-the-art research and applications," in *Proc. 20th Conf. Fruct Assoc.*, 2016.
[35] R. Burke, "Hybrid Web recommender systems," in *The Adaptive Web*. Berlin, Germany: Springer, 2007, pp. 377–408.
[36] A. Caputo, P. Basile, M. de Gemmis, P. Lops, G. Semeraro, and G. Rossiello, "SABRE: A sentiment aspect-based retrieval engine," in *Information Filtering and Retrieval*. Cham, Switzerland: Springer, 2017, pp. 63–78.
[37] Y. Zhang, R. Liu, and A. Li, "A novel approach to recommender system based on aspect-level sentiment analysis," in *Proc. 4th Nat. Conf. Electr., Electron. Comput. Eng. (NCEECE)*, 2015, pp. 1453–1458.
[38] Z. Cheng, Y. Ding, L. Zhu, and M. Kankanhalli, "Aspect-aware latent factor model: Rating prediction with ratings and reviews," in *Proc. World Wide Web Conf.*, 2018, pp. 639–648.

[39] G. Adomavicius and A. Tuzhilin, "Personalization technologies: A process-oriented perspective," *Commun. ACM*, vol. 48, no. 6, pp. 83–90, 2005.
[40] G. Adomavicius and Y. Kwon, "New recommendation techniques for multicriteria rating systems," *IEEE Intell. Syst.*, vol. 22, no. 3, pp. 48–55, May/Jun. 2007.
[41] K. Lakiotaki, S. Tsafarakis, and N. Matsatsinis, "UTA-Rec: A recommender system based on multiple criteria analysis," in *Proc. ACM Conf. Recommender Syst.*, 2008, pp. 219–226.
[42] Y. Zhang, Y. Zhuang, J. Wu, and L. Zhang, "Applying probabilistic latent semantic analysis to multi-criteria recommender system," *AI Commun.*, vol. 22, no. 2, pp. 97–107, 2009.
[43] M. Wasid and R. Ali, "An improved recommender system based on multi-criteria clustering approach," *Procedia Comput. Sci.*, vol. 131, pp. 93–101, Jan. 2018.
[44] Y. Zheng, "Utility-based multi-criteria recommender systems," in *Proc. 34th ACM/SIGAPP Symp. Appl. Comput.*, 2019, pp. 2529–2531.
[45] D. Jannach, M. Zanker, and M. Fuchs, "Leveraging multi-criteria customer feedback for satisfaction analysis and improved recommendations," *Inf. Technol. Tourism*, vol. 14, no. 2, pp. 119–149, 2014.
[46] Y. Zheng, "Situation-aware multi-criteria recommender system: Using criteria preferences as contexts," in *Proc. Symp. Appl. Comput.*, 2017, pp. 689–692.
[47] Y. Zheng, "Criteria chains: A novel multi-criteria recommendation approach," in *Proc. 22nd Int. Conf. Intell. Interfaces*, 2017, pp. 29–33.
[48] D. Tallapally, R. S. Sreepada, B. K. Patra, and K. S. Babu, "User preference learning in multi-criteria recommendations using stacked auto encoders," presented at the Proc. 12th ACM Conf. Recommender Syst., Vancouver, BC, Canada, 2018.
[49] F. Strub, R. Gaudel, and J. Mary, "Hybrid recommender system based on autoencoders," in *Proc. 1st Workshop Deep Learn. Recommender Syst.*, 2016, pp. 11–16.
[50] F. Cacheda, V. Carneiro, D. Fernández, and V. Formoso, "Comparison of collaborative filtering algorithms: Limitations of current techniques and proposals for scalable, high-performance recommender systems," *ACM Trans. Web*, vol. 5, no. 1, 2011, Art. no. 2.
[51] L. Sun, J. Guo, and Y. Zhu, "Applying uncertainty theory into the restaurant recommender system based on sentiment analysis of online Chinese reviews," *World Wide Web*, vol. 22, no. 1, pp. 83–100, 2018.
[52] B. Bickart and R. M. Schindler, "Internet forums as influential sources of consumer information," *J. Interact. Marketing*, vol. 15, no. 3, pp. 31–40, 2001.
[53] E. van Kleef, H. C. M. van Trijp, and P. Luning, "Consumer research in the early stages of new product development: A critical review of methods and techniques," *Food Qual. Preference*, vol. 16, no. 3, pp. 181–201, 2005.
[54] B. Pang and L. Lee, "Opinion mining and sentiment analysis," *Found. Trends Inf. Retr.*, vol. 2, nos. 1–2, pp. 1–135, 2008.
[55] S. Jamroonsilp and N. Prompoon, "Analyzing software reviews for software quality-based ranking," in *Proc. 10th Int. Conf. Elect. Eng./Electron., Comput., Telecommun. Inf. Technol. (ECTI-CON)*, May 2013, pp. 1–6.
[56] S. G. Esparza, M. P. O'Mahony, and B. Smyth, "A multi-criteria evaluation of a user generated content based recommender system," presented at the 3rd Workshop Recommender Syst. Social Web (RSWEB), 5th ACM Conf. Recommender Syst., Chicago, IL, USA, Oct. 2011.
[57] F. Wang, W. Pan, and L. Chen, "Recommendation for new users with partial preferences by integrating product reviews with static specifications," in *Proc. Int. Conf. Modeling, Adaptation, Pers.*, 2013, pp. 281–288.
[58] D. Poirier, I. Tellier, F. Fessant, and J. Schluth, "Towards text-based recommendations," in *Proc. Adaptivity, Pers. Fusion Heterogeneous Inf.*, 2010, pp. 136–137.
[59] Š. Pero and T. Horváth, "Opinion-driven matrix factorization for rating prediction," in *Proc. Int. Conf. Modeling, Adaptation, Pers.*, 2013, pp. 1–13.
[60] R. M. D'Addio, M. A. Domingues, and M. G. Manzato, "EExploiting feature extraction techniques on users' reviews for movies recommendation," *J. Brazilian Comput. Soc.*, vol. 23, Jun. 2017, Art. no. 7.
[61] R. M. D'Addio and M. G. Manzato, "A collaborative filtering approach based on user's reviews," in *Proc. Brazilian Conf. Intell. Syst. (BRACIS)*, Oct. 2014, pp. 204–209.
[62] R. Dong, M. Schaal, M. P. O'Mahony, K. McCarthy, and B. Smyth, "Opinionated product recommendation," in *Proc. Int. Conf. Case-Based Reasoning*, 2013, pp. 44–58.
[63] F. Wang and L. Chen, "Recommendation based on mining product reviewers preference similarity network," in *Proc. 6th SNAKDD Workshop*, 2012, p. 166.
[64] G. Chen and L. Chen, "Augmenting service recommender systems by incorporating contextual opinions from user reviews," *User Model. User-Adapted Interact.*, vol. 25, no. 3, pp. 295–329, 2015.
[65] F. Wang and L. Chen, "Recommending inexperienced products via learning from consumer reviews," in *Proc. IEEE/WIC/ACM Int. Conf. Web Intell. Intell. Agent Technol. (WI-IAT)*, Dec. 2012, pp. 596–603.
[66] H. Wang, Y. Lu, and C. Zhai, "Latent aspect rating analysis on review text data: A rating regression approach," in *Proc. 16th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2010, pp. 783–792.
[67] N. Jakob, S. H. Weber, M. C. Müller, and I. Gurevych, "Beyond the stars: Exploiting free-text user reviews to improve the accuracy of movie recommendations," in *Proc. 1st Int. CIKM Workshop Topic-Sentiment Anal. Mass Opinion*, 2009, pp. 57–64.
[68] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet allocation," *J. Mach. Learn. Res.*, vol. 3, pp. 993–1022, Mar. 2003.
[69] C. Musto, M. de Gemmis, G. Semeraro, and P. Lops, "A multi-criteria recommender system exploiting aspect-based sentiment analysis of users' reviews," presented at the Proc. 11th ACM Conf. Recommender Syst., Como, Italy, 2017.
[70] G. Chen and L. Chen, "Recommendation based on contextual opinions," in *Proc. Int. Conf. Modeling, Adaptation, Pers.*, 2014, pp. 61–73.
[71] K. Zhang, R. Narayanan, and A. N. Choudhary, "Voice of the customers: Mining online customer reviews for product feature-based ranking," *Proc. WOSN*, vol. 10, 2010, p. 11.
[72] M. Ganapathibhotla and B. Liu, "Mining opinions in comparative sentences," in *Proc. 22nd Int. Conf. Comput. Linguistics*, vol. 1, pp. 241–248, 2008.
[73] W. Zhang, G. Ding, L. Chen, C. Li, and C. Zhang, "Generating virtual ratings from chinese reviews to augment online recommendations," *ACM Trans. Intell. Syst. Technol.*, vol. 4, no. 1, 2013, Art. no. 9.
[74] Y. Moshfeghi, B. Piwowarski, and J. M. Jose, "Handling data sparsity in collaborative filtering using emotion and semantic based features," in *Proc. 34th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, 2011, pp. 625–634.
[75] S. Raghavan, S. Gunasekar, and J. Ghosh, "Review quality aware collaborative filtering," in *Proc. 6th ACM Conf. Recommender Syst.*, 2012, pp. 123–130.
[76] M. Siering, A. V. Deokar, and C. Janze, "Disentangling consumer recommendations: Explaining and predicting airline recommendations based on online reviews," *Decis. Support Syst.*, vol. 107, pp. 52–63, Mar. 2018.
[77] R. Dong, M. P. O'Mahony, M. Schaal, K. McCarthy, and B. Smyth, "Sentimental product recommendation," in *Proc. 7th ACM Conf. Recommender Syst.*, 2013, pp. 411–414.
[78] N. Pappas and A. Popescu-Belis, "Adaptive sentiment-aware one-class collaborative filtering," *Expert Syst. Appl.*, vol. 43, pp. 23–41, Jan. 2016.
[79] T. Wilson, J. Wiebe, and P. Hoffmann, "Recognizing contextual polarity in phrase-level sentiment analysis," presented at the Proc. Conf. Hum. Lang. Technol. Empirical Methods Natural Lang. Process., Vancouver, BC, Canada, 2005.
[80] M. S. Conrado, R. G. Rossi, T. A. S. Pardo, and S. O. Rezende, "Applying transductive learning for automatic term extraction: The case of the ecology domain," in *Proc. 2nd Int. Conf. Inform. Appl. (ICIA)*, Sep. 2013, pp. 264–269.
[81] H. Schütze, P. Raghavan, and C. D. Manning, *Introduction to Information Retrieval* vol. 39. Cambridge, U.K.: Cambridge Univ. Press, 2008.
[82] R. Socher, A. Perelygin, J. Wu, J. Chuang, C. D. Manning, A. Ng, and C. Potts, "Recursive deep models for semantic compositionality over a sentiment treebank," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2013, pp. 1631–1642.
[83] T. Tomokiyo and M. Hurst, "A language model approach to keyphrase extraction," presented at the Proc. ACL Workshop Multiword Expressions, Anal., Acquisition Treat., Sapporo, Japan, vol. 18, 2003.
[84] C. Musto, G. Semeraro, and M. Polignano, "A comparison of lexicon-based approaches for sentiment analysis of microblog posts," in *Proc. DART AI IA*, vol. 59, 2014, pp. 59–68.
[85] F. Å. Nielsen, "A new ANEW: Evaluation of a word list for sentiment analysis in microblogs," 2011, *arXiv:1103.2903*. [Online]. Available: https://arxiv.org/abs/1103.2903
[86] N. Akhtar, N. Zubair, A. Kumar, and T. Ahmad, "Aspect based sentiment oriented summarization of hotel reviews," *Procedia Comput. Sci.*, vol. 115, pp. 563–571, Jan. 2017.

[87] K. Bauman, B. Liu, and A. Tuzhilin, "Aspect based recommendations: Recommending items with the most valuable aspects based on user reviews," in *Proc. 23rd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2017, pp. 717–725.
[88] G. Qiu, B. Liu, J. Bu, and C. Chen, "Opinion word expansion and target extraction through double propagation," *Comput. Linguistics*, vol. 37, no. 1, pp. 9–27, 2011.
[89] B. Liu, *Sentiment Analysis: Mining Opinions, Sentiments, and Emotions*. Cambridge, U.K.: Cambridge Univ. Press, 2015.
[90] L. Tesniére, *Élements de Syntaxe Structurale*. Paris, France: Klincksieck, 1965.
[91] M. Hu and B. Liu, "Mining and summarizing customer reviews," in *Proc. 10th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2004, pp. 168–177.
[92] J. S. Justeson and S. M. Katz, "Technical terminology: Some linguistic properties and an algorithm for identification in text," *Natural Lang. Eng.*, vol. 1, no. 1, pp. 9–27, 1995.
[93] S. Moghaddam and M. Ester, "Opinion digger: An unsupervised opinion miner from unstructured product reviews," in *Proc. 19th ACM Int. Conf. Inf. Knowl. Manage.*, 2010, pp. 1825–1828.
[94] A. Esuli and F. Sebastiani, "SentiWordNet: A high-coverage lexical resource for opinion mining," *Evaluation*, vol. 17, pp. 1–26, 2007.
[95] M.-C. De Marneffe, B. MacCartney, and C. D. Manning, "Generating typed dependency parses from phrase structure parses," in *Proc. LREC*, 2006, pp. 449–454.
[96] T. Wilson, P. Hoffmann, S. Somasundaran, J. Kessler, J. Wiebe, Y. Choi, C. Cardie, E. Riloff, and S. Patwardhan, "OpinionFinder: A system for subjectivity analysis," in *Proc. HLT/EMNLP Interact. Demonstrations*, 2005, pp. 34–35.
[97] M. G. Kendall, "A new measure of rank correlation," *Biometrika*, vol. 30, pp. 81–93, Jun. 1938.
[98] F. Coallier, *Software Engineering—Product Quality—Part 1: Quality Model*, Standard ISO/IEC 9126-1, International Organization for Standardization, Geneva, Switzerland, 2001.
[99] D. Mairiza, D. Zowghi, and N. Nurmuliani, "An investigation into the notion of non-functional requirements," in *Proc. ACM Symp. Appl. Comput.*, 2010, pp. 311–317.
[100] D. Rafailidis and F. Crestani, "Adversarial training for review-based recommendations," in *Proc. 42nd Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, 2019, pp. 1057–1060.
[101] Z. Cheng, X. Chang, L. Zhu, R. C. Kanjirathinkal, and M. Kankanhalli, "MMALFM: Explainable recommendation by leveraging reviews and images," *ACM Trans. Inf. Syst.*, vol. 37, no. 2, 2019, Art. no. 16.
[102] C. Manning, M. Surdeanu, J. Bauer, J. Finkel, S. Bethard, and D. McClosky, "The Stanford CoreNLP natural language processing toolkit," in *Proc. 52nd Annu. Meeting Assoc. Comput. Linguistics, Syst. Demonstrations*, 2014, pp. 55–60.
[103] S. D. Kamvar and J. Harris, "We feel fine and searching the emotional Web," in *Proc. 4th ACM Int. Conf. Web Search Data Mining*, 2011, pp. 117–126.
[104] P. J. Stone, D. C. Dunphy, and M. S. Smith, "The general inquirer: A computer approach to content analysis," Tech. Rep., 1966. [Online]. Available: http://www.wjh.harvard.edu/ inquirer/homecat.htm
[105] X. Zheng, Y. Luo, L. Sun, J. Zhang, and F. Chen, "A tourism destination recommender system using users' sentiment and temporal dynamics," *J. Intell. Inf. Syst.*, vol. 51, no. 3, pp. 557–578, 2018.
[106] X. L. L. H. P. Yu and R. H. C. Jianmei, "Constructing the affective lexicon ontology," *J. China Soc. Sci. Tech. Inf.*, vol. 2, p. 6, 2008.

SUMAIA MOHAMMED AL-GHURIBI was born in Taiz, Yemen, in 1986. She received the B.S. degree in computer science from Taiz University, in 2008, and the M.S. degree in computer science from King Abdulaziz University, Jeddah, Saudi Arabia, in 2014. She is currently pursuing the Ph.D. degree with the Universiti Kebangsaan Malaysia (UKM), Malaysia. Her research interests include artificial intelligent, web mining, sentiment analysis, and recommender systems.

SHAHRUL AZMAN MOHD NOAH received the B.Sc. degree (Hons.) in mathematics from the Universiti Kebangsaan Malaysia, in 1992, and the M.Sc. and Ph.D. degrees in information studies from The University of Sheffield, U.K., in 1994 and 1998, respectively. He is currently a Professor with the Center for Artificial Intelligence Technology, Universiti Kebangsaan Malaysia, where he also heads the Knowledge Technology Research Group. His research interest includes information retrieval and ontology with special emphasis on semantic search and recommender systems. He has published more than 200 research articles in these areas. He is a member of International Association for Ontology and its Applications (IAOA) and of the IEEE Computer Science Society associations. He is currently the Chair of the Persatuan Capaian Maklumat dan Pengurusan Pengetahuan (PECAMP).