IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received July 12, 2019, accepted July 22, 2019, date of publication August 1, 2019, date of current version September 3, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2932500

SocialRec: A Context-Aware Recommendation Framework With Explicit Sentiment Analysis

RIZWANA IRFAN[1], OSMAN KHALID[1], MUHAMMAD USMAN SHAHID KHAN[1], FAISAL REHMAN[1], ATTA UR REHMAN KHAN[2], (Senior Member, IEEE), AND RAHEEL NAWAZ[3]

[1]Department of Computer Science, COMSATS University Islamabad, Islamabad 45550, Pakistan
[2]Faculty of Computing and Information Technology, Sohar University, Sohar 311, Oman
[3]Department of Operations, Technology, Events and Hospitality Management, Manchester Metropolitan University, Manchester M15 6BH, U.K.

Corresponding author: Muhammad Usman Shahid Khan (ushahid@ciit.net.pk)

ABSTRACT In recent years, recommendation systems have seen significant evolution in the field of knowledge engineering. Usually, the recommendation systems match users' preferences based on the star ratings provided by the users for various products. However, simply relying on users' ratings about an item can produce biased opinions, as a user's textual feedback may differ from the item rating provided by the user. In this paper, we propose SocialRec, a hybrid context-aware recommendation framework that utilizes a rating inference approach to incorporate users' textual reviews into traditional collaborative filtering methods for personalized recommendations of various items. We apply text-mining algorithms on a large-scale user-item feedback dataset to compute the sentiment scores. We propose a greedy heuristic to produce ranking of items based on users' social similarities and matching preferences. To address challenges resulting from cold start and data sparseness, SocialRec introduces pre-computation models based on Hub-Average (HA) inference. Rigorous evaluations of SocialRec (on large-scale datasets) demonstrate high accuracy, especially in comparison with previous related frameworks.

INDEX TERMS Text mining, recommendation system, collaborative filtering, Hub-Average inference.

I. INTRODUCTION

The advancement in web services and the tremendous increase in the online data has introduced new research problems in information retrieval. Web users are not only consuming but also contributing and disseminating information in a vastly decentralized manner via social networks, such as sharing reviews and personal interests [1]. The continuous accumulation of such massive Web contents consequently leads to the challenging problem of information search and retrieval. Recommender system has emerged as a promising research area to provide personalized recommendations from massively accumulated data [2]–[6]. Such systems apply machine learning, knowledge engineering, and data mining based procedures to generate personalized recommendations for users [1], [7]–[12]. Especially, in the e-business applications, the recommendation systems play an integral role [1], [7]. For instance, Amazon has its own integrated recommendation system to provide personalized recommendations [13].

The associate editor coordinating the review of this article and approving it for publication was Jun Shen.

The emergence of mobile social networks, such as Facebook, Google Locations, and Yelp have continously seen the increase in the number of subscribers [1], [10]. Such social networking services not only allow a user to provide explicit feedback in the form of preference rating (star rating), but also allow users to provide textual reviews about the items of interest [1]. The text mining approaches can be utilized to extract trends of users' opinions towards particular items [14], [15]. Based on such trends, a new user can be recommended with some item of interest by taking into account the similarity in the user's preferences with the existing users.

A. CHALLENGES

A predominant challenge of rating based on textual feedback is the inherent biasness caused by users' personal interest, choice, and current trends in the form of social influence. Such biasness can degrade the recommender system's performance in terms of accuracy and precision; consequently compromising the system's ability to provide high-quality recommendations. For instance, Fig. 1 demonstrates the biasness in the comments from two different reviews for

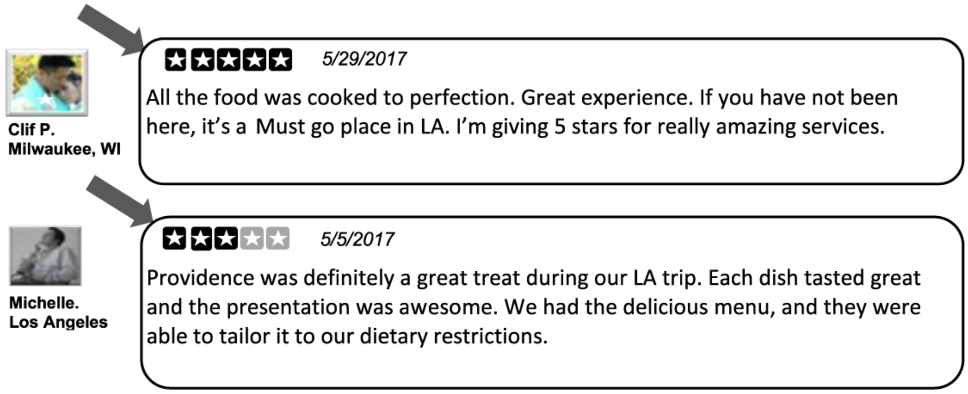

FIGURE 1. Example of biasness in user's rating.

a restaurant. There is a clear difference of opinion in rating assignments by the two reviewers. As shown in Fig. 1, the two users Clif and Michelle wrote about quality of a restaurant in LA, USA. Both the users have provided a positive feedback and seem to be pleased with their experience at the restaurant. The users described the restaurant service with multiple positive words, such as "perfection", "great experience", "awesome". However, the first user gave 5 stars to the restaurant whereas the second user gave 3 stars.

User-generated preference rating can play a significant role in the popularity of a venue of interest. However, such rating systems are often targeted by rating spammers who seek to distort the perceived popularity of a venue by creating fraudulent rating. To improve the popularity of any particular venue, one of the business tricks is to hire people who make fake identity and rate the desired venue high by assigning highest star rating. Such rating will increase the overall popularity of the venue and the targeted venue becomes popular amongst the other venues not representing the actual popularity trend.

B. MOTIVATION

In scientific literature, several works, such as [2], [7], [9], [16], [17] have applied Collaborative Filtering (CF) to the recommendation problem. The CF-based approaches intend to generate recommendations based on the similarity in actions and preferences of users [18]–[21]. Generally, CF-based recommendation systems operate on the ratings entered by users in the form of explicit feedbacks, usually from a 1-to-5 rating scale. However, many users prefer to use free form text to express their opinions. Reviews written by tourists about tourist spots are popular source of information that may influence users' decisions to choose among the spots [22]. For instance, a user may want to acquire information about a certain feature or aspect of a venue, such as quality of service and decor. Despite the importance of such information, CF-based recommendation systems ignore the significance of feedback embedded in the textual review, especially in a scenario when enough explicit feedback in the form of numerical rating is not available [21]. Therefore, it is important to bridge the gap between recommendations based on numeric ratings, and those of users' opinions entered as free text to reach an unbiased rating about an item/venue.

C. CONTRIBUTIONS

In this paper, we propose SocialRec, a hybrid context-aware recommendation framework that performs recommendations based on mining and analysis of users' textual feedbacks. We attempt to address some of the inherent issues of recommendation systems, such as cold start and data sparseness. To address the cold start, our framework utilizes the Hub-Average (HA) inference model [23] that maintains a pre-computed list of most popular venues in a user's current vicinity. To address data sparseness caused by zero similarity values, we enhanced the CF algorithm by utilizing textual review as an additional source of user preferences. To make our system context-aware, we consider the region, e.g., a user's current city, and the category of item of interest, of which the user wants the recommendations.

In summary, the contributions of our work are as follows.

- We proposed a Context-Aware Recommendation Framework, termed as *SocialRec* that utilized the aggregated preference score acquired from preference rating and textual opinion to suggest optimal venues recommendations.
- A data pre-processing phase is introduced to minimize the data sparseness and cold start problem.
- We perform extensive experiments on our internal OpenNebula cloud setup. The experiments were conducted on real-world "Yelp" dataset [17].

The rest of the paper is organized as follows. Section 2 discusses the related work. Section 3 presents the system overview. In Section 4, we discuss the *SocialRec* framework. Section 5 presents the performance evaluation with simulation results, and Section 6 concludes the paper.

II. RELATED WORK

In the past, most work focused on predicting the user's preferences by utilizing the collaborative filtering based methods [1], [18], [24]–[26]. The authors in [27] proposed a collaborative sequential map filtering algorithm for E-learning. The proposed algorithm is designed to assist users in accessing learning resources for individual as well as collaborative learning. Jain *et al.* proposed a movie recommendation system based on collaborative and content-based filtering [28]. The authors in [2] utilized GPS trajectories of the travelling passengers and geotagged photos with collaborative filtering to recommend preferred point of interests to tourists. In [9], the authors utilized collaborative filtering to produce recommendation for a group of users that satisfy the most members of the group. The authors applied Ant Colony Algorithm on a dataset filtered with Hubs/Authorities approach to compute the expert users and popular venues. The new user is then recommended with venues that were visited by the experts whose preferences match with the new user. Cui *et al.* in [29] proposed a context-aware recommendation algorithm with two level SVD named CTLSVD. First, the authors divide the rating matrix into user matrix and item matrix. Then, more refined factor vectors are extracted and SVD is applied to divide the user and item matrix into two matrices. Finally, the time as contextual information is utilized to filter out the unsuitable recommendation results to improve the overall quality of recommendations. Irfan *et al.* applied multiple objective optimization to compute an optimal list of venues against a given user's preferences [10]. A main issue with approaches based on collaborative filtering is that they are compute intensive and consume a major chunk of memory if the data size is quite large. Moreover, a user's rating may be biased and may not actually reflect the textual feedback of a user. This motivated the inclusion of text mining approaches in the field of recommendation systems.

Recently, many scientific literatures (e.g., [14], [30]–[40]) have applied text mining approaches to improve performance of recommendation systems by augmenting numerical rating with sentiment analysis of textual reviews. The authors in [30] performed the sentiment analysis of textual data extracted from Facebook and Twitter to capture the emotional state of a user when he/she is performing the review and then used this information to address the cold start issue. Teso *et al.* applied text mining techniques to the online available user-generated content and analyzed shared reviews to find the gender of users [14]. The authors intended to extract the differences between male and female discourses in a specific product category. In [31], the authors used logistic regression in sentiment analysis to compute a user's sentiment score. The authors built an item-feature matrix to calculate the improved similarity of items for recommendations. Yang *et al.* utilized text mining on textual contents posted on social gaming sites to find the personality traits of a player [32]. Recommendation is then performed based on the similarity between the traits of the user and the game. The authors in [33] proposed a time-aware recommender system with an assumption that a user's preferences may change over time. The authors classify users with similar vocabulary in the given reviews in same groups, and then compute the similarity between authors based on sentiment analysis of reviews.

An on demand ubiquitous venue recommendation system was proposed by Liu *et al* [39]. The authors utilized explicit ratings, an implicit opinion, and collaborative filtering method to provide optimal venue recommendations. The authors introduced a method termed as PORE to infer a user's preference and to recommend venues to users. Similarly, a personalized venue recommendation system is introduced by Taylor *et al.* [37] that utilized textual contents from *TripAdvisor* and introduced aspect-based opinion mining technique to discover users' preferences for venue selection. A similar approach is presented in [38] and [39] that keep track of users' traveling point-of-interest, reduce computation cost, and suggest the best route to the user by utilizing the users' textual comments. Differing from the aforementioned work, in our paper we do not keep track of user's travel point-of-interest that causes heavy computation. In [41], the authors propose a scheme for social image tag refinement. The proposed approach considers user, visual, and tag information with a tri-clustered tensor completion framework to improve the social image tagging. Shu *et al.* proposed a weakly-shared Deep Transfer Network (DTN) to translate cross-domain information from web texts domain to image domain [42]. The authors modeled two stacked autoencoders that takes paired input of text and image, followed by multiple parameter-sharing network layers at the top. The output of the shared layer in DTN yields the translator function that can be used to transfer cross-modal information. Cambria *et al.* [43] presented a language visualization and analysis system for concept-level sentiment analysis. They presented a vector space model, AffectiveSpace 2, that performs reasoning by analogy on natural language concepts, even when these are represented by highly dimensional semantic features [44].

In contrast to the above mentioned approaches, our proposed framework specifically targets the venue recommendation systems, and similarity computations among users by utilizing rating inference approach, which infer the numerical rating from textual reviews so that users' preferences can easily be fed into existing CF methods that predict the users' interests and preferences for an unvisited venue. To address the data sparsity problem usually found in most of the existing schemes, we introduce a pre-processing phase to find the mutual reinforce relationships among users and venues. The venues and users are given scores, and venues with higher scores are recommended to a new user. In the following Section, we discuss proposed system components.

III. SYSTEM OVERVIEW

This section provides an overview of the proposed system along with major components. The proposed *SocialRec* framework maintains a record of sentiment score for each review by detecting polarity from the textual review given by a user. Moreover, a rating history is also maintained that contains a record of the venues' preference rating, the users' information, and venues' information whose feedback is provided by the users.

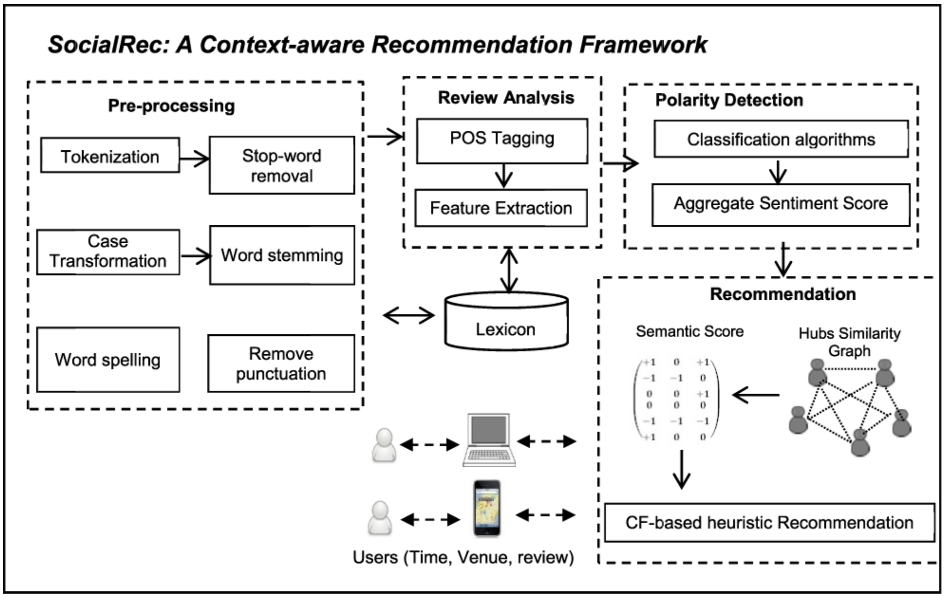

FIGURE 2. Polarity detection with machine learning algorithms.

As indicated in **Fig. 2**, the proposed framework architecture comprises of four modules, namely: (a) review pre-processing module, (b) review analysis module (c) polarity detection module, and (d) recommendation module. The pre-processing module transforms the unstructured textual data into tokenized and structured format by eliminating noisy text, such as spelling mistakes, grammatical errors, and improper casing [34]. Pre-processing module ensures the quality of the text in terms of comprehensibility and representativeness. Various steps, such as tokenization, word stemming, and stop-word removal have been implemented to refine the text for further offline processing. Moreover, we define the boundaries of the text for sentence-wise better understanding. All the above-mentioned pre-processing phases refine the data that will be aggregated and utilized during the next review analysis module.

Review analysis module analyzes each and every sentence of the review and categorizes the words in each sentence according to the grammatical structure. Part-of-Speech (POS) tagging technique have been implemented to categorize the words of the sentence syntactically that play a vital role in identification of relevant feature and opinion of the comment generated by a reviewer. Moreover, we proposed a method to extract the relevant features that are mostly under discussion in feature extraction process.

In polarity detection phase we compute the polarity of every single sentence in a review using different classification algorithms, such as Naïve Bayes and Support Vector Machine (SVM) [35]. Such classification algorithms assign a polarity score to every sentence in a review. Finally, we calculate the aggregated polarity score of every review that can be further utilized for the online recommendation process.

Online recommendation module inputs the sentiment score of the review and recommends the top-N venues for an active reviewer (N is the number of venues recommended by the framework). The recommendation module computes numerical ranks of reviewers and venues by utilizing the HA-based inference model [45]. The basic idea of the HA-based inference model is to assign ranking to the reviewers and venues based on a mutual reinforcement relationship [45]. A reviewer is assigned a higher rank (and called as expert in later the text), if the reviewer has given feedback about many higher ranked venues. Similarly, a venue gets a higher score (called as popular in the later text) if the venue is given feedback by many higher ranked reviewers [23] , [46]. Moreover, the recommendation module computes a similarity graph of the expert reviewers. The reviewers and venues that have very low scores are pruned from the dataset during online recommendation phase to reduce the online processing time. The recommendation module utilizes a heuristic based approach to generate suggestions in the form of venues that best matches a user's preferences. The venues at the top of the recommendation list will be the ones that most satisfy the user's preferences.

IV. THE SOCIALREC FRAMEWORK

In this section, we discuss in detail the computations involved in various phases of the proposed framework. **Table 1** lists some of the selected parameters used in the framework.

A. REVIEW PRE-PROCESSING PHASE

As discussed earlier, the purpose of the pre-processing phase is to remove noisy text, such as grammatical mistakes, spelling errors, improper casing, ad-hoc abbreviations, incorrect punctuations, and malformed sentences from the users' input reviews [36]. Such noisy text can complicate the text mining process and increase the dimensionality of the text. Scientific literature witnessed the implication of

Algorithm 1 Sentence Boundaries Transformation

Input: A set R of reviews
Output: A set $R^{'}$ of bounded sentences in a review set R.
Definitions: $\{D\}$ = set of pre-defined words in a dictionary, w = set of words in a review.

for each word w $\in$ R **do**
  if w.end =”.” and w.end $-1 \neq$ D[i] **then**
    SentenceBoundary(true)
    $R^{'} \leftarrow R^{'} \cup \{w\}$
    capitalize (0, w.end +1)
  else if *w.end* = “.” **then**
    capitalize (0, w.end +1)
    $R^{'} \leftarrow R^{'} \cup \{w\}$
  else
    SentenceBoundary(false)
  end if
  if w.end =”.” and
    w.end +1 = digit or w.end −1= digit **then**
    SentenceBoundary(false)
  end if
end for
return $R^{'}$

TABLE 1. Notations and their meaning.

Notation	Meaning
V	Vocabulary set
w_k	kth word in the dictionary
b_j	Feature vector with jth sentence
S^j	jth sentence
C	Review class C
R	Set of n reviews
M_r	User to venue matrix
p_v	Score of popular venues
e_u	Score of expert reviewers
S_{xy}	Set of venues rated by x and y
r_{xv}	Rating of venue x by user v

pre-processing phase as a substantial improvement of text classification process [36]. The common pre-processing steps considered in the paper are discussed in the subsequent text.

1) WORD STEMMING, TOKENIZATION, AND STOP-WORD REMOVA

Tokenization is the process of splitting the sentence into different words, such as number, punctuation marks, and names [47]. Another morphological technique is remove-stop-word. Stop-words, such as “the”, “am”, “an”, and “a”, construct the syntactic structure of the sentence and are the most frequently occurring words. However, these words do not contribute enough to represent the information [36], [47]. Therefore, stop-words are removed from the text corpus. Word stemming is another morphological technique that refers to a linguistic normalization to remove the prefixes and suffixes from a word. For instance, the word “connection” is reduced to the root word “connect.”

2) LOWER TO UPPER CASE TRANSFORMATIO

Proper use of lower case and upper case is necessary for the syntactic interpretation of the sentence. Syntactically correct sentences end with predefined punctuation markers, such as exclamation mark (!), full stop (.) and interrogation mark (?). We have employed rule-based approach to identify sentence boundaries in noisy text. Sentence boundary detection comprises of two major tasks: (a) identifying end of sentences based on correct punctuation mark implication and (b) disambiguation of full stop (.) from decimal point and abbreviated ending. Basic steps for sentence boundary detection are presented in **Algorithm 1**. If the word end with a symbol “.” and the word is not preceded by a pre-defined set of words defined in the dictionary (e.g., Org., Prof.) then, the symbol “.” can be treated as a sentence boundary. The alphabet immediately after the sentence boundary can be converted into uppercase letter. Moreover, the symbol “.” will be ignored if it appears immediately after or before the digit.

3) IRRATIONAL USE OF PUNCTUATION MARKS

Irrational use of punctuation also causes noise in the text. If a punctuation symbol is a valid mark and appears at the end of the sentence, then only one instance of the symbol will be retained in the sentence. Similarly, if the symbol is not a valid punctuation mark, then the symbol can be removed from the sentence. For instance, a visitor posted a review may read as follows: “*Grape leaves are a popular starter. An order yields a little collection of cigar shaped rolls with the perfect ratio of soft, supple leaves and flavorful rice!!!!!!!!!*” The previous sentence is refined in pre-processing as follows:

Algorithm 2 Feature-Opinion Extraction

Input: A collection of n number of reviews $R = \{r_1,\ r_2, r_3, \ldots, r_n\}$
Output: *FPlist* = A set of opinion-feature pair.
tempF ← ∅; tempAdj ← ∅; c ← 0
for each *sentence s in Review R* **do**
  $P \leftarrow ExtractOpinion\ (s, adj)$
  $tempAdj_s \leftarrow P$
  $OFP_s \leftarrow (user\_id, venue\_id, s\_id, \emptyset, tempAdj_s)$
end for
for each *sentence s in Review R* **do**
  for each *adj in tempAdj$_s$* **do**
    $nph \leftarrow ExtractNoun(s, adj)$
    $c \leftarrow noun\_frequen\_count(nph)$
    $tempF \leftarrow nph$
    $OFP_s \leftarrow (user\_id, venue\_id, s\_id, tempF, tempAdj_s, c)$
  end for
end for
for each sentence s *in Review R*
  if $OFP_s.c >$ *threshold frequency*
    $FPlist \leftarrow FPlist \cup \{OFP_s\}$
  end if
end for
return *FPlist*

"Grape leaves are a popular starter. An order yields a little collection of cigar shaped rolls with the perfect ratio of soft, supple leaves and flavorful rice!"

4) WORD SPELLIN

Erroneous spellings are the major hindrance to extract meaning from text. If a word is erroneously spelt, then it may lead to incorrect interpretation of the meaning associated with a text. We have employed a *PyEnchant* library [48], a free available spell checker that replaces the miss-spelt word with the most probable correct words from the dictionary. It is worth mentioning here that the focus of the system is not to correct all the errors in the text. The basic purpose of pre-processing phase is to focus on minimizing errors in opinion mining.

B. REVIEW ANALYSIS

The review analysis phase analyzes the linguistic features of review so that the opinion about the review can be identified. Two majorly adopted tasks for review analysis are POS tagging and feature extraction. The detailed description of the aforementioned is presented in subsequent text.

1) POS TAGGING

POS tagging is an important step in the proposed framework. POS tagging reflects the syntactic category of the words that play a vital role in identification of relevant features from reviewers' sentences. A rule-based approach (Brill tagging) is implemented using nltk [49] to parse each review and split text into sentences. Such sentences are further divided and assigned POS tag for each single word. The taggers extract the nouns, verbs, and adjectives' information from the reviewers' comments. The review sentence with the POS tag is further used for feature extraction (subjectivity detection) and feature reduction steps.

2) FEATURE EXTRACTION AND REDUCTION

Sentence level sentiment classification comprises of feature extraction and feature reduction phases. Feature extraction process identifies subjective and objective sentences from the review. Objective sentence in a review does not contain users' opinion, whereas subjective sentences contain users' opinion. For instance, sentence 1 is objective sentence and sentence 2 and 3 are subjective in the following review: *"Me and my friend visited Sam Choy's restaurant. Breakfast sandwich served on French toast with a side of syrup is great. For lunch, their hot Panini sandwiches are excellent."* Usually, nouns (e.g., life, help, issue, pain), verbs (e.g., like, degrade), and adjectives (e.g., bitter, delicious) are used for subjectivity determination of a word. We used *SentiWord Net* [50], a lexical resource specifically designed for sentiment classification and opinion mining applications. A sentence in a review that comprises of noun, verb, and adjective is referred to as subjective sentence, otherwise, the sentence is referred to as an objective. The objective sentences do not contribute enough in opinion orientation. Therefore, the objective sentences are extracted from the review in the phase of feature reduction. Feature reduction step reduces the dimensionality of the reviewed comment that consequently reveals better results in classification process. In most of the sentences, the opinions are expressed by utilizing adjectives.

The detailed description of opinion-feature extraction process is described in **Algorithm 2**. A set of collection of reviews are the input to the algorithm. We extract the opinion from every single sentence s of each review R. The opinions are obtained by extracting the adjective of every sentence. Simultaneously, we assign the Opinion Feature Pair (OFP) as user ID, venue ID, sentence ID, Ø, and adj, (where Ø represents the feature is empty) into a global OFP set. After the initial opinion set is extracted based on the adjectives, we extract the features associated with every opinion in Line 8-Line 12. Features are usually presented as noun or noun phrase denoted as *nph*. In a single sentence, each noun centered within an opinion window is added to the candidate feature set and the corresponding OFP is updated. We assume the distance between two neighbor words is 1. For the opinion window in a sentence, we assume the opinion as a center point. Each noun or noun phrase with the distance to the center less than 5 is extracted by utilizing *ExtractNoun()* function in Line 9. Consequently, the frequency of every noun or noun phrase (if repeated) is also accumulated in Line 10. The OFP is also updated with the newly derived noun or noun phrase as (reviewer_id, venue_id, sentence_id, feature, adj, c). In Line 15-Line 19 only those pair of OFP are extracted that have higher noun frequency count than user defined threshold frequency. We are more interested in the frequently discussed features. Therefore, the infrequently discussed features are pruned and the updated list of OFP will be further used for polarity detection using classification.

C. POLARITY DETECTION

Polarity detection process classifies sentences of a review as positive, negative, and neutral. We prefer sentence-level polarity classification as the sentence-level polarity detection provides a more fine-grained interpretation of each sentence in a review [51]. The basic mechanism of polarity detection by classification is presented in **Fig. 3**.

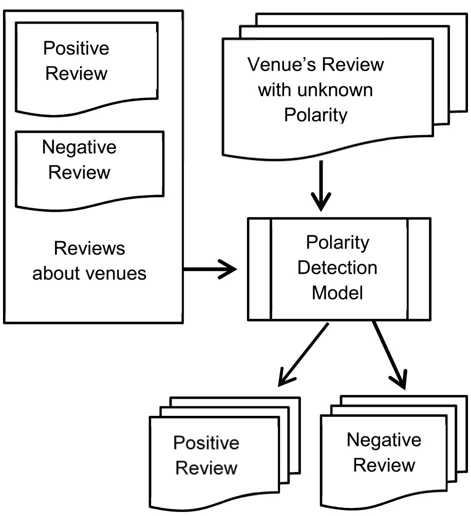

FIGURE 3. Polarity detection with machine learning algorithms.

Various classification algorithms have been widely deployed for polarity detection [35]. In our particular scenario, each review also has a preference integer rating, and training and testing data are readily available. Generally, a review with 4 to 5 preference ratings is considered a positive review and a review with preference ratings 1 to 2 is considered a negative review. The existing studies present various supervised learning methods to classify the sentences into positive, negative and neutral sentences. We selected Naïve Bayes and Support Vector Machine (SVM) for the classification of the sentences into positive, negative, and neutral sentences, as these techniques have better comparative performance for text classification [35].

1) NAÏVE BAYES MODEL FOR SENTIMENT CLASSIFICATION

Naïve Bayes classifier is a probabilistic machine learning technique [46]. The classifier models the distribution of the reviews in each class using a probabilistic model. We assume that the reviews are modelled according to a Bernoulli document model [23] that computes the posterior probability of each class based on the distribution of words in a review. In Bernoulli document model, presence and absence of words in a review is considered as a binary vector that represents a point in a space of words. If we have a vocabulary set V of $|V|$ words, then the kth dimension of a review's vector corresponds to word w_k in the vocabulary. Let b_j be the feature vector for the jth sentence S^j, then the kth element of b_j termed as b_{jk} is either 0 or 1 representing the absence and presence of word w_k in the jth sentence of a review. Let $P(w_k|C)$ be the probability of occurrence of a word w_k given a review of class C. The probability of w_k not occurring in a review of class C is given by $1 - P(w_k|C)$. To classify each unlabeled sentence S^j in a review, we estimate the posterior probability for each class as follows

$$P\left(C \mid S^j\right) = P(C) \prod_{k=1}^{|V|} b_{jk}\big[P(w_k|C) + \left(1 - b_{jk}\right) 1 - P(w_k|C)\big]. \quad (1)$$

$$s.t \quad P = \begin{cases} P(w_k|C) & \text{if } b_{jk} = 1 \\ & \text{otherwise} \\ (1 - P(w_k|C)). & \text{if } b_{jk} = 0 \end{cases} \quad (2)$$

2) SVM MODEL FOR SENTIMENT CLASSIFICATION

SVM is a very popular machine learning technique for the text classification [52]. SVM finds an optimal hyperplane represented by vector $\vec{v}$ that separates a review in positive class from a review in negative class. As presented in a **Fig. 4**, a set of positive and negative labeled reviews' vector is considered to be linearly separable if there exists a vector $\vec{v}$ and a scalar b such that the following inequalities are applicable

$$Polarity(s) = \begin{cases} Positive & \vec{v} \cdot \vec{r} - b \geq +1 \\ Negative & \vec{v} \cdot \vec{r} - b < -1 \\ Neutral & \vec{v} \cdot \vec{r} - b = 0 \end{cases} \quad (3)$$

where s is the sentence in single review. The margin of the decision boundary is presented by the distance between the

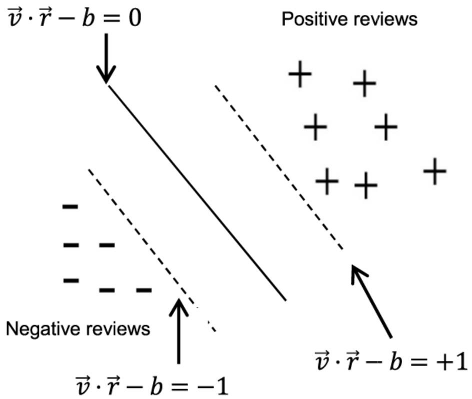

FIGURE 4. **Positive and negative review representation using SVM.**

two hyperplanes $\vec{v}\cdot\vec{r}-b=+1$ and $\vec{v}\cdot\vec{r}-b=-1$, and these hyperplans are shownas dotted lines in Fig. 4 [52].

3) AGGREGATED SENTIMENT SCORE

After classification phase, each sentence has a sentiment score $+1$, 0, -1 for positive, neutral, and negative sentence, respectively. In aggregate sentiment score phase, we aggregate the sentiment score of each sentence to obtain the overall score of the entire review. Let R be a set of n reviews $\{r_1, r_2, r_3, \ldots, r_n\}$ by a single user for a single venue, and S be a set of m sentences in each review $\{s_1, s_2, s_3, \ldots, s_m\}$. The overall sentiment score calculated for each sentence in the set of reviews as follows

$$Score_{overall} = \sum_{i=1}^{n}\sum_{j=1}^{m} Sco_{r_i s_j}, \tag{4}$$

where Sco is the sentiment score calculated for each sentence in a review. The $Score_{overall}$ will be used in user-to-venue matrix to compute popular venues and users in Subsection D.1. The polarity of a review can be defined as

$$Polarity\,(r) = \begin{cases} Positive & if \ \sum_{i=1}^{n} Sco_{s_i} > 0 \\ Negative & if \ \sum_{i=1}^{n} Sco_{s_i} < 0 \\ Neutral & if \ \sum_{i=1}^{n} Sco_{s_i} = 0. \end{cases} \tag{5}$$

The neutral sentences can be considered as objective sentences containing no information about the venue and can be extracted from the dataset.

D. RECOMMENDATIO

In terms of functionality, the proposed CF-based hybrid recommendation model has three main modules: (a) popularity ranking of reviewers and venues, (b) similarity graph generation among popular reviewers, and (c) recommendation module that is responsible for generating the recommendation for a reviewer. The detailed functionality of the above mentioned modules is discussed in the following subsections.

1) REVIEWER-VENUE POPULARITY RANKING

This subsection presents the process of assigning popularity ranking to reviewers and venues. The higher ranked venues and reviewers are known as popular venues and expert reviewers, respectively. HA inference model [45] is utilized to perform the ranking for producing a set of expert reviewers and popular venues. To compute the expert reviewers' and popular venues', first we need to create a user-to-venue matrix. Let the matrix be represented by M_r, and its values are the scores computed in (4). Let $[p_v]$ and $[e_u]$ represent the score matrices for a popular venue and an expert reviewer, respectively, with initial values set to 1s. The following formulas compute the score for popular venues and expert reviewers [9].

$$p_v = M_r^T \times e_u. \tag{6}$$
$$e_u = M_r \times p_v. \tag{7}$$

If we use $p_v^{<n>}$ and $e_u^{<n>}$ to represent the score of popular venue and expert reviewers at nth iteration, then the following equations generate the score of popular venues and expert reviewers iteratively.

$$p_v^{<n>} = (M_r^T \times M_r) \times p_v^{<n-1>}. \tag{8}$$
$$e_u^{<n>} = (M_r \times M_r^T) \times e_u^{<n-1>}. \tag{9}$$

The purpose behind using HA method is to generate a subset of reviewers who have commented popular venues, and a subset of venues that are frequently commented by expert reviewers.

2) REVIEWER-VENUE SOCIAL GRAPH CREATIO

This phase creates social graph among expert reviewers. The idea is to generate a network of like-minded people (reviewers) who share the similar comments by assigning same sentiment score for various venues. The graph constructed in current phase will be made available for CF-based heuristic recommendation process that finds an optimal path on the graph. Such a path carries a collective opinion about venues by expert reviewers who are also most similar to an active reviewer.

The similarity computation between two reviewers in the similarity graph is performed by applying the Pearson Correlation Coefficient (PCC) [1]. The value of PCC ranges between -1 and $+1$. Positive values indicate that the similarity exists between two reviewers with highest similarity at 1, whereas negative PCC values means the choices of the two reviewers does not match. PCC is computed by using (11).

In (11), the similarity between two reviewers x and y is computed only for venues that are commented by both of the reviewers.

$$sim(x,y) = \frac{\sum_{v\in S_{xy}}(r_{xv}-\overline{r}_l)(r_{yv}-\overline{r}_y)}{\sqrt{\sum_{v\in S_{xy}}(r_{xv}-\overline{r}_u)^2 \sum_{v\in S_{xy}}(r_{yv}-\overline{r}_y)^2}}, \tag{10}$$

where

$$S_{ij} = \{v \epsilon V \mid r_{xv} \neq 0 \wedge r_{yv} \neq 0\}$$

The similarity computation in (11) results into a very sparse similarity graph due to the fact that majority of the venues have no feedback provided by either of the two reviewers. Therefore, to address the data sparseness problem, we augment the similarity computation with the *trust measure*. The trust measure can be interpreted as a conditional probability that given a feedback provided for a venue by a reviewer, the feedback for the same venue is also provided by another reviewer in the dataset. Moreover, it depicts the amount of trust (or confidence) showed by both the reviewers in the venues commonly reviewed by them. The following equation is utilized to calculate the weight of an edge between two reviewers.

$$\omega_{ij} = \begin{cases} sim(x,y) & \text{if } sim(x,y) > 0 \\ & \text{otherwise} \\ P\left(r_x \mid r_y\right) \times \dfrac{1}{1 + \sum\limits_{x \in V_y} |r_{xv} - r_{yv}|}, & P[r_y] \neq 0, \end{cases} \tag{11}$$

where V_y is the set of venues checked-in by user y. The parameter $P\left(r_x \mid r_y\right) = P\left[r_x \cap r_y\right]/P[r_y]$ is the likelihood ratio that both reviewers may have checked-in at the similar set of venues. The expression multiplied by probability keeps the overall value lower than the similarity, so that the similarity is given preference. Using (11), an edge weight is assigned in the graph if the similarity value is greater than zero, otherwise, the lower term value is used as an edge weight. This helps in addressing the data sparseness issue that results due to zero similarity values.

3) HEURISTIC RECOMMENDATION APPROACH

In this subsection, a heuristic approach is presented that generates a set of top-N venue recommendations in a user's current context based on a graph of the expert reviewers. The graph of expert reviewers for a given region (e.g., Newyork City) and a category of interest (e.g., restaurant) as a user's context is retrieved from the database. The similarity of the active user is computed with all of the reviewer nodes in the graph using (11) and only those nodes are connected with active user with which the similarity is greater than zero. Each edge of the graph has a weight that is calculated by utilizing the weight computation formula described in (11). The edges connecting the nodes at the same level L are intentionally labeled blank as represented in **Fig. 5**, because they are not traversed during the execution of **Algorithm 3**. The top-N venues recommended by the heuristic approach are the ones that were not previously visited by the active reviewer. Algorithm 3 illustrates the step by step procedure of the heuristic approach for online recommendations.

1. Initializations (Line 1–Line 5):

- The identification of the active user, type of venues to be recommended for active user and features of the venues (category and location) for which the user needs recommendation are taken as the input of the Algorithm 3.

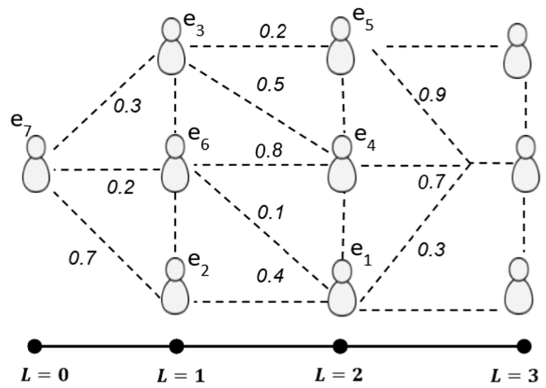

FIGURE 5. Expert reviewers' similarity graph.

- In Line 2 and Line 3, the similarity graph of the expert reviewers is retrieved. Only those neighbors of active user are selected from the graph that have non-zero similarity computation with the active user. In Line 4, the current reviewer node is stored in list V

2. Iterative solution construction (Line 5–Line 22):

- In the Line 5, the weights are assigned to each neighbor node (N_a) based on the similarity function $sim\,(a, j)$ (defined in (11)) that is further multiplied by the $1/\Delta_{rj}$ that is the edge count between the active reviewer and neighboring node.
- Only those venues are selected from the neighboring nodes that were not previously visited by the active user (Line 7). The selected venues are appended in the matrix A. The visited neighbor is stored in the list V (Line 6–Line 10).
- If at Line 11, the venue count in the matrix A is greater than the required number of venues N, then the control jumps to Line 22 that generates the ranking of the venues in matrix A.
- If the required venue count is not achieved, then new active node (a) is selected amongst the neighbor set N_a. The criterion for the new active node selection is that the node must have the maximum of the required number of venues. If no such node is found, then the control parses the Line 23. Otherwise, the edge count will also be incremented in Line 18 and in Line 19 and the control will jump back to Line 5.

3. Aggregate venues provided by the best nodes (Line 22):

- The venues are ranked to generate top-N venues to be recommended to the active user. The following equation is used to rank the venues.

$$Rank_x = \frac{\sum_{e \in V} w\,(r, e) \times r_{ex}}{\sum_{e \in V} w(r, e)}. \tag{12}$$

In (12), x is the venue to be ranked, the parameter r is the active reviewer node, and r_{ex} is the review score calculated for expert reviewer $e \in V$ at venue x. The parameter $w(r, e)$ represents the weight of the link in the similarity graph between the root node r and the expert reviewer e.

Algorithm 3 *SocialRec* Venue Recommendation Algorithm

Input: Active user : r, Category: C, region: R

Output: A set $S^{'}$ of top-N venues visited by expert reviewer similar to active user.

Definitions N_j = *neighbor set of node j*, Δ_{ij} = *edge count between reviewers i* and*j*, w_{aj} = *edge weight between reviewer a and j*, Z_j = *number of required venues found at a node j*, V = *list of reviewers traversed by the active user r*

1: $a \leftarrow r; L \leftarrow 1; V \leftarrow \emptyset$
2: $G_f \leftarrow SimGraph\,(C, R)$
3: $N_a \leftarrow \{x : G_f \,|\, sim\,(a, x) > 0\}$
4: $V \leftarrow a$
5: $\forall j \in N_a, w_{aj} \leftarrow [sim\,(a, j) \times 1/\Delta_{rj}],\ j \in N_a$
6: **for** each $e \in N_a$ **do**
7: $S \leftarrow \{v : V_e | v \notin V_r\}$
8: $A \leftarrow A.append(e, S)$
9: $V \leftarrow V \cup \{e\}$
10: **end for**
11: **if** $venueCount(A) \geq N$ **then**
12: go to Line 23
13: **else**
14: $\forall j \in N_a$, select $a \leftarrow j$, such that we have arg $max \left[w_{aj} \times \frac{Z_j}{N} \right] \wedge N_j \neq \emptyset \wedge \forall g \in N_j | g \notin V$
15: **if** *No any such node found in Step* 14 **then**
16: go to Line 23
17: **else**
18: $L \leftarrow L + 1;$
19: go to Line 5
20: **end if**
21: **end if**
22: $S^{'} = generaterank\,(A)$
23: **return** $S^{'}$

V. PERFORMANCE EVALUATION

In this section, we present the performance evaluation of the proposed *SocialRec* framework. We compare our framework with the following schemes: (a) User-based Collaborative Filtering (UBCF) [53], (b) Singular Value Decomposition (SVD) [52], (c) Random Walk with Restart (RWR) [45], and (d) Popular [41]. We utilized *"Yelp"* dataset that consists of 6,442,890 check-ins performed by 150,734 users at a total number of 1,280,969 venues [45]. In the selected dataset, out of the entire records, 80% of the records are used as the training set and 20% constitute the test set for the evaluation. We used a standard 10-fold cross validation technique for evaluating the accuracy rate of the proposed *SocialRec* framework [16].

A. PERFORMANCE METRICS

Four commonly used performance evaluation metrics have been utilized to evaluate the proposed *SocialRec* recommendation frameworks: (a) precision, (b) recall, and (c) F-measure. The precision represents a ratio of the precise recommendations (true positive (tp)) to the total number of predicted recommendations (tp + false positive (fp)). A precise recommendation is the recommendation that has been predicted correctly in the top-*N* recommended venues.

$$Precision = \frac{tp}{tp + fp}. \tag{13}$$

The recall measures the average quality of the individual recommendations. The recall presents the proportion of all the precise recommendations in the top-*N* recommended venues and can be represented as:

$$Recall = \frac{tp}{tp + fn}. \tag{14}$$

The F-measure is the harmonic mean of precision and recall and denoted as follows:

$$Fmeasure = \frac{2 \times Precision \times Recall}{Precision + Recall}. \tag{15}$$

B. RESULTS

A series of simulation runs are conducted to test the performance of the proposed *SocialRec* system and the results are shown in **Fig. 6** (SocialRec is abbreviated as SRec in the graphs).

We compare *SocialRec* with the UBCF, SVD, Popular, and RWR algorithms using the underlying classifiers as SVM, NB, and *preference* rating. As depicted in the **Fig. 6**, the *SocialRec* outperforms the other schemes in terms of precision, recall and F-measure, respectively. The sentiment classifiers address the inherent biasness caused by the users' personal interest by filtering out actual positive and negative reviews from textual data that has been acquired through the classification process. Such biasness in users' textual reviews can easily be evident in the "Yelp" dataset through statistical analysis as presented in the **Fig. 7**. Out of 335,023 number of reviews, only 53.23% of the reviews had similar preference rating and sentiment score. In the original dataset, there were 68.57% of preference score of 4 or 5, 17.9% of preference score of 3, and 14.24% of preference score of less than 3.

Fig. 7 shows that out of 68.57% of positive reviews only 63.9% reviews were those that are actually positive reviews, and the rest of 28.4% are identified as neutral reviews marked as positives reviewers. Similarly, out of 68.57% of positive reviews, 7.7% reviews were wrongly marked as positive reviews during preference rating. The figure also depicts that out of 17.19% of neutral reviews indicated by preference rating, there were 45.19% positive, 39.79% neutral, and 15.05% negative, when evaluated by sentiment score. Instead of relying on the integer scale (preference rating), the proposed scheme utilize the sentiment classifiers to classify the textual feedback into positive and negative reviews, consequently providing better recommendations in terms of high precision, accuracy, and F-measure.

Considering the preference rating, the *SocialRec* framework provides better solution for data sparseness problem by augmenting the similarity computation with

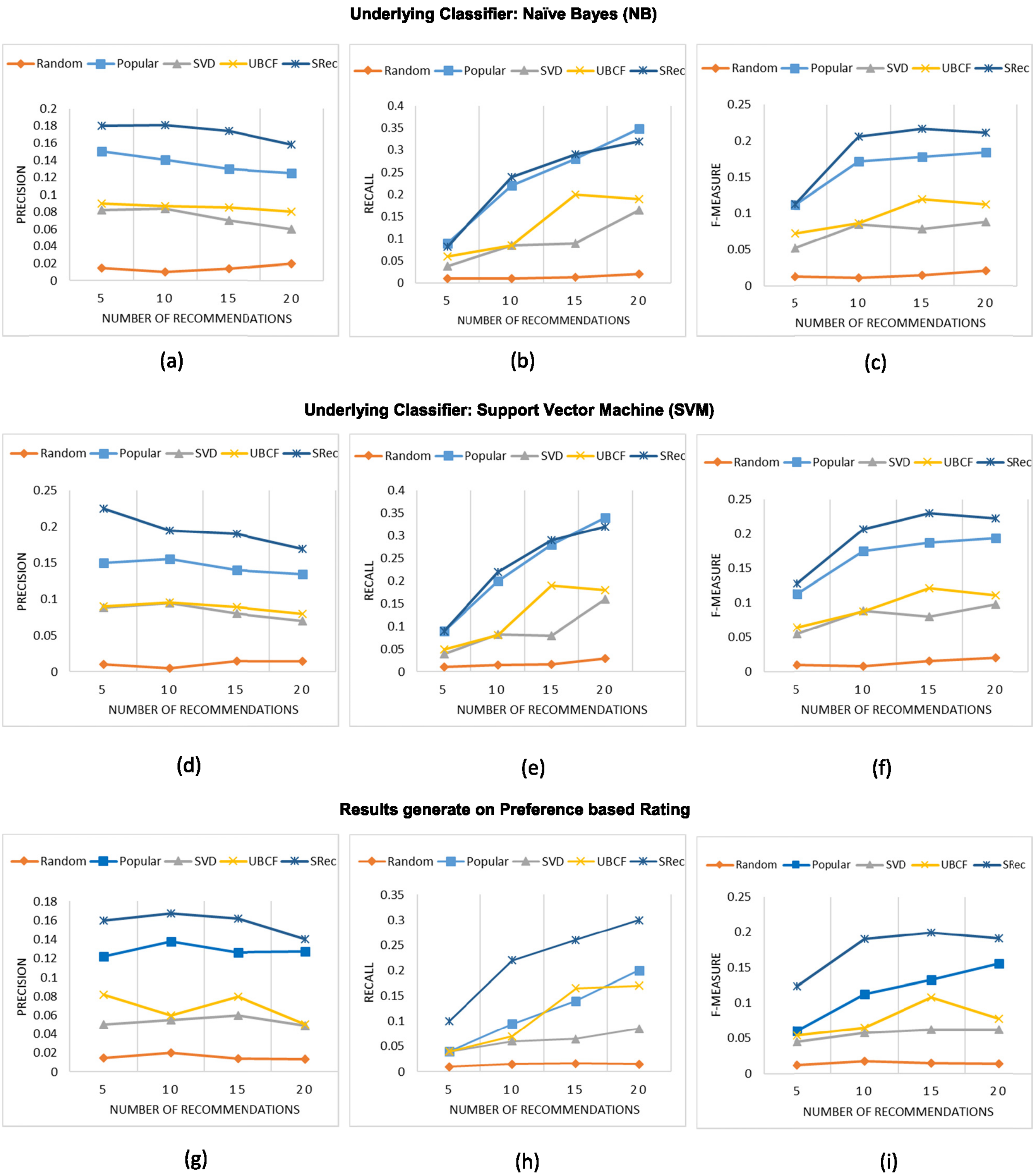

FIGURE 6. Performance evaluation results for NB, SVM, and Preference.

conditional probability. Data sparseness results in zero similarity values when we have sparse preference rating matrix. No matter if the users have visited the same venues, the difference in the visit counts will decrease the similarity of users. The zero similarity values cause data sparsness. We addressed the cold start problem by utilizing the HA inference model that helps inferring for the most popular venues within a specific region. The application of confidence measure and HA inference model effectively helps to obtain better solution that results in an increased recommendation precision.

The well-known collaborative filtering techniques, such as SVD and UBCF presented low performance in terms of precision, recall, and f-measure due to high data sparseness. The popularity-based approach exhibited comparatively better performance than the collaborative filtering approach. The main reason is that the popularity-based approach does not compute the similarity matrix, however, with tradeoff of reduced precision. Therefore, the popularity-based approach is not significantly affected by data sparseness problem.

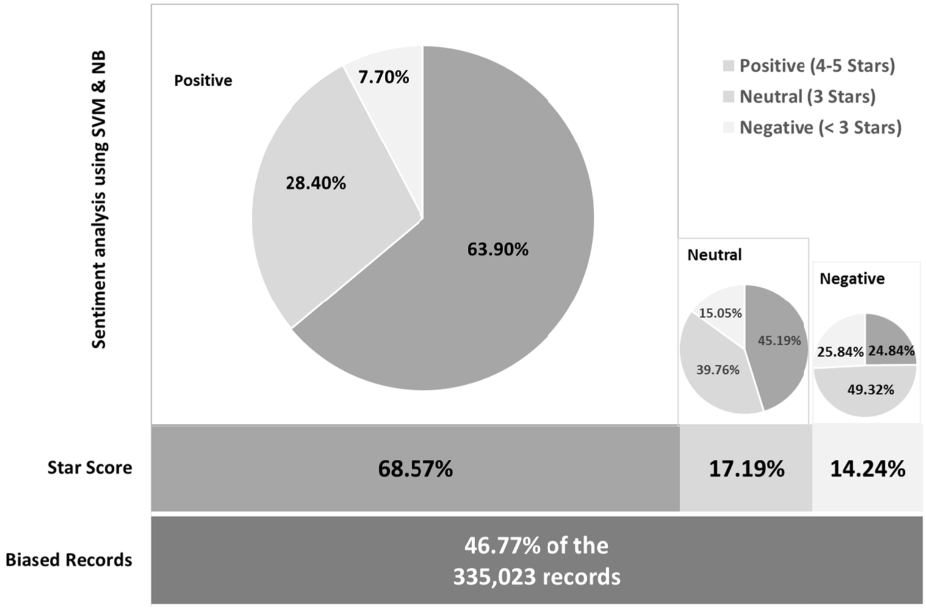

FIGURE 7. Statistical analysis of positive and negative reviews.

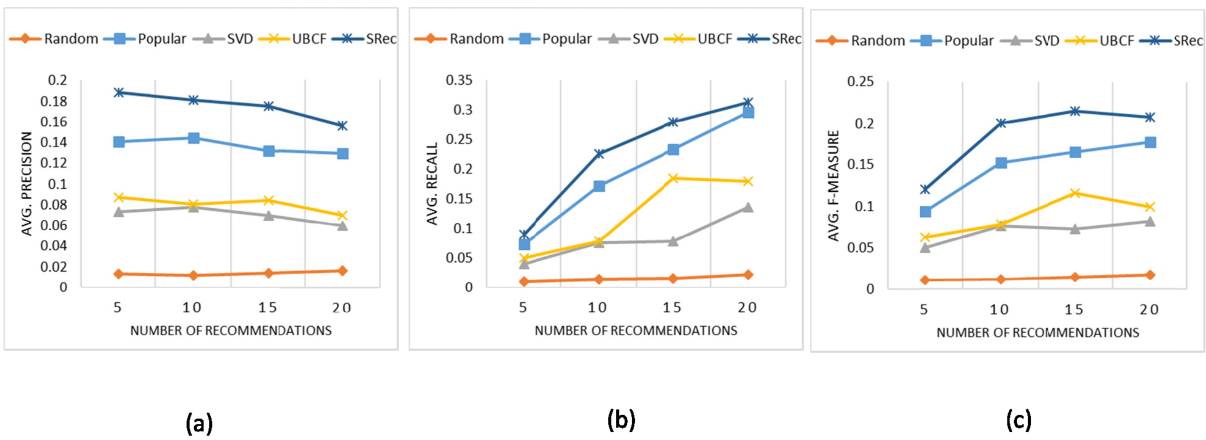

FIGURE 8. Performance evaluation for top-K recommendations.

As presented in **Fig. 6 (b)**, the recall of *SocialRec* framework is the highest for $N = 15$, which indicates that the framework provides a greater coverage in terms of recommendations. The performance of RWR remains low for all the aforementioned metrics.

Fig. 8 presents the average performance of the proposed scheme for top K recommendations. We can observe from the figure that as the number of recommendations K increases, the precision value of almost all the techniques in **Fig. 8** also decreases. This phenomenon has been well reported in the existing literature of recommendation systems. However, even at $K = 20$, the average oveall peformance of our proposed scheme (*SRec*) remains superior than the existing schemes.

C. DISCUSSIONS

By considering not only the explicit star ratings, but also the significance of embedded text in feedback, our proposed scheme attempts to address the inherent biasness caused by the deviation in a feedback's numerical rating and sentiments. Compared to just only star ratings, such approach may not be having very high precision/accuracy values, but the ratings are more optimal as they are encompasing the inherent embedded features that are mostly neglected by the traditional rating based recommender systems. Therefore, the proposed framework can be beneficial for applications such as restaurant recommendation, tourists spots recommendations, movie recommendations, etc., where the user sentiments play an important role.

VI. CONCLUSION

We have presented a hybrid recommendation framework named *SocialRec* to produce recommendations by considering textual data and utilizing the user textual review for computing recommendations. The significance of the proposed framework is the adaptation of popular collaborative filtering and sentiment classification, such as SVM and NB method to compute positive and negative reviews for the recommendation. We addressed the data sparseness by augmenting the similarity computation with trust measure based on the intuition that the visit patterns of two users may be different, however, they might have visited the same venues. Such knowledge is utilized in computation of confidence measure.

Our future work involves the use of multi-objective optimization to find out an optimal solution from multiple conflicting objectives, thus improving the overall quality fo recommendations. Moreover, we intend to use our expertise in machine learning in detection of fake news, online bots, and many nlp related applications.

REFERENCES

[1] F. Rehman, O. Khalid, and S. A. Madani, "A comparative study of location-based recommendation systems," *Knowl. Eng. Rev.*, vol. 32, no. e7, pp. 1–30, Jan. 2017.

[2] A. Majid, L. Chen, G. Chen, H. T. Mirza, I. Hussain, and J. Woodward, "A context-aware personalized travel recommendation system based on geotagged social media data mining," *Int. J. Geograph. Inf. Sci.*, vol. 27, no. 4, pp. 662–684, Apr. 2013.

[3] M. Shardlow, R. Batista-Navarro, P. Thompson, R. Nawaz, J. McNaught, and S. Ananiadou, "Identification of research hypotheses and new knowledge from scientific literature," *BMC Med. Informat. Decis. Making*, vol. 18, no. 1, p. 46, Dec. 2018.

[4] P. Thompson, R. Nawaz, I. Korkontzelos, W. Black, J. McNaught, and S. Ananiadou, "News search using discourse analytics," in *Proc. Digit. Heritage Int. Congr. (Digit. Heritage)*, Oct. 2013, pp. 597–604.

[5] P. Thompson, R. Nawaz, J. McNaught, and S. Ananiadou, "Enriching news events with meta-knowledge information," *Lang. Resour. Eval.*, vol. 51, no. 2, pp. 409–438, Jun. 2017.

[6] S. Mohammad, M. U. S. Khan, M. Ali, L. Liu, M. Shardlow, and R. Nawaz, "Bot detection using a single post on social media," in *Proc. WorldS4*, 2019.

[7] F. Rehman, O. Khalid, K. Bilal, N. Haq, K. Bilal, and S. A. Madani, "Diet-Right: A smart food recommendation system," *KSII Trans. Internet Inf. Syst.*, vol. 11, no. 6, pp. 2910–2925, Jun.2017.

[8] R. Yunus, O. Arif, H. Afzal, M. F. Amjad, H. Abbas, H. N. Bokhari, S. T. Haider, N. Zafar, and R. Nawaz, "A framework to estimate the nutritional value of food in real time using deep learning techniques," *IEEE Access*, vol. 7, pp. 2643–2652, 2019.

[9] O. Khalid, M. U. S. Khan, S. U. Khan, and A. Y. Zomaya, "OmniSuggest: A ubiquitous cloud-based context-aware recommendation system for mobile social networks," *IEEE Trans. Services Comput.*, vol. 7, no. 3, pp. 401–414, Jul./Sep. 2014.

[10] R. Irfan, O. Khalid, M. U. S. Khan, C. Chira, R. Ranjan, F. Zhang, S. U. Khan, B. Veeravalli, K. Li, and A. Y. Zomaya, "MobiContext: A context-aware cloud-based venue recommendation framework," *IEEE Trans. Cloud Comput.*, vol. 5, no. 4, pp. 712–724, Oct./Dec. 2017.

[11] M. U. S. Khan, O. Khalid, Y. Huang, R. Ranjan, F. Zhang, F. Zhang, J. Cao, B. Veeravalli, S. U. Khan, K. Li, A. Y. Zomaya, "MacroServ: A route recommendation service for large-scale evacuations," *IEEE Trans. Services Comput.*, vol. 10, no. 4, pp. 589–602, Jul./Aug. 2017.

[12] H. Qadir, O. Khalid, M. U. S. Khan, A. U. R. Khan, and R. Nawaz, "An optimal ride sharing recommendation framework for carpooling services," *IEEE Access*, vol. 6, pp. 62296–62313, 2018.

[13] B. Smith and G. Linden, "Two decades of recommender systems at amazon.com," *IEEE Internet Comput.*, vol. 21, no. 3, pp. 12–18, May/Jun. 2017.

[14] E. Teso, M. Olmedilla, M. R. Martínez-Torres, and S. L. Toral, "Application of text mining techniques to the analysis of discourse in eWOM communications from a gender perspective," *Technol. Forecasting Social Change*, vol. 129, pp. 131–142, Apr. 2018.

[15] R. T. Batista-Navarro, G. Kontonatsios, C. Mihăilă, P. Thompson, R. Rak, R. N. I. Korkontzelos, and S. Ananiadou, "Facilitating the analysis of discourse phenomena in an interoperable NLP platform," in *Proc. Int. Conf. Intell. Text Process. Comput. Linguistics*, 2013, pp. 559–571.

[16] J. J.-C. Ying, E. H.-C. Lu, B.-N. Shi, and V. S. Tseng, "TripCloud: An intelligent cloud-based trip recommendation system," in *Proc. Int. Conf. Adv. Spatial Temporal Databases*, 2013, pp. 472–477.

[17] Q. Qi, Z. Chen, J. Liu, C. Hui, and Q. Wu, "Using inferred tag ratings to improve user-based collaborative filtering," in *Proc. 27th ACM Symp. Appl. Comput.*, Mar. 2012, pp. 2008–2013.

[18] M. Eirinaki, J. Gao, I. Varlamis, and K. Tserpes, "Recommender systems for large-scale social networks: A review of challenges and solutions," *Future Gener. Comput. Syst.*, vol. 78, pp. 403–418, Jan. 2018.

[19] A. Bellogín, P. Castells, and I. Cantador, "Improving memory-based collaborative filtering by neighbour selection based on user preference overlap," in *Proc. 10th Conf. Open Res. Areas Inf. Retr.*, May 2013, pp. 145–148.

[20] C. Cechinel, M. Á. Sicilia, S. Sánchez-Alonso, and E. García-Barriocanal, "Evaluating collaborative filtering recommendations inside large learning object repositories," *Inf. Process. Manage.*, vol. 49, no. 1, pp. 34–50, Jan. 2013.

[21] Y. Shi, M. Larson, and A. Hanjalic, "Collaborative filtering beyond the user-item matrix: A survey of the state of the art and future challenges," *ACM Comput. Surv.*, vol. 47, no. 1, p. 3, Jul. 2014.

[22] R. Nawaz, P. Thompson, and S. Ananiadou, "Identification of manner in bio-events," in *Proc. LREC*, May 2012, pp. 3505–3510.

[23] P. Pitchandi and N. Raju, "Improving the performance of multivariate Bernoulli model based documents clustering algorithms using transformation techniques," *J. Comput. Sci.*, vol. 7, no. 5, pp. 762–769, 2011.

[24] S. Ayyaz, U. Qamar, and R. Nawaz, "HRS-CE: A hybrid framework to integrate content embeddings in recommender systems for cold start items," *J. Comput. Sci.*, vol. 29, pp. 9–18, Nov. 2018.

[25] S. Ayyaz, U. Qamar, and R. Nawaz, "HCF-CRS: A hybrid content based fuzzy conformal recommender system for providing recommendations with confidence," *PLoS ONE*, vol. 13, Oct. 2018, Art. no. e0204849.

[26] S. Ananiadou, P. Thompson, and R. Nawaz, "Enhancing search: Events and their discourse context," in *Proc. Int. Conf. Intell. Text Process. Comput. Linguistics*, 2013, pp. 318–334.

[27] P. Appalla, R. Selvaraj, V. M. Kuthadi, and T. Marwala, "Hybrid fuzzy recommendation system for enhanced E-learning," *Adv. Syst., Control Autom.*, vol. 442, Dec. 2017, pp. 21–32.

[28] K. N. Jain, V. Kumar, P. Kumar, and T. Choudhury, "Movie recommendation system: Hybrid information filtering system," in *Intelligent Computing and Information and Communication*. 2018, pp. 677–686.

[29] L. Cui, W. Huang, Q. Yan, F. R. Yu, Z. Wena, and N. Lu, "A novel context-aware recommendation algorithm with two-level SVD in social networks," *Future Gener. Comput. Syst.*, vol. 86, pp. 1459–1470, Sep. 2018.

[30] F. G. Contratres, S. N. Alves-Souza, L. V. L. Filgueiras, and L. S. DeSouza, "Sentiment analysis of social network data for cold-start relief in recommender systems," in *Proc. WorldCIST*, 2018, pp. 122–132.

[31] D. Miao and F. Lang, "A recommendation system based on text mining," in *Proc. IEEE Int. Conf. Cyber-Enabled Distrib. Comput. Knowl. Discovery*, Oct. 2017, pp. 318–321.

[32] H. Yang, C. S. Lin, Z. Huang, and T. Tsai, in *Proc. 4th Multidisciplinary Int. Social Netw. Conf.*, Jul. 2017.

[33] P. Kefalas and Y. Manolopoulos, "A time-aware spatio-textual recommender system," *Expert Syst. Appl.*, vol. 78, pp. 396–406, Jul. 2017.

[34] G. Ganu, Y. Kakodkar, and A. Marian, "Improving the quality of predictions using textual information in online user reviews," *Inf. Syst.*, vol. 38, no. 1, pp. 1–15, Mar. 2013.

[35] M. Roy, "Empirical study of different classifiers for sentiment analysis," *Data Mining Knowl. Eng.*, vol. 6, no. 4, 2014.

[36] D. Munková, M. Munk, and M. Vozár, "Data pre-processing evaluation for text mining: Transaction/sequence model," *Procedia Comput. Sci.*, vol. 18, pp. 1198–1207, Jan. 2013.

[37] E. Marrese-Taylor, J. D. Velásquez, F. Bravo-Marquez, and Y. Matsuo, "Identifying customer preferences about tourism products using an aspect-based opinion mining approach," *Procedia Comput. Sci.*, vol. 22, pp. 182–191, Jan. 2013.

[38] G. Chen and L. Chen, "Recommendation based on contextual opinion," in *User Modeling, Adaptation, and Personalization* (Lecture Notes in Computer Science), vol. 8538. 2014, pp. 61–73.

[39] X. Liu, Y. Liu, and K. Aberer, "Personalized point-of-interest recommendation by mining users' preference transition," in *Proc. 22nd ACM Int. Conf. Inf. Knowl. Manage.*, Oct. 2013, pp. 733–738.

[40] H. Liu, J. He, T. Wang, W. Song, and X. Du, "Combining user preferences and user opinions for accurate recommendation," *Electron. Commerce Res. Appl.*, vol. 12, no. 1, pp. 14–23, Jan./Feb. 2013.
[41] J. Tang, X. Shu, G.-J. Qi, Z. Li, M. Wang, S. Yan, and R. Jain, "Tri-clustered tensor completion for social-aware image tag refinement," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 8, pp. 1662–1674, Aug. 2017.
[42] X. Shu, G. Qi, J. Tang, and J. Wang, "Weakly-shared deep transfer networks for heterogeneous-domain knowledge propagation," in *Proc. 23rd ACM Int. Conf. Multimedia*, Brisbane, QLD, Australia, Oct. 2015, pp. 35–44.
[43] E. Cambria, J. Fu, F. Bisio, and S. Poria, "AffectiveSpace 2: Enabling affective intuition for concept-level sentiment analysis," in *Proc. 29th AAAI Conf. Artif. Intell.*, Feb. 2015, pp. 1–7.
[44] M. Jahangir, H. Afzal, M. Ahmed, K. Khurshid, and R. Nawaz, "An expert system for diabetes prediction using auto tuned multi-layer perceptron," in *Proc. Intell. Syst. Conf. (IntelliSys)*, Sep. 2017, pp. 722–728.
[45] B. Chandra and S. Bhaskar, "Patterned Growth algorithm using hub-averaging without pre-assigned weights," in *Proc. IEEE Int. Conf. Syst., Man, Cybern.*, Oct. 2010, pp. 3518–3523.
[46] V. Narayanan, I. Aora, and A. Bhatia, "Fast and accurate sentiment classification using an enhanced naive Bayes model," in *Intelligent Data Engineering and Automated Learning* (Lecture Notes in Computer Science), vol. 8206. 2013, pp. 194–201.
[47] J. Atwan, M. Mohd, and G. Kanaan, "Enhanced arabic information retrieval: Light stemming and stop words," in *Proc. Int. Multi-Conf. Artif. Intell. Technol.*, vol. 378, 2013, pp. 219–228.
[48] *PyEnchant Library*. Accessed: Jul. 29, 2019.[Online]. Available: http://pythonhosted.org/pyenchant/
[49] R. Forsati and M. Shamsfard, "Hybrid PoS-tagging: A cooperation of evolutionary and statistical approaches," *Appl. Math. Model.*, vol. 38, no. 13, pp. 3193–3211, Jul. 2014.
[50] *SentiWordNet*. Accessed: Jul. 29, 2018. [Online]. Available: http://sentiwordnet.isti.cnr.it/
[51] J. K.-C. Chung, C. C.-E. Wu, and R. T-H. Tzong, "Polarity detection of online reviews using sentiment concepts: NCU IISR team at ESWC-14 challenge on concept-level sentiment analysis," in *Semantic Web Evaluation Challenge*, vol. 475, 2014, pp. 55–58.
[52] S. Maldonado and G. L'Huillier, "SVM-based feature selection and classification for Email filtering," in *Pattern Recognition—Applications and Methods*, vol. 204, 2013, pp. 135–148.
[53] L. Lu, M. Medo, C. H. Yeung, Y. Zhang, Z. Zhang, and T. Zhou, "Recommender systems," *Phys. Rep.*, vol. 519, no. 1, pp. 40–49, 2012.

RIZWANA IRFAN received the master's degree from the COMSATS Institute of Information Technology, Islamabad, Pakistan, and the Ph.D. degree from North Dakota State University, USA. She is currently an Assistant Professor with the COMSATS Institute of Information Technology. Her research interests include recommender systems and knowledge engineering.

OSMAN KHALID received the master's degree from the Center for Advanced Studies in Engineering and the Ph.D. degree from North Dakota State University, USA. He is currently an Assistant Professor with the COMSATS Institute of Information Technology. His research interests include recommender systems, network routing protocols, the Internet of Things, and fog computing.

MUHAMMAD USMAN SHAHID KHAN received the master's degree from National University of Science and Technology (NUST), Islamabad, and the Ph.D. degree from North Dakota State University, USA. He is currently an Assistant Professor with the COMSATS Institute of Information Technology. His research interests include recommender systems, cyber security, and the Internet of Things.

FAISAL REHMAN received the Ph.D. degree in computer science from COMSATS University, Abbottabad, Pakistan, in 2018, where he is currently an Assistant Professor. His research interests include recommender systems and machine learning.

ATTA UR REHMAN KHAN was the Director of the National Cybercrime Forensics Laboratory Pakistan, the Head of the Air University Cybersecurity Center, and the Conference Chair of the IEEE Islamabad Section. He is currently an Associate Professor with the Faculty of Computing and Information Technology, Sohar University, Oman. His research interests include cybersecurity, mobile cloud computing, ad hoc networks, and the IoT. He is a Steering Committee Member/Track Chair/Technical Program Committee (TPC) Member of over 60 international conferences. He also serves as a domain expert for multiple international research funding bodies, and has received multiple awards, fellowships, and research grants. He is also serving as an Associate Editor of IEEE ACCESS and the *Journal of Network and Computer Applications* (Elsevier), an Associate Technical Editor of the *IEEE Communications Magazine*, and an Editor of the *Journal of Cluster Computing* (Springer), *Oxford Computer Journal*, the IEEE SDN NEWSLETTER, the *KSII Transactions on Internet and Information Systems*, *Human-Centric Computing and Information Sciences* (SpringerOpen), *SpringerPlus*, and *Ad hoc & Sensor Wireless Networks journal*. For more updated information, visit his website at www.attaurrehman.com.

RAHEEL NAWAZ served in various senior leadership positions in the private Higher and Further Education sector; and was an Army Officer before that. He is currently the Director of the Digital Technology Solutions and a Reader in Analytics and Digital Education with Manchester Metropolitan University (MMU). He has founded and/or headed several research units specializing in artificial intelligence, digital transformations, data science, digital education, and apprenticeships in higher education. He has led on numerous funded research projects in the U.K., EU, South Asia, and Middle East. He holds adjunct or honorary positions with several research, higher education, and policy organizations, both in the U.K. and overseas. He regularly makes media appearances and speaks on a range of topics, including digital technologies, artificial intelligence, digital literacy, and higher education.