

Augmenting service recommender systems by incorporating contextual opinions from user reviews

Guanliang Chen · Li Chen

Received: 17 August 2014 / Accepted in revised form: 16 February 2015 /

Published online: 17 March 2015

© Springer Science+Business Media Dordrecht 2015

Abstract Context-aware recommender systems have been widely investigated in both academia and industry because they can make recommendations based on a user's current context (e.g., location, time). However, most existing context-aware techniques only use contextual information at the item level when modeling users' preferences, i.e., contextual information that correlates with users' overall evaluations of items such as ratings. Few studies have attempted to detect more fine-grained contextual preferences at the level of item aspects (e.g., a hotel's "location", "food quality", and "service"). In this study, we use contextual weighting strategies to derive users' *aspect-level context-dependent preferences* from user-generated textual reviews. The inferred context-dependent preferences are then combined with users' *context-independent preferences* that are also inferred from reviews to reflect their stable requirements over time. To automatically incorporate both types of user preferences into the recommendation process, we propose a linear-regression-based algorithm that uses a stochastic gradient descent learning procedure. We tested the proposed recommendation algorithm with two real-life service datasets (one with hotel review data and the other with restaurant review data) and compared its contribution with three previously suggested approaches: one that does not consider contextual information; one that uses contextual information to pre-filter rating data before applying the recommendation algorithm; and one that generates recommendations according to users' aspect-level contextual preferences. The experiment results demonstrate that our approach outperforms the others in terms of recommendation accuracy.

G. Chen (✉) · L. Chen

Department of Computer Science, Hong Kong Baptist University, Hong Kong, China
e-mail: anguschan7733@gmail.com

L. Chen

e-mail: lichen@comp.hkbu.edu.hk

Keywords Context-aware recommender systems · Service recommendation · User reviews · Contextual review analysis · Context-independent preference · Context-dependent preference · Stochastic gradient descent learning

1 Introduction

In the era of ubiquitous and pervasive computing (Jameson and Krger 2005), the increasing amount of personal data collected by digital devices (e.g., smart phones, Google glass) can be used to construct accurate and flexible user models for personalized recommender systems. Most existing systems use *contextual* data (Adomavicius and Tuzhilin 2011; Yu et al. 2006), which generally refer to any information that characterizes the situation of an entity (e.g., a person, a place, or an object) (Abowd et al. 1999). For example, in a typical context-aware recommendation approach named *contextual pre-filtering* (Adomavicius et al. 2005), when the recommender is estimating the rating of an item for the target user, it considers data from other users that were acquired in the same context, because these data are more relevant for predicting the target user's contextual preference. Empirical studies indicate that context-aware approaches can produce more accurate recommendations than non-context-aware approaches (Karatzoglou et al. 2010; Adomavicius et al. 2005; Zheng et al. 2013; Hariri et al. 2012; Park et al. 2006). However, most existing context-aware recommendation methods are limited in that the users' preferences are modeled purely at the item level (i.e., the contextual preferences are related to the overall evaluations of items); they do not consider that the preferences can be modeled at the more fine-grained aspect level. Aspects are general features that are used to describe the item. For example, a hotel may have aspects such as "location", "food quality", and "service" (Liu et al. 2011; Jannach et al. 2012; Ganu et al. 2013). Indeed, users' aspect-level preferences can be likely to be influenced by contextual factors, especially for *service items* (i.e., items consisting of certain business services in return for money, such as hotels, restaurants, movies, etc.) (Fuchs and Zanker 2012). Consider a hotel review from *TripAdvisor*¹ (see Fig. 1) as an example. We can clearly see that this reviewer, in the context of a *business trip*, places more emphasis on the aspect "location", but if he was taking a *family trip*, the aspect "room" would become more important. Therefore, understanding users' contextual preferences as they relate to aspects should be meaningful.

As the goal of this study is to develop more effective service recommender systems, we propose a method for deriving users' aspect-level contextual preferences. Given the increasing number of users who share their experiences (i.e., opinions) with products and services in *online reviews* (Moghaddam and Ester 2012), we exploit the value of this type of textual information to accomplish our goal. Specifically, we contribute to the development of context-aware recommender systems in the following ways: (1) we develop an automatic technique for extracting aspect-level contextual opinions from user-generated reviews; (2) we use contextual weighting strategies to derive users' aspect-level contextual preferences; and (3) we implement a stochastic gradient

¹ www.tripadvisor.com.



Fig. 1 A hotel review example from *TripAdvisor*. The user's opinions about the item's aspects are *highlighted with full lines*, and the contexts are *highlighted with dash lines*

descent learning method to automatically integrate users' contextual preferences into the recommendation process. In our technique, we discriminate between two types of user preferences: *context-dependent* and *context-independent*. The *context-dependent preferences* are the aspect-level contextual preferences that are common to users in the same context, whereas *context-independent preferences* reflect users' stable requirements for an item's aspects over time and are, as a result, less sensitive to contextual change.

An intuitive method to determine the *context-dependent preferences* is to count an aspect's occurrence frequency (i.e., the occurrence frequency of any term related to the aspect²) in reviews written in a specific context. In other words, the more frequently an aspect is mentioned, the more important it is to users in that context (i.e., the higher its weight) (Levi et al. 2012). However, this method cannot distinguish between aspects that appear the same number of times. We argue that the relative importance of each aspect-related term should also be considered when determining the aspect's weight. To this end, we borrow knowledge from text categorization (Yang and Pedersen 1997) and propose three alternative contextual weighting methods for determining a term's weight. Each variant is based on a different text feature selection strategy: mutual information (MI), information gain (IG), and chi-square statistic (CHI). On the other hand, *context-independent preferences* can also be extracted from reviews, but to do this accurately it is necessary to consider different properties between new users and repeated users. For new users (i.e., those with few history records in system Jamali and Ester 2009; Massa and Avesani 2007), we apply the probabilistic regression model (PRM) that can detect the preferences of new users by treating the detection as a Bayesian learning process. For repeated users (i.e., those with abundant history data), we compare the effectiveness of two models, i.e., PRM and the linear regression model (LRM), as the latter can be used to detect users' preferences in a rich data condition.

² In reviews, terms that are descriptive of a certain aspect are denoted as *aspect-related terms*; for example, terms "service", "waiter", and "waitress" are related to aspect "service" in hotel reviews.

Finally, to automatically combine the two types of user preferences into the recommendation process, we propose a linear-regression-based algorithm that uses a stochastic gradient descent learning procedure. We demonstrate the superior accuracy of our approach in comparison with the related methods on two real-life datasets (hotel reviews from *TripAdvisor*, and restaurant reviews from *Yelp*³).

The rest of this article is organized as follows. We first summarize the related works, and classify them into two categories: context-aware and review-based recommender systems (Sect. 2). After discussing their respective strengths and limitations, we state our research problem and sketch the flow of our proposed system (Sect. 3). In Sect. 4, we describe the methodology we have developed. We compare the variations of our method with the related approaches in Sect. 5, and summarize the experiment results and discuss our work's practical implications and limitations in Sect. 6. Finally, we conclude with our main findings and describe directions for future research (Sect. 7).

2 Related work

Our work is closely related to two types of recommender system: context-aware recommender systems and review-based recommender systems. In this section, we describe the state-of-the-art on these two subjects.

2.1 Context-aware recommender systems

One of broadly accepted definitions of context is given in (Abowd et al. 1999): “*Context is any information that can be used to characterize the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and application themselves.*” Adomavicius and Tuzhilin (2011) classified existing context-aware recommendation techniques into three categories according to the phase of the process in which the contextual information is applied: (1) *contextual pre-filtering* uses context to filter out irrelevant rating data before running a classical recommendation approach such as collaborative filtering (Adomavicius et al. 2005; Panniello et al. 2009); (2) *contextual post-filtering* uses context to distill the recommendation results after the classical approach has been applied (Panniello et al. 2009); and (3) *contextual modeling* directly incorporates context into the recommendation model (Zheng et al. 2013; Karatzoglou et al. 2010). Although contextual pre/post-filtering based approaches have been successful in some applications, researchers have pointed out that they are highly dependent on the selection of the recommendation algorithm, and a simple filtering strategy can cause the loss of valuable contextual information and hence damage the system's prediction accuracy (Adomavicius et al. 2005; Panniello et al. 2009; Karatzoglou et al. 2010). In comparison, contextual modeling based approaches provide a more natural way to capture the interaction between user behavior and related context, so they have received more attention in recent years. For example, Karatzoglou et al. (2010)

³ www.yelp.com.

modeled the user-item-context relationship as a multiple dimensional tensor, which is an extension of the traditional two-dimensional (i.e., user-item) matrix factorization model. The tensor model is approximated by applying the stochastic gradient descent method. However, these approaches mostly use contextual information as hard constraints, and this cannot work well when there is severe data sparsity phenomenon.

A new trend in context-aware recommender systems is to measure the similarity between ratings acquired in different contexts. In this way, it can be decided whether a rating given in a specific context can be used to calculate recommendations in another context. For instance, [Zheng et al. \(2013\)](#) stated that the user-based k-Nearest Neighbor (k-NN) algorithm's different steps (like searching similar users and calculating a user's average rating) can be performed with different data selection strategies, among which the data are weighted by their context similarities, calculated by applying the particle swarm optimization algorithm. It is proven that this algorithm can improve prediction accuracy while maintaining prediction coverage (i.e., predicting as many unknown ratings as possible). [Codina et al. \(2013\)](#) proposed a singular value decomposition based analysis method to measure the semantic similarity between contexts, which in turn indicates the similarity between ratings attained in different contexts. Then, when computing recommendations requested in a certain context, if there is lack of the target user's history ratings pertinent to that context, the ratings acquired in other contexts are taken into account by weighting them with context similarity. These works assume that context is explicitly specified by users. However, datasets that contain both ratings and user-specified contexts are rare in practice ([Li et al. 2010](#)).

Our study can be regarded as an extension of the contextual pre-filtering based approach, as it also first filters out ratings according to the target user's contexts and then generates recommendations; but the innovation is that our pre-filtering is conducted at the aspect level instead of at the item level. Our approach is superior to the above-mentioned approaches for the following reasons: (1) it capitalizes on textual reviews to acquire users' contextual information; and (2) it refines users' preferences by establishing the relationship between aspect-level opinions and contextual factors, and then incorporates the fine-grained contextual preferences into the recommendation process.

2.2 Review-based recommender systems

The common rationale behind review-based recommenders is that advanced opinion mining techniques can transform user-generated textual reviews into opinion ratings. For instance, some studies inferred the so-called virtual ratings from reviews ([Zhang et al. 2013](#)). The inferred ratings have been found to be comparable to users' real ratings for the purposes of performing collaborative filtering techniques ([Zhang et al. 2013](#); [Leung et al. 2006](#); [Poirier et al. 2010](#)). In addition, researchers have attempted to combine users' real ratings and review texts when they are both available. For example, [Pero and Horvth \(2013\)](#) incorporated both users' real ratings and ratings inferred from reviews into the Matrix Factorization model in which either (1) real ratings are adjusted by inferred ratings before being input to the model; (2) inferred ratings and real ratings serve as separate inputs into the model and the resulting predictions are combined for

recommendation; or (3) both inferred ratings and real ratings are used in the training phase when constructing the model.

In another sub-branch of research, aspect-level ratings are derived from reviews and used to represent the reviewer's perception of an item from multiple dimensions. For instance, [Ganu et al. \(2013\)](#) developed a multi-label text classifier based on the support vector machine to classify review sentences into different aspect categories (e.g., food, service) and sentiment categories (i.e., positive and negative). Sentences classified into a specific *(aspect, sentiment)* pair are used to calculate the opinion rating of the corresponding aspect. All of the aspects' opinion ratings are then used to produce recommendations through regression-based and clustering-based algorithms. In a different approach, [Wang and Chen \(2012\)](#) and [Chen and Wang \(2013\)](#) used the latent class regression model to leverage reviews so that they can detect reviewers' cluster-level weight preferences placed on features, and then use these preferences to compute user-user similarity during the recommendation process. [Dong et al. \(2013a, b\)](#) harnessed the extracted product features (i.e., aspect-related terms) and the accompanying opinions to build product profiles. These profiles are used to prioritize retrieved products that are similar to a user's query product and have also been positively reviewed by users. In contrast to these heuristic-based algorithms, some researchers developed model-based recommendation approaches for capitalizing on aspect-level ratings as derived from reviews. [Jakob et al. \(2009\)](#) used the multi-relational matrix factorization to model interactions between users, items, and users' opinions about aspects. The predicted rating of an item for the target user is calculated by multiplying the latent factors of the involved entities (i.e., user and item). Similarly, [Wang et al. \(2012\)](#) implemented a three-dimensional tensor model to accommodate the latent relationship between users, items, and aspect-level opinions. The tensor model is concretely approximated by a decomposition-based method named CP-WOPT ([Acar et al. 2011](#)), and the learnt latent factors of user, item, and overall rating are used to predict the rating. This sub-branch of work based on aspect-level ratings is essentially similar to multi-criteria recommender systems ([Adomavicius et al. 2011](#)), as the latter type of system also uses users' evaluations of multiple aspects of an item to enhance recommendation ([Liu et al. 2011](#); [Adomavicius and Kwon 2007](#); [Jannach et al. 2012](#); [Zhang et al. 2009](#)). However, unlike the ratings that users assigned to a fixed set of aspects that the system predefined in multi-criteria recommenders, reviews can contain aspects that users freely mentioned in text. Moreover, the words in a review text may more precisely indicate the reviewer's personal opinions about aspects, which may help recommenders to more accurately model her/his preferences.

Several studies have additionally used the contextual information extracted from reviews to improve recommendation accuracy. [Hariri et al. \(2011\)](#) applied the labeled latent dirichlet allocation (LDA) to extract contexts from hotel reviews and compute recommendations by taking into account both context-based and rating-based similarities when predicting an item's utility for the target user. In ([Li et al. 2010](#)), two methods, string matching based method and text classifier, were adopted to extract four types of context from restaurant reviews: *time*, *occasion*, *location*, and *companion*. This work postulates that a user's interest in an item is influenced by (1) the user's long-term preference, which can be learnt from the user's history ratings, and (2) the current context. Our approach differs from this study in the following ways: (1) we

propose to detect users' contextual preferences at the aspect level, instead of at the item level, and (2) we explicitly model users' aspect-level contextual preferences through review analysis, rather than only using the available information (i.e., users' history ratings and the extracted contexts) as input for training a probabilistic latent relational model.

All of the aforementioned studies provide insights into how to use free-text review information to improve recommender systems. However, their main limitation is that they do not explore the underlying relationship between aspect-level opinions and contexts. To the best of our knowledge, few studies have attempted to fill this gap. One work is (Carter et al. 2011). After extracting users' opinions about camera features from reviews, the authors manually correlated the opinions with the product usage information (also expressed in reviews) so as to construct aspect-context relations. This study has three limitations: (1) it requires manual effort to identify aspect-context relations; (2) the contextual influences on users' aspect-level preferences are not explicitly modeled; and (3) it lacks an experimental evaluation of their proposed recommendation algorithm. Another study (Levi et al. 2012) suggested that the aspect-level opinions expressed in users' hotel reviews can be correlated with their self-specified contexts (such as trip intent and nationality) to capture underlying aspect-context relations. The derived relations are then used to calculate users' relative weights for aspects in different contexts. This approach is still limited, as the researchers did not extract from reviews different opinions about the same aspect in different contexts (e.g., a user's different opinions about the aspect "room" in contexts *business trip* and *family trip* as expressed in a review; see Fig. 1). To overcome these limitations, we previously proposed determining users' contextual opinions through review analysis, and deriving users' aspect-level contextual preferences through using feature selection metrics (Chen and Chen 2014). However, this approach is only applicable to repeated users. Another limitation is that users' contextual preferences are fused into the recommendation process via a fixed parameter that cannot adapt to changes in users' preferences between different contexts.

In comparison with the above-described methods, the innovations of our current approach are as follows: (1) we identify the effect of contextual factors on users' aspect-level preferences in a more precise way by discriminating the aspect-related term's relative importance to the context; (2) we propose a recommendation algorithm that is applicable to both new users and repeated users; and (3) we integrate users' context-dependent preferences into the recommendation process using stochastic gradient descent learning.

3 Research problems and our system's workflow

We believe that the widely available user-generated reviews on the web can be used to more accurately model users' preferences, especially preferences influenced by contextual factors; this approach considers the users' opinions about aspects of an item, instead of solely relying on their overall ratings. We particularly aim to augment service recommender systems by detecting users' aspect-level contextual preferences and combining them with context-independent preferences. This study solves two

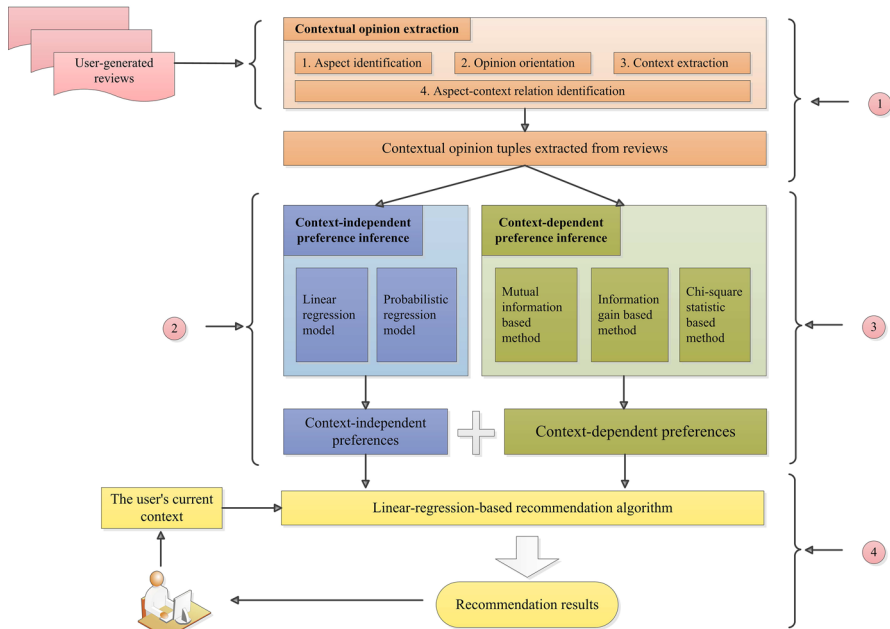


Fig. 2 The workflow of our developed recommender system that is based on contextual opinions extracted from user reviews

research problems: (1) *how to discover the relation between aspect-level opinions and contextual information in reviews, and to use this information to derive users' context-dependent preferences*, and (2) *how to leverage both context-dependent and context-independent preferences from reviews and use them together to generate recommendations*.

As discussed in Sect. 1, using an aspect's occurrence frequency in reviews as the only feature pertinent to a specific context might not truly reflect its importance to users in that context. Therefore, a more sophisticated contextual weighting method should be investigated. In addition, we also need to detect users' context-independent preferences, as they reflect users' relatively stable requirements for item aspects over time. A recommendation method should combine both types of preferences in a precise way. Our system's workflow can be seen in Fig. 2.

- (1) *Contextual opinion extraction* We first implement an automatic method to conduct contextual review analysis that will extract users' aspect-related contextual opinions from their reviews. Specifically, users' contextual opinions are their evaluations of an item's aspects (e.g., a hotel's "location", "food quality", and "service") contingent upon a certain context. We formally denote the contextual opinion as a tuple consisting of four elements: $\langle i, rev_{u,i}, a_k, Con_{u,i,k} \rangle$ that represents user u 's opinion a_k of aspect k of item i in contexts $Con_{u,i,k}$, as expressed in review $rev_{u,i}$ (where $1 \leq k \leq K$, K denotes the number of aspects, and $Con_{u,i,k}$ is a boolean vector whose element value is equal to 1 when the associated context occurs, and 0 otherwise). For example, suppose $Con_{u,i,k}$ is five-dimensional in

the restaurant domain containing five context values *family*, *friends*, *colleague*, *couple*, and *solo*. If a contextual opinion is tagged with the context values *friends* and *couple*, then $Con_{u,i,k}$ is represented as $\langle 0, 1, 0, 1, 0 \rangle$. In this step, the main question is how to extract both aspect-level opinions and contexts from reviews and reveal their relationship. We give our solution in Sect. 4.1.

- (2) *Context-independent preference inference* As mentioned before, context-independent preferences reflect users' relatively stable requirements for item aspects over time. Therefore, we believe that a user's history data can be used to determine such preference. Normally, there are two types of users in a dataset: *new users*, who have few history records; and *repeated users*, who possess abundant history data. For a new user, it is almost impossible to build a preference inference model with the limited amount of history records s/he has provided. We hence test the performance of the probabilistic regression model (PRM) to derive new users' preferences by treating the problem as a Bayesian learning process. For deriving context-independent preferences of repeated users, in addition to PRM, we investigate the linear regression model's (LRM) suitability. In our previous work (Chen and Chen 2014), we only used LRM. It is hence meaningful to experimentally compare it with PRM in the current work. The details of these two models are given in Sect. 4.2, and the results of comparing them are in Sect. 5.
- (3) *Context-dependent preference inference* In contrast to context-independent preferences, context-dependent preferences reflect users' desire for certain item aspects in a specific context. Some recent studies have pointed out that people in the same context tend to have similar preferences for item aspects (Fuchs and Zanker 2012; Levi et al. 2012); this finding motivates us to consider all of the reviews written within a context when determining users' context-dependent preferences. In other words, the inference of context-dependent preferences is not user-specific, and consequently there is no need to discriminate between new users and repeated users in this process. We concretely derive users' context-dependent preferences based on two observations: (a) a more important aspect usually has a higher occurrence frequency in reviews; and (b) aspect-related terms may be of varying importance to users in different contexts. For example, the term "Wifi" that is related to aspect "facility" may be more important to users in the context of staying in a hotel *for business* than in the context of being *with friends*, as business travelers often expect hotels to have Wifi. Hence, we first implement a frequency-based approach to assign weights to aspects, and then refine the weights using knowledge from text categorization (Yang and Pedersen 1997) that assesses the aspect-related term's relative importance. Specifically, we propose three alternative contextual weighting methods (Chen and Chen 2014) for capturing users' aspect-level contextual preferences. The three methods are respectively based on mutual information (MI), information gain (IG), and chi-square statistic (CHI), as these feature selection metrics can all be used to measure the dependency between two random variables (i.e., an aspect-related term and a context in our case); this enables us to discriminate the relative importance of an aspect-related term in different contexts. The differences between the three methods are detailed in Sect. 4.3, and their performance is tested in the experiment (see Sect. 5.4).

- (4) *Ranking and recommendation* The above three steps result in two types of preferences: *context-independent preferences* (including variations of LRM-based and PRM-based preferences) and *context-dependent preferences* (including variations of MI-based, IG-based and CHI-based preferences). To automatically incorporate both types of preferences into the recommendation process, we propose a linear-regression-based method that uses stochastic gradient descent learning (see Sect. 4.4). Moreover, as these two types of preferences can be combined in different ways, i.e., at the holistic level or the aspect level, we conduct an in-depth investigation of different combination strategies through experimental comparison (see Sect. 5.4). The recommendation algorithm returns the top- N items, and our evaluation task is to identify whether the user's *target choice* appears in the recommendation list.

In the following, we describe each step in detail.

4 Our methodology

4.1 Extracting contextual opinion tuples from reviews

As mentioned, the first task is to transform user-generated reviews into structured contextual opinion tuples, which can be formally denoted as $\{(i, rev_{u,i}, a_k, Con_{u,i,k}) \mid 1 \leq k \leq K\}$. Following related works on aspect-level opinion mining (Jakob et al. 2009; Chen and Wang 2013; Wang et al. 2010) and contextual information extraction (Li et al. 2010), we propose an automatic method to perform contextual review analysis. The proposed method has four main sub-steps.

- (1) *Aspect identification* Notice that each aspect of an item is concretely represented as a set of related terms in the reviews (e.g., the aspect “service” corresponds to the terms such as “service”, “waiter”, “waitress”, “attitude”, etc.). Therefore, we need to first identify aspect-related terms. The approaches used to complete this task in previous studies can be classified into two categories: heuristic-based and model-based. The heuristic-based approaches usually initialize each aspect with a few of predefined keywords, and then search for the other related terms by applying clustering method (Wang and Chen 2012), relying on certain syntactic relations (Wang et al. 2012), or measuring the dependency between terms (Wang et al. 2010; Jakob et al. 2009). In the model-based approaches, the latent dirichlet allocation (LDA) model has been popularly applied (Blei et al. 2003); for this model, we only need to define the number of aspects and then the aspect-related terms will be automatically retrieved. In our study, because there is prior knowledge that can be used to describe the recommended service (i.e., the aspects defined to describe the service), we prefer the heuristic-based method. We concretely apply the bootstrapping method introduced in (Wang et al. 2010), as it has been proven effective for processing service reviews. For example, for hotels, we define eight aspects: “value”, “location”, “service”, “room”, “facility”, “sleep quality”, “food quality”, and “cleanliness”; for restaurants, we define five aspects: “value”, “food quality”, “atmosphere”, “service”, and “location”. Then, each aspect is equipped

Table 1 Hotel aspects and aspect-related terms

Hotel aspects	Aspect-related terms (seed words)
Value	Value, price
Location	Location, place
Room	Room, size, bathroom
Cleanliness	Cleanliness, cleaning
Sleep quality	Sleep, bed, bedroom
Service	Service, staff, waiter
Facility	Facility, wifi, pool, gym
Food quality	Food, drink, dish, wine, salad

Table 2 Restaurant aspects and aspect-related terms

Restaurant aspects	Aspect-related terms (seed words)
Value	Value, price
Food quality	Food, drink, dish, wine, salad
Atmosphere	Atmosphere, ambiance
Service	Service, staff, waiter
Location	Location, place

with a few terms as seed words (see examples in Tables 1 and 2), and the other terms are searched out by measuring the dependency between the seed words and the candidate term based on the chi-square statistic (Yang and Pedersen 1997). Notice that only frequently occurring nouns and noun phrases, which are extracted by using a part-of-speech (POS) tagger⁴, are considered to be prospective terms.

- (2) *Opinion orientation* Adjectives and adverbs in reviews can be regarded as users' opinion carriers. We first use POS tagger to extract these words from reviews, and then determine their orientations as numeric scores (+1 for positive and -1 for negative (Ding et al. 2008; Hu and Liu 2004)) with the aid of an opinion lexicon constructed in (Wilson et al. 2005). We then consider two strategies to reveal the connection between the aspect-related terms and opinions: syntactic-based (Wang et al. 2012; Jakob et al. 2009) and distance-based (Levi et al. 2012; Ding et al. 2008). The syntactic-based approach relies on certain syntactic patterns such as *adjectival modifiers* (e.g. “the comfortable bed”, in which the adjective “comfortable” modifies the term “bed”) and *nominal subjects* (e.g., “The location of the hotel is perfect”, in which the term “location” is the subject of “perfect”). The distance-based approach applies a flexible strategy. That is, if the aspect-related term and the opinion co-occur in the same sentence, they are correlated. In our collected reviews, we notice that some users tend to write reviews in a rather free and unrestricted way. In other words, some sentences in reviews do not strictly follow standard syntactic patterns (such as the sentence “Everybody is there to make you

⁴ <http://nlp.stanford.edu/software/tagger.shtml>.

happy, from the owner to the chef in the kitchen.”) If we purely rely on standard patterns, opinions buried in these sentences cannot be captured. Therefore, we aggregate all of the opinions expressed in a single sentence for each aspect-related term using the distance-based technique: $score(s, f) = \sum_{op \in s} sent_{op} / d(op, f)$, where f denotes an aspect-related term that appears in sentence s , op denotes an opinion word in sentence s , $sent_{op}$ denotes its sentiment score, and $d(op, f)$ gives the distance from op to f (e.g., in “such a wonderful place”, the distance from “wonderful” to “place” is 1).

In addition, when performing this sub-step, we adopt two opinion rules (Ding et al. 2008): *Negation rule* (i.e., the opinion’s sentiment score will be reversed if there exists a negation word such as “no”, “not”, “never”, etc.) and *But rule* (i.e., the opinions expressed before and after the word “but” are opposite to each other).

- (3) *Context extraction* To reveal the underlying relation between aspects and contexts, not only the aspect-level opinions but also the contexts should be extracted from reviews. Following (Abowd et al. 1999), we regard context as any information that can be used to characterize the situation of an entity. For example, the contextual variables *Companion* (whether a user is accompanied by others) and *Occasion* (e.g., *anniversary*, *birthday*, etc.) have often been considered as important factors when the user is choosing a hotel to stay or a restaurant to take meals (Fuchs and Zanker 2012). In addition, for restaurant service, *Time* (i.e., time of taking the meal) has been regarded as an important contextual factor in influencing users’ choice (Li et al. 2010). Each contextual variable can be concretely assigned a value that we call “context value”. For example, the optional context values of *Companion* are *family*, *friends*, *colleague*, *couple*, *solo*, etc. Moreover, each context value can be defined by a set of keywords. For instance, the keywords related to the context value *colleague* are “colleague”, “business”, “coworker”, “boss”, and so on. Thus, if any of the keywords appear in a review sentence, that sentence will be labeled with the corresponding context value. Table 3 lists the contextual variables, context values, and value-related keywords for hotel and restaurant services.
- (4) *Aspect-context relation identification* From the above three sub-steps, we can obtain both aspect-level opinions and contextual information from reviews. The question then becomes how to determine their relations. We have observed two common patterns in user-generated reviews. (a) Users usually specify their context in the first sentence of the review, e.g., “We went to this restaurant for dinner,” “I chose this hotel for enjoying the holiday with my wife.” This observation is supported by a statistical analysis of our experiment datasets, i.e., 72.3 % of hotel reviews and 64.9 % of restaurant reviews contain this pattern. (b) In addition to stating context at the beginning, users may evaluate item aspects in another imagined context later in the text, such as “However, this hotel might not suit those enjoying a family trip due to its limited room space” (see Fig. 1). Statistically, 23.1 % of hotel reviews and 20.7 % of restaurant reviews in our datasets possess such a writing pattern. Based on these two observations, we propose the following rules for automatically identifying aspect-context relations: (a) if both an aspect-level opinion and a context occur in the same sentence, they

Table 3 Contextual variables, context values, and value-related keywords for hotel and restaurant services

Contextual variables	Context values	Value-related keywords
<i>Companion</i> (hotel, restaurant)	Couple	Wife, husband, girlfriend, boyfriend, spouse, honeymoon
	Friend	Friend
	Solo	Solo
	Colleague	Colleague, business, coworker, conference, boss
	Family	Family, children, child, son, daughter, mom, dad, hubby, father, mother, parent, kid, brother, sister
<i>Occasion</i> (hotel, restaurant)	Birthday	Birthday
	Anniversary	Anniversary
	Promotion	Promotion, coupon, discount
	Holiday	Holiday, vacation, festival, new year, Christmas, X'mas, thanksgiving, ester
<i>Time</i> (restaurant)	Breakfast	Breakfast, morning
	Lunch	Lunch, noon
	Brunch	Brunch
	Afternoon tea	Afternoon tea, afternoon drink
	Dinner	Dinner, supper, evening meal, night, evening

will be related; (b) if a sentence only contains an aspect-level opinion without mentioning a context, the opinion will be related to the context that appears in the previous, nearest sentence. Then, for a certain context, we sum all of the opinions pertinent to an aspect that is related to this context. That is, aspect k 's opinion a_k as contained in the tuple $\langle i, rev_{u,i}, a_k, Con_{u,i,k} \rangle$ is the result of aggregating all of the opinion scores of the aspect-related terms that are associated with the context $Con_{u,i,k}$. An aspect may be assigned to different opinion tuples in different contexts. For instance, the aspect "room" in the review presented in Figure 1 is contained in two tuples $\langle i, rev_{u,i}, a_{\text{room}} = 1, Con_{u,i,\text{room}} = \text{"business"} \rangle$ and $\langle i, rev_{u,i}, a_{\text{room}} = -1, Con_{u,i,\text{room}} = \text{"family"} \rangle$ ⁵, which have opposite opinions about this aspect in two different contexts.

4.2 Inferring context-independent preferences

As context-independent preferences reflect a user's relatively stable requirements for item aspects, the user's history data can be used to infer these preferences. To accomplish this task, we consider two alternative inference models: the linear regression model (LRM) and the probabilistic regression model (PRM).

⁵ For clarity, we use context value in this example, but it is formally represented as a boolean vector in our implementation (see example in Sect. 3).

4.2.1 Linear regression model based inference

This approach assumes that a user's overall evaluation of an item (like the overall rating) is the sum of her/his opinions about different aspects of the item, so it can be generated by aggregating the aspect-level opinions. For our purpose, the coefficient assigned to each aspect variable in the aggregation function can be interpreted as the weight that the user gives to that aspect; it essentially defines the relative contribution of the aspect to the overall rating.

More specifically, we apply the linear least-square regression function (Franklin 2005) to define this aggregation relationship. By performing contextual review analysis described in Sect. 4.1, each review written by a user can be represented as a rating vector $\mathbf{A}_{u,rev_{u,i}} = \langle a_1, \dots, a_K \rangle$, in which a_k ($1 \leq k \leq K$) represents the user's opinion rating for aspect k . All of the rating vectors (that correspond to the set of reviews written by the user) can then be used to construct the linear least-square regression function, which is formally denoted as:

$$R_{rev_{u,i}} = \mathbf{W}_u^T \mathbf{A}_{u,rev_{u,i}} + \varepsilon \quad (1)$$

where $R_{rev_{u,i}}$ denotes the overall rating that accompanies the review, ε denotes the error term, and $\mathbf{W}_u = \langle w_{u,1}, \dots, w_{u,K} \rangle$ denotes the weight vector that the user gives to different aspects.

As the obtained weights might not all be statistically significant, i.e., a_k has little influence on $R_{rev_{u,i}}$ and thus there is no significant linear relationship between a_k and $R_{rev_{u,i}}$, we apply the statistical t test to select weights that pass the significance level (0.1) and regard these weights as the user's context-independent preferences. To be specific, the *Null hypothesis* of the t test is that there is not a significant linear relationship between a_k and $R_{rev_{u,i}}$ and thus $w_{u,k}$ is equal to 0. Then, we calculate the test for each weight $w_{u,k}$ via $\frac{w_{u,k}}{\sqrt{\frac{1}{K} \sum_{k=1}^K (w_{u,k} - \varsigma)^2}}$, where $\varsigma = \frac{1}{K} \cdot \sum_{k=1}^K w_{u,k}$ denotes the mean of all of the acquired weights. If the corresponding p value is lower than the significance level that we set, we can reject the *Null hypothesis*, and conclude that the aspect k is important to the user and its associated weight reflects the user's preference for it.

4.2.2 Probabilistic regression model based inference

Like the linear regression model, the probabilistic regression model (PRM) postulates that the relation between the overall rating and all aspects' opinions is essentially a regression problem. However, the difference is that PRM models the underlying relation via Bayesian treatment so that prior knowledge can be incorporated into the model. Specifically, this approach considers the noise term ε in Eq. 1 is drawn from a Gaussian distribution, with a mean of 0 and a variance of σ^2 : $\varepsilon \sim \mathcal{N}(0, \sigma^2)$. Inspired by (Yu et al. 2011; Chen and Wang 2013), we treat the overall rating $R_{rev_{u,i}}$ as a sample drawn from a Gaussian distribution with a mean of $\mathbf{W}_u^T \mathbf{A}_{u,rev_{u,i}}$ and a variance of σ^2 . In other words, the conditional probability that a user u gives the overall rating $R_{rev_{u,i}}$ to an item i can be defined as follows:

$$\begin{aligned}
 p(R_{rev_{u,i}} | \mathbf{W}_u, \mathbf{A}_{u,rev_{u,i}}) &= \mathcal{N}(R_{rev_{u,i}} | \mathbf{W}_u^T \mathbf{A}_{u,rev_{u,i}}, \sigma^2) \\
 &= \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(R_{rev_{u,i}} - \mathbf{W}_u^T \mathbf{A}_{u,rev_{u,i}})^2}{2\sigma^2}\right) \quad (2)
 \end{aligned}$$

According to the Bayes theory (Franklin 2005), the posterior probability of \mathbf{W}_u can be defined as the product of Eq. 2 and the incorporated prior probability:

$$p(\mathbf{W}_u | \mathcal{S}) \propto \prod_{\langle u,i \rangle \in \mathcal{S}} p(R_{rev_{u,i}} | \mathbf{W}_u, \mathbf{A}_{u,rev_{u,i}}) \times p(\mathbf{W}_u | \mu, \Sigma) \times p(\mu, \Sigma) \quad (3)$$

where \mathcal{S} denotes the set of user-item pairs, in which $\langle u, i \rangle \in \mathcal{S}$ indicating that user u posted a review to item i , and $p(\mathbf{W}_u | \mu, \Sigma)$ is the prior probability of \mathbf{W}_u , which can be drawn from a multivariate Gaussian distribution with μ as the mean and Σ as the covariance matrix:

$$p(\mathbf{W}_u | \mu, \Sigma) \sim \mathcal{N}(\mu, \Sigma) \quad (4)$$

Given that important aspects are usually commented on more frequently by users, we concretely incorporate an aspect's occurrence frequency as the prior knowledge (denoted as μ_0) into $\mathcal{N}(\mu, \Sigma)$. The prior probability of the distribution $p(\mu, \Sigma)$ is hence defined as:

$$p(\mu, \Sigma) = \exp(-\psi \cdot KL(\mathcal{N}(\mu, \Sigma) | \mathcal{N}(\mu_0, \mathbf{I}))) \quad (5)$$

where $KL(\cdot | \cdot)$ is the KL-divergence for computing the difference between distributions $\mathcal{N}(\mu, \Sigma)$ and $\mathcal{N}(\mu_0, \mathbf{I})$, ψ is the trade-off parameter, and \mathbf{I} is an identity covariance matrix.

The parameters in the constructed model include $\Psi = \{\mathbf{W}_1, \dots, \mathbf{W}_{|U|}, \mu, \Sigma, \sigma^2\}$, in which U denotes the set of users. To estimate these parameters, we optimize the following function, which searches for an optimal $\hat{\Psi}$ to maximize the following probability given the review corpus:

$$\hat{\Psi} = \operatorname{argmax}_{\Psi} (\Psi | \mathcal{S}) = \sum_{\langle u,i \rangle \in \mathcal{S}} \log(p(\mathbf{W}_u | \mathcal{S})) \quad (6)$$

This can be solved by applying the Expectation–Maximization (EM) algorithm (Dempster et al. 1977). Thus, with PRM, we can also obtain the weights \mathbf{W}_u that a user holds for different aspects as the user's context-independent preferences.

4.2.3 Discussion

In our previous study (Chen and Chen 2014), we used only the linear regression model (LRM) to derive users' context-independent preferences, as the goal of that project was to improve recommendations for repeated users who are with abundant

history data. In such cases, the number of data samples used as input for training the regression model can be larger than or equal to the number of independent variables. In the current study, we consider new users who have few ratings and reviews, where the amount of user data does not satisfy the LRM's requirement. We hence believe that the probabilistic regression model (PRM) might address this limitation by treating the preference inference problem as a Bayesian learning process and considering the prior knowledge (i.e., the aspect's occurrence frequency). Therefore, in this study, we use PRM for new users, but we consider both LRM and PRM for repeated users, and compare their effectiveness in an experiment (see Sect. 5).

4.3 Inferring context-dependent preferences

Unlike context-independent preferences, context-dependent preferences indicate the aspect-level contextual needs that are common to users in the same context. To capture such preferences, we propose three variations of contextual weighting methods.

Intuitively, if an aspect appears more frequently than others in reviews pertaining to a certain context, this aspect should be more valued by users in this context and thus receive a higher weight. Therefore, our basic approach is to assign weights to aspects by analyzing the relation between the aspect's occurrence frequency and the context. We first develop the following formula to calculate the occurrence frequency of aspect k in context value c :

$$freq_{k,c} = \frac{\sum_{rev \in R} \sum_{s \in rev} \Delta_{s,c} \cdot \left(\sum_{f \in s} \Theta_{f,k} \right)}{\sum_{rev \in R} \sum_{s \in rev} \Delta_{s,c} \cdot \left(\sum_{f \in s} 1 \right)} \quad (7)$$

where f , s , and rev , respectively, represent an aspect-related term, a sentence, and a review; R denotes the set of all reviews; $\Delta_{s,c}$ is an indicator function, whose value is equal to 1 if the sentence s is related to context value c , and 0 otherwise; and $\Theta_{f,k}$ is another indicator function, whose value is equal to 1 if the term f is related to aspect k , and 0 otherwise. In fact, Eq. 7 calculates the aspect's occurrence frequency based on its related terms' occurrences in sentences labeled with context value c . Once the frequencies of the aspect in different context values are obtained, we compute the aspect's average frequency as $avg_k = \sum_{c \in \mathcal{C}} freq_{k,c} / |\mathcal{C}|$, the standard deviation as $stdv_k = \sqrt{\sum_{c \in \mathcal{C}} (freq_{k,c} - avg_k)^2 / |\mathcal{C}|}$ (where \mathcal{C} denotes the set of context values), and $dev_{k,c} = freq_{k,c} - avg_k$ (Levi et al. 2012). Next, we adopt the strategy proposed in (Levi et al. 2012) for computing the weight of aspect k regarding context value c :

$$w_{k,c} = \begin{cases} 1, & \text{if } |dev_{k,c}| < stdv_k \\ \text{Max} \left(0.1, 1 / \left| \frac{dev_{k,c}}{stdv_k} \right| \right), & \text{if } \frac{dev_{k,c}}{stdv_k} < -1 \\ \text{Min} \left(3, \frac{dev_{k,c}}{stdv_k} \right), & \text{else} \end{cases} \quad (8)$$

However, Eq. 7 does not distinguish the relative importance of the aspect-related term in different contexts. In our view, the same term might be valued differently by

users in different contexts, as explained by the example we gave in Sect. 3. To account for this, we extend this method (Eq. 8) by weighting the term using knowledge from text categorization (Yang and Pedersen 1997). We concretely build on the text categorization methods for selecting representative features (i.e., terms) when categorizing documents, and develop three contextual weighting methods; then we compare their effectiveness in an experiment.

4.3.1 Mutual information (MI)

Originally, mutual information was used to measure the mutual dependence between two random variables in information theory (Yang and Pedersen 1997). For our task, the two random variables can be *aspect-related term* and *context*. Given a term f and a context value c , the mutual information between them is defined as:

$$MI(f, c) = \log \frac{p(f \wedge c)}{p(f) \cdot p(c)} \quad (9)$$

where $p(f)$ denotes the probability of f appearing in sentences, $p(c)$ denotes the probability of sentences that are associated with context value c , and $p(f \wedge c)$ denotes the probability that f appears in sentences that are related to context value c .

4.3.2 Information gain (IG)

In the area of text categorization, information gain is used to measure the number of bits of information for categorizing documents by knowing the presence or absence of a word in a document (Yang and Pedersen 1997). Hence, we can use this metric to measure the importance of an aspect-related term within a specific context. To suit our need, we implement this metric as a binary classification model, in which each sentence is classified into two categories, *related to* context value c or *not*: $\mathcal{O} = \{c_{\text{presence}}, c_{\text{absence}}\}$. The information gain is then calculated as follows:

$$\begin{aligned} IG(f, c) = & - \sum_{c \in \mathcal{O}} p(c) \cdot \log p(c) + p(f) \sum_{c \in \mathcal{O}} p(c | f) \log p(c | f) \\ & + p(\bar{f}) \sum_{c \in \mathcal{O}} p(c | \bar{f}) \log p(c | \bar{f}) \end{aligned} \quad (10)$$

where \bar{f} denotes the absence of f in a sentence, and $p(c | f)$ denotes the probability that sentences containing f are related to context value c .

4.3.3 Chi-square statistic (CHI)

Generally speaking, we can measure the lack of independence between two random variables by computing the variance between the sample distribution and Chi-square distribution (Yang and Pedersen 1997). For our purpose, the lack of independence is computed between an aspect-related term f and a context value c , and regarded as f 's weight for c , which is formally defined as follows:

$$CHI(f, c) = \frac{D \times (D_1 D_4 - D_2 D_3)^2}{(D_1 + D_3) \times (D_2 + D_4) \times (D_1 + D_2) \times (D_3 + D_4)} \quad (11)$$

where D_1 is the number of times that f occurs in sentences related to c , D_2 is the number of times that f occurs in sentences not related to c , D_3 is the number of sentences in c that do not contain f , D_4 is the number of sentences that are neither related to c nor containing f , and D is the number of times that all of the terms occur in sentences related to c .

There are several inherent differences between the above three methods: (1) MI treats an aspect-related term and a context value as independent of each other and computes the dependency based on the probability of them co-occurring in a sentence, which is rather straightforward; (2) IG regards the reviews written in a context as a corpus, and computes the dependency as the amount of information (measured by applying the entropy theory) obtained by observing the aspect-related term in the corpus; and (3) like MI, CHI also assumes that the two random variables are independent of each other, but computes the dependency as the variance between the sample distribution and Chi-square distribution. The common property between IG and CHI is that they both consider a term as having presence and absence statuses in relation to a context value.

Through any of these three methods, we can obtain the weights of the aspect-related terms with respect to different context values; this information can then be used for computing an aspect's frequency by modifying Eq. 7 as follows:

$$freq_{k,c} = \frac{\sum_{rev \in R} \sum_{s \in rev} \Delta_{s,c} \cdot \left(\sum_{f \in s} \Theta_{f,k} \cdot MI(f, c) \right)}{\sum_{rev \in R} \sum_{s \in rev} \Delta_{s,c} \cdot \left(\sum_{f \in s} MI(f, c) \right)} \quad (12)$$

where $MI(f, c)$ is calculated via Eq. 9, which can be replaced with $IG(f, c)$ (Eq. 10) or $CHI(f, c)$ (Eq. 11). The results can then be applied to Eq. 8 to determine the aspect's weight in a certain context.

4.4 Generating recommendation

Considering that users' behavior can be influenced by both *context-independent* and *context-dependent preferences*, we implement a linear-regression-based method to combine both types of preferences when computing a score for review $rev_{v,i}$ (i.e., a review written by reviewer v for item i) for the target user u (suppose item i is unknown to u):

$$score(u, rev_{v,i}, T) = \sum_{(i, rev_{v,i}, a_k, Con_{v,i,k}) \in S(rev_{v,i})} \left(\prod_{c \in T} (1 + \alpha_{k,c} \cdot w_{k,c}) \right) \cdot w_{u,k} \cdot a_k \cdot g(Con_u, Con_{v,i,k}) \quad (13)$$

In Eq. 13, $w_{k,c}$ is the user u 's context-dependent preference for aspect k in context value c (derived via one of the three contextual weighting methods as proposed in

Sect. 4.3), $w_{u,k}$ is the user's context-independent preference for aspect k (see Sect. 4.2), a_k is the aspect k 's opinion score contained in the contextual opinion tuple $\langle i, rev_{v,i}, a_k, Con_{v,i,k} \rangle$, $S(rev_{v,i})$ is the set of contextual opinion tuples derived from $rev_{v,i}$, T contains the target user's current contexts, and Con_u denotes its vector form. The indicator function $g(Con_u, Con_{v,i,k})$ is defined as follows:

$$g(Con_u, Con_{v,i,k}) = \begin{cases} 1, & \text{if } Con_u \cdot Con_{v,i,k} \neq 0 \\ 0, & \text{else} \end{cases} \quad (14)$$

Equation 14 is used to ensure that only the aspect-level opinions pertinent to the target user's current contexts are taken into account.

The score of item i for user u is then calculated by averaging the scores of all of its reviews using the following formula:

$$score(u, i) = avg_{rev_{v,i} \in R(i)} [score(u, rev_{v,i}, T)] \quad (15)$$

where $R(i)$ denotes the set of reviews for item i . As Eq. 13 considers the target user's context-dependent and context-independent preferences, the predicted score of a review reflects its relevance to the target user. The higher the predicted score, the more interested the target user in the aspects mentioned in the review. So, if the average score across all of the reviews of an item is high, this item could be recommended to the target user. The top- N items with the highest scores are retrieved in our system. In the experiment, we set N as 5, 10, and 15.

It is worth noting that in Eq. 13, $\alpha_{k,c}$ is a combination parameter used to control the relative contributions of a user's *context-independent* and *context-dependent* preferences for aspect k in context value c , when computing a review's score. To automatically learn the parameter for each $\langle aspect, context \rangle$ pair, we propose a stochastic gradient descent learning method. As our task is to perform the top- N recommendation, we may use an objective function that measures the ranking error (i.e., items enjoyed by the target user are ranked below those not enjoyed by her/him) (Weston et al. 2011):

$$\sum_{u \in U} \sum_{i \in I^+} \sum_{\bar{i} \in I^-} L(F(score(u, \bar{i}) \geq score(u, i))) \quad (16)$$

Here, U denotes the set of users, I^+ denotes the set of items enjoyed by user u (i.e., *positive items*⁶), and I^- denotes the set of items not enjoyed by u (i.e., *negative items*). The computation of $score(u, i)$ is via Eq. 15, which involves the parameter $\alpha_{k,c}$ that we aim to learn. The indicator function $F(\tau)$ is equal to 1 if τ is true, and 0 otherwise. The function $L(\varrho)$ is used to convert the ranking error ϱ (i.e., $F(\tau)$) into a weight. There are two choices for $L(\varrho)$: 1) $L(\varrho) = \mathcal{H} \cdot \varrho$, in which \mathcal{H} denotes a constant; and 2) $L(\varrho) = \sum_{x=1}^{\varrho} \frac{1}{x}$. It has been demonstrated that the first choice optimizes the mean rank of the recommendation list, whereas the second optimizes the top of the ranked

⁶ In our work, these items are selected as those that received ratings above four stars (out of five).

list (Jason et al. 2012). For instance, given two items, if their true ranking positions are 1 and 100, respectively, the first choice tends to favor functions that rank them both at 50, whereas the second prefers functions that rank them at their true positions, which matches our aim of optimizing the top- N items' ranking in the recommendation list. We thus adopt the second choice for defining $L(q)$.

However, Eq. 16 is not continuous and thus not arbitrarily differentiable, which prevents us from applying the stochastic gradient descent based method to solve it. Inspired by (Weston et al. 2011), we add a margin to Eq. 16 and approximate it as follows:

$$\sum_{u \in U} \sum_{i \in I^+} \sum_{\tilde{i} \in I^-} L(F(1 + \text{score}(u, \tilde{i}) \geq \text{score}(u, i))) \cdot |1 + \text{score}(u, \tilde{i}) - \text{score}(u, i)| \quad (17)$$

Algorithm 1 Stochastic gradient descent algorithm for learning the combination parameters

```

1: Randomly initialize the combination parameters
2: repeat
3:   For user  $u$ , randomly pick a positive item  $i \in I^+$ 
4:   Compute  $\text{score}(u, i)$ 
5:   Initialize  $N = 0$ 
6:   repeat
7:     Randomly select an item from  $\{I^+ \cup I^-\}$ 
8:      $N = N + 1$ 
9:   until  $\text{score}(u, \tilde{i}) + 1 > \text{score}(u, i)$ 
10:  if  $\text{score}(u, \tilde{i}) + 1 > \text{score}(u, i)$  then
11:    Minimize the objective function (i.e., Eq. 17) by the gradient-based rule defined in Eq. 18
12:  end if
13:  if  $\|\alpha_c\| > \mathcal{H}$  then
14:    Regularize the learned parameter vector via Eq. 19
15:  end if
16: until The output of the objective function becomes stable

```

By doing this, we make the stochastic gradient descent learning method feasible for minimizing the objective function so as to learn the optimal combination parameter $\alpha_{k,c}$ for each $\langle \text{aspect}, \text{context} \rangle$ pair. Algorithm 1 sketches the general scheme of our developed method. Specifically, it works as follows. Before the learning process starts, the combination parameters are randomly initialized (line 1). In each iteration, for each positive item i enjoyed by user u , we calculate the corresponding ranking error, i.e., $F(1 + \text{score}(u, \tilde{i}) \geq \text{score}(u, i))$ (lines 5 ~ 9). Particularly, due to the large number of items in real-life datasets, the computation of the ranking error would be costly. Therefore, we adopt the following sampling approximation: for a positive item i , sample N items until a violation is found, i.e., $\text{score}(u, \tilde{i}) + 1 > \text{score}(u, i)$, and then approximate the ranking error with $|I^+ \cup I^-| / N$. Then, the ranking error is converted into a weight $L(|I^+ \cup I^-| / N)$ and used to adjust the value of $\alpha_{k,c}$ (that is used to compute $\text{score}(u, i)$ and $\text{score}(u, \tilde{i})$ via Eq. 15) in the direction in which we expect an improvement (lines 10 ~ 12):

$$\alpha_{k,c} \leftarrow \alpha_{k,c} + \lambda L \left(\left| I^+ \cup I^- \right| / N \right) \quad (18)$$

where $k \in [1, K]$, $c \in [1, C]$, and λ is used to control the learning rate and set as 0.02 in our experiment. To avoid over-fitting problem, in each step we need to ensure that the learned parameter vectors (i.e., $\alpha_c = \langle \alpha_{1,c}, \dots, \alpha_{K,c} \rangle$) are constrained as follows: $\|\alpha_c\| \leq \mathcal{H}$, where \mathcal{H} denotes a constant value and is set as 8 through experimental trials. If $\|\alpha_c\| > \mathcal{H}$, we carry out the following strategy as regularization (lines 13 ~ 15):

$$\alpha_c \leftarrow \mathcal{H} \alpha_c / \|\alpha_c\| \quad (19)$$

The algorithm stops when the difference between the objective function in two successive iterations is smaller than a pre-defined threshold (line 16). Then, the learned parameter $\alpha_{k,c}$ is incorporated into Eq. 13 for calculating the score of a review of the candidate item for the target user.

5 Experiment and results

5.1 Datasets and experiment setup

We use two real-life datasets to test our approach: the first is a dataset of hotel service crawled from *TripAdvisor*, and the second is a dataset of restaurant service from *Yelp* as published by the RecSys'13 challenge⁷. In both datasets, each textual review is accompanied by an overall rating ranging from 1 to 5 stars as assigned by the reviewer. To ensure that each review contains sufficient evaluation information and that each item receives sufficient reviews to be analyzed, we first perform the following cleaning procedure: (1) remove reviews that contain less than three sentences; (2) remove users that have posted only one review; and (3) remove items that have received less than 15 reviews. The descriptions of the two datasets are given in Table 4. Note that the data sparsity is defined as $1 - \frac{\# \text{ of reviews}}{\# \text{ of users} \times \# \text{ of items}}$. The whole sets of retrieved aspect-related terms for the two datasets are shown in Tables 5 and 6.

For the evaluation procedure, we adopt a widely used per-user evaluation scheme (Shani and Gunawardana 2011; Codina et al. 2013). That is, for each user, we randomly select a certain number of ratings that are above four stars (i.e., enjoyed items) as testing data, while the remaining ratings serve as training data. In the hotel dataset, the average number of ratings (and reviews) given by new users (i.e., users who have less than five history records Jamali and Ester 2009) is 2.37; it is 14.08 for repeated users. In the restaurant dataset, the average number of ratings (and reviews) given by new users is 2.80; it is 15.99 for repeated users. Therefore, in the experiment, for each new user we randomly select one rating as the testing data, but for each repeated user we randomly select three ratings. As for the target user's current contexts, in the hotel dataset, such context information is attained from the tested item's associated context (as provided by the user); in the restaurant dataset, because such information is not available, it

⁷ <http://recsys.acm.org/recsys13/recsys-2013-challenge/>.

Table 4 Descriptions of the two datasets

	Hotels (from <i>TripAdvisor</i>)	Restaurants (from <i>Yelp</i>)
# of reviews	357,113	237,077
# of users	30,039	23,152
# of items	11,405	11,485
Average reviews per user	11.89 ± 5.84	10.24 ± 20.40
Average reviews per item	31.31 ± 56.81	20.64 ± 38.95
Average sentences per review	6.04 ± 3.39	7.72 ± 4.99
Average aspects per sentence	1.77 ± 0.68	1.33 ± 0.42
Data sparsity	99.90 %	99.91 %

Table 5 The whole set of retrieved aspect-related terms for hotel dataset

Aspects	Retrieved aspect-related terms
Value	Value, price, money, quality, deal, hotel, package, rate, resort, budget, ticket, accommodation, amount, credit card, dollar, building, travel, discount, agent, luxury, corner
Location	Location, place, distance, station, shuttle, cab, taxi, subway, airport, attraction, shopping, block, bus, ride, metro, mall, train, bus stop, downtown, park, theater, strip, district, museum, transportation, quarter, tourist interest, heart, trolley, middle, square, sight-seeing
Room	Room, size, bathroom, bath, closet, kitchen, kitchenette, dryer, microwave, fridge, hotel, view, floor, shower, tv, stay, property, sink, screen, window, balcony, cafe maker, option, refrigerator, mirror, ceiling, water pressure, neighborhood
Cleanliness	Cleanliness, cleaning, smell, smoke, carpet, smoking, hallway, furniture, wall, air conditioner, elevator, hall, air conditioning, conditioner, stair, noise, neighbor, construction, sound, toilet, complaint, fan, maintenance, ceiling, heat, level
Sleep quality	Sleep, bed, bedroom, pillow, sofa, linen, sheet, suite, mattress, bedding, couch, living room, towel, unit, chair, apartment, lobby, experience, space, studio, choice, town, road, boutique, employee, comfort, neighborhood
Service	Service, staff, reception, check-in, checkout, bartender, valet, member, concierge, front desk, maid, clerk, doorman, question, smile, check, bellman, bell man, manager, attitude, direction, help, request, notch, care, information, person, arrival, suggestion, guy, customer, luggage, bag
Facility	Facility, wifi, pool, gym, business, internet, parking, conference room, swimming pool, casino, garage, area, center, fitness room, fee, internet access, traveler, spa, computer, connection, meeting, charge, activity, river, tub, grounds, pass, rooftop, slot, lounge, jacuzzi, machine, game, music, movie, beach, convention
Food quality	Food, drink, dish, wine, salad, restaurant, meal, bar, breakfast, pizza, buffet variety, court, shop, dinner, selection, snack, fruit, lunch, cereal, egg, cheese, juice, variety, coffee, bagel, pastry, waffle, cafe, menu, tea, beer, cocktail, downstairs, option, item, gift, cup, dining

Table 6 The whole set of retrieved aspect-related terms for restaurant dataset

Aspects	Retrieved aspect-related terms
Value	Value, price, money, bill, tip, dollar, cash, quality, portion, range, card, star, credit, charge, amount, coupon, cost
Food quality	Food, drink, dish, wine, salad, menu, dessert, steak, breakfast, chicken, pizza, pork, bread, shrimp, cheese, sandwich, meal, buffet, potato, drink, pasta, dinner, lunch, breakfast, flavor, soup, meat, beef, burger, fish, tasting, chocolate, pudding, egg, crab, rib, rice, fries, cake, cream, seafood, appetizer, lobster, plate, glass, mushroom, bean, bacon, onion, tomato chip, butter, sausage, salmon, vegetable, bottle, lamb, sauce, table, restaurant, experience, dining experience
Atmosphere	Atmosphere, ambiance, music, seat, seating, decor, window, room, wood, conversation, bar, lighting, sport, view, wall, ceiling, booth, tv, fountain, floor
Service	Service, staff, waiter, waitress, manager, bartender, server, owner, reservation, smile, wait, customer, table, attention, hostess, care, attitude, choice
Location	Location, place, street, downtown, parking, walk, block, mall, selection, local place, course, neighborhood, quarter

is simulated by performing a contextual analysis of the user's review for the tested item. All of the reported results are the averages of per-user evaluations; the Student's *t* Test (Smucker et al. 2007) is applied to compute the statistical significance of the differences between the compared methods.

The experiment is designed to answer the following questions: (1) how can we accurately infer the context-independent and context-dependent preferences of both new and repeated users? and (2) when the two types of preferences are combined to generate recommendations, is the stochastic gradient descent based method capable of learning the combination parameters? Accordingly, the experiment is divided into three parts: (1) apply the method to a sample group of new users to identify the ideal strategy for inferring their preferences; (2) apply the method to a sample group of repeated users to identify the ideal strategy for inferring their preferences; and (3) apply the method to the whole dataset to test the effectiveness of the proposed stochastic gradient descent learning method. The results are given in Sect. 5.4.

5.2 Compared methods

The variations of our recommendation algorithm are denoted as LRM/PRM + MI/IG/CHI connector, which include different combinations of users' context-independent preferences (inferred by either the linear regression model (LRM) or probabilistic regression model (PRM); see Sect. 4.2) and context-dependent preferences (inferred by one of the three contextual weighting methods respectively based on mutual information (MI), information gain (IG), and chi-square statistics (CHI); see Sect. 4.3).

We compared our algorithms with three related methods: the first does not consider contextual information when generating recommendations (i.e., context freer); the second only uses contextual information to filter data before a traditional recommendation algorithm is applied (i.e., context pre-filter); and the third uses reviews to infer users' contextual preferences, but does not take into account aspect-related terms' relative importance (i.e., simple connector).

- **Context freer** this method adopts the regression-based method proposed in (Adomavicius and Kwon 2007); it uses the aspect-level opinions, i.e., $\{a_k \mid 1 \leq k \leq K\}$ (where K denotes the number of aspects), to calculate the score of a review $rev_{v,i}$ for the target user u . In fact, this method uses a simplified version of Eq. 13, which does not consider the user's context-dependent preferences:

$$score(u, rev_{v,i}) = \sum_{k=1}^K a_k \cdot w_{u,k} \quad (20)$$

Here, $w_{u,k}$ represents the user u 's context-independent preference for aspect k . Then, Eq. 15 is applied to compute the score of item i for user u . We denote this method as *Freer*.

- **Context Pre-filter** following (Adomavicius et al. 2005), the contextual information is used at the item level in this method. That is, we first pre-filter the ratings according to the user-specified contexts and then apply the recommendation algorithm *Freer*. This results in a modified version of Eq. 13 as follows:

$$score(u, rev_{v,i}) = g(Con_u, Con_{rev_{v,i}}) \cdot \sum_{k=1}^K a_k \cdot w_{u,k} \quad (21)$$

Here, $Con_{rev_{v,i}}$ denotes the contexts extracted from review $rev_{v,i}$, Con_u denotes the contexts specified by the target user, and $g(Con_u, Con_{rev_{v,i}})$ is an indicator function used to ensure that only the opinions pertinent to the target user's current contexts are considered (as defined in Eq. 14). In fact, this method only considers reviews written in the target user's contexts when calculating the item's score via Eq. 15. We denote it as *Pre-filter*.

- **Simple connector** this method originates from (Levi et al. 2012). It uses the results of the contextual review analysis that we described in Sect. 4.1 to assign context-dependent weights to aspects by Eq. 8. Compared to our approaches, this method does not consider the relative weights of aspect-related terms in different contexts. We denote it as *Simpler*.

5.3 Evaluation metrics

For the top- N recommendations, researchers have commonly stressed two points (Deshpande and Karypis 2004; Gunawardana and Shani 2009): whether a user's target choice appears in the recommendation list and how highly the target choice is ranked in the list. Therefore, we apply two metrics to measure the recommendation accuracy.

- **Hit Ratio @ top- N recommendations** shortened to $H@N$, measures whether a user's target choice appears in the top- N recommendations list (Chen and Wang 2013). It is computed as the percentage of hits among all users:

$$H@N = \sum_{z=1}^Z \delta_{rank_z \leq N} / Z \quad (22)$$

where Z is the number of testings, $rank_z$ is the ranking position of the user's target choice in the z -th testing, and $\delta_{rank_z \leq N}$ is an indicator function that is equal to 1 if $rank_z \leq N$ (i.e., the recommendation list contains the choice), or 0 otherwise.

- **Mean reciprocal rank** shortened to MRR , evaluates the ranking position of a user's target choice in the recommendation list (Chen and Wang 2013), and is formally defined as:

$$MRR = \sum_{z=1}^Z \frac{\delta_{rank_z \leq N}}{rank_z} / Z \quad (23)$$

5.4 Results analysis

As mentioned, we divide our experiment into three parts: experiments respectively on new users, repeated users, and the whole dataset. Notice that for each sample, we can determine the combination parameter α for generating recommendations (via Eqs. 13–15) by applying either of the two methods: (1) *manual selection*, which manually selects the best parameter value based on experimental trials (Chen and Chen 2014); or (2) *automatic selection*, which automatically decides the parameter value for each $\langle aspect, context \rangle$ pair through applying the stochastic gradient descent learning algorithm we proposed in Sect. 4.4. In the first two experiment parts, because our main goal is to investigate the best strategies for inferring context-independent and context-dependent preferences for new users and repeated users respectively, we simply use the first strategy. In the third part, we focus on the second selection strategy to investigate whether our recommendation algorithm can be further improved.

5.4.1 Evaluation of preference inference for new users

As new users' context-independent preferences can only be estimated by applying the probabilistic regression model (PRM) (see the discussion in Sect. 4.2), we primarily compare the three different contextual weighting methods (i.e., MI-based, IG-based, and CHI-based), which differ in terms of how they detect context-dependent preferences. That is, there are three variations of the method for preference inference for new users: $PRM + MI$, $PRM + IG$, and $PRM + CHI$. The experiment results are shown in Table 7.

First, we observe that both *Pre-filter* and *Simpler* significantly defeat *Freer* with respect to the two evaluation metrics. For instance, the $H@5$ achieved by *Simpler* is

Table 7 Experiment results of preference inference for new users. Results marked with * are significantly better than the method being compared ($p < 0.001$ by Student's t Test)

Dataset	Method	H@5	H@10	H@15	MRR@5	MRR@10	MRR@15
Hotel	<i>PRM based Freer</i>	0.0031	0.0068	0.0093	0.0013	0.0017	0.0019
	<i>PRM based Pre-filter</i>	0.0060*	0.0101*	0.0138*	0.0027*	0.0032*	0.0035*
	<i>PRM based Simpler</i>	0.0093*	0.0155*	0.0194	0.0048*	0.0057*	0.0060*
	<i>PRM + MI</i>	0.0193*	0.0316*	0.0448*	0.0106*	0.0122*	0.0133*
	<i>PRM + IG</i>	0.0200*	0.0313*	0.0480*	0.0109*	0.0126*	0.0138*
	<i>PRM + CHI</i>	0.0325*	0.0512*	0.0630*	0.0158*	0.0162*	0.0212*
Restaurant	<i>PRM based Freer</i>	0.0045	0.0099	0.0133	0.0014	0.0021	0.0024
	<i>PRM based Pre-filter</i>	0.0077*	0.0132*	0.0177*	0.0031*	0.0039*	0.0042*
	<i>PRM based Simpler</i>	0.0107	0.0131*	0.0314*	0.0060*	0.0071*	0.0080*
	<i>PRM + MI</i>	0.0265*	0.0476*	0.0711*	0.0143*	0.0172*	0.0191*
	<i>PRM + IG</i>	0.0312*	0.0587*	0.0867*	0.0148	0.0195*	0.0217*
	<i>PRM + CHI</i>	0.0304	0.0707*	0.0903*	0.0179*	0.0220*	0.0235*

Here, the significance values are calculated between *PRM based Pre-filter* and *PRM based Freer*, between *PRM based Simpler* and *PRM based Pre-filter*, and between *PRM + MI/IG/CHI* and *PRM based Simpler*

0.0093 and the one achieved by *Pre-filter* is 0.0060 in the hotel dataset; these values are, respectively, 200 and 94 % higher than that achieved by *Freer* that does not consider contextual information. Similar improvements are observed for the other evaluation metrics. This proves that contextual information as extracted from reviews can enhance recommendation. Moreover, the comparison between *Pre-filter* and *Simpler* shows that, in most cases, *Simpler* is better than *Pre-filter*, indicating that contextual opinions can be used to build users' aspect-level context-dependent preferences.

The results also show that *PRM + MI/IG/CHI* is significantly superior to *Simpler* with respect to all of the evaluation metrics in most conditions. For example, the improvements brought by *MI*, *IG*, and *CHI* over *Simpler* in terms of H@5 in the hotel dataset are, respectively, 109, 116 and 251 %; for MRR@5, they are 120, 126 and 227 %. This demonstrates that the accuracy of users' aspect-level context-dependent preferences can be increased by considering the relative importance of aspect-related terms in different contexts. Among the three contextual weighting methods, *CHI* achieves the best performance, followed by *IG* and then *MI*. This can be explained by the way in which these methods compute the relevance of an aspect-related term to a specific context. *MI* (i.e., Eq. 9) tends to favor low-frequency terms, which might result in biases in the calculation of a term's relevance. In comparison, both *CHI* (i.e., Eq. 11) and *IG* (i.e., Eq. 10) compute a term's weight by considering all of the possible combinations of "presence" and "absence" statuses of an aspect-related term in relation to a specific context. The better performance obtained by *CHI* relative to *IG* is likely because *CHI* computes the dependency between an aspect-related term and a context value by directly measuring their co-occurrence frequency; this depicts the term's relative importance more precisely and thus results in better inference accuracy.

Table 8 Experiment results of preference Inference for repeated users on the **hotel** dataset

Method	H@5	H@10	H@15	MRR@5	MRR@10	MRR@15
<i>LRM based Freer</i>	0.0125	0.0191	0.0269	0.0065	0.0073	0.0079
<i>LRM based pre-filter</i>	0.0226	0.0398*	0.0554*	0.0115*	0.0137*	0.0149*
<i>LRM based simpler</i>	0.0242*	0.0459*	0.0544*	0.0117*	0.0146*	0.0152*
<i>LRM + MI</i>	0.0334*	0.0541*	0.0647*	0.0193*	0.0221*	0.0259*
<i>LRM + IG</i>	0.0439*	0.0536*	0.0975*	0.0231*	0.0212*	0.0358*
<i>LRM + CHI</i>	0.0565*	0.0806*	0.1131*	0.0217	0.0425*	0.0565*
<i>PRM based Freer</i>	0.0156 \diamond	0.0253 \diamond	0.0364 \diamond	0.0097 \diamond	0.0099 \diamond	0.0108 \diamond
<i>PRM based Pre-filter</i>	0.0232* \diamond	0.0434* \diamond	0.0569* \diamond	0.0113* \diamond	0.0139* \diamond	0.0150* \diamond
<i>PRM based Simpler</i>	0.0340* \diamond	0.0557* \diamond	0.0740* \diamond	0.0145* \diamond	0.0194* \diamond	0.0250* \diamond
<i>PRM + MI</i>	0.0434*	0.0640* \diamond	0.0848*	0.0213* \diamond	0.0240* \diamond	0.0249* \diamond
<i>PRM + IG</i>	0.0535* \diamond	0.0733* \diamond	0.1067* \diamond	0.0300* \diamond	0.0347* \diamond	0.0377* \diamond
<i>PRM + CHI</i>	0.0766*\diamond	0.1005*\diamond	0.1537*\diamond	0.0340*\diamond	0.0459*\diamond	0.0599*\diamond

Results marked with * or \diamond are significantly better than the method being compared ($p < 0.001$ by Student's t test). Here, the significance values marked with * are calculated between *LRM/PRM based Pre-filter* and *LRM/PRM based Freer*, between *LRM/PRM based Simpler* and *LRM/PRM based Pre-filter*, between *LRM/PRM + MI/IG/CHI* and *LRM/PRM based Simpler*; and those marked with \diamond are calculated between *PRM based Freer/Pre-filter/Simpler* and *LRM based Freer/Pre-filter/Simpler*, between *PRM + MI/IG/CHI* and *LRM + MI/IG/CHI*

5.4.2 Evaluation of preference inference for repeated users

The context-independent preferences of repeated users can be learned by applying either the linear regression model (LRM) or the probabilistic regression model (PRM), and their context-dependent preferences can be learned by applying one of the MI-, IG- and CHI-based contextual weighting methods. Therefore, there are six different combinations to be tested: *LRM + MI*, *LRM + IG*, *LRM + CHI*, *PRM + MI*, *PRM + IG*, and *PRM + CHI*. The experiment results are reported in Tables 8 (on the hotel dataset) and 9 (on the restaurant dataset).

The results are similar to those presented in Table 7. That is, of the three methods, *Freer*, *Pre-filter*, and *Simpler*, *Simpler* still achieves the best performance, followed by *Pre-filter* and then *Freer*. This supports our postulation that reviews are valuable resources, containing contextual opinions that can be used to more accurately depict users' aspect-level contextual preferences. In addition, all of our proposed context-dependent preference inference methods defeat *Simpler*, and *CHI* still performs the best. Comparing these results with those reported in Table 7, we find that the improvement brought by discriminating aspect-related terms (e.g., *CHI* over *Simpler*) is more obvious for new users than for repeated users in terms of the metric H@N ($N = 5, 10, 15$). Specifically, in the hotel dataset, the average improvement is up to 236 % for new users, but only 105 % for repeated users; in the restaurant dataset, the improvement is 295 % for new users vs. 75 % for repeated users.

For the context-independent preference inference for repeated users, *PRM based Freer*, which only considers users' context-independent preferences as inferred by

Table 9 Experiment results of preference Inference for repeated users on the **restaurant** dataset

Method	H@5	H@10	H@15	MRR@5	MRR@10	MRR@15
<i>LRM based Freer</i>	0.0131	0.0249	0.0335	0.0067	0.0082	0.0089
<i>LRM based Pre-filter</i>	0.0189*	0.0325*	0.0485*	0.0094*	0.0112*	0.0124*
<i>LRM based Simpler</i>	0.0338*	0.0550*	0.0775*	0.0178*	0.0206*	0.0224*
<i>LRM + MI</i>	0.0335*	0.0558*	0.0811*	0.0178*	0.0207*	0.0226*
<i>LRM + IG</i>	0.0463*	0.0702*	0.1154*	0.0183*	0.0325*	0.0445
<i>LRM + CHI</i>	0.0679*	0.0773*	0.1331*	0.0282*	0.0394*	0.0554*
<i>PRM based Freer</i>	0.0185 \diamond	0.0309 \diamond	0.0449 \diamond	0.0092 \diamond	0.0107 \diamond	0.0118 \diamond
<i>PRM based pre-filter</i>	0.0224*	0.0407* \diamond	0.0582* \diamond	0.0123* \diamond	0.0147* \diamond	0.0161* \diamond
<i>PRM based simpler</i>	0.0363* \diamond	0.0631* \diamond	0.0896*	0.0186* \diamond	0.0222* \diamond	0.0242* \diamond
<i>PRM + MI</i>	0.0397* \diamond	0.0723* \diamond	0.0967*	0.0194* \diamond	0.0235* \diamond	0.0255* \diamond
<i>PRM + IG</i>	0.0474* \diamond	0.0722* \diamond	0.1291* \diamond	0.0240* \diamond	0.0329* \diamond	0.0448* \diamond
<i>PRM + CHI</i>	0.0686*\diamond	0.1066*\diamond	0.1495*\diamond	0.0284*\diamond	0.0434*\diamond	0.0559*\diamond

Results marked with * or \diamond are significantly better than the method being compared ($p < 0.001$ by Student's t Test). Note that the significance values are calculated in the same way as in Table 8

PRM, significantly outperforms *LRM based Freer*, in terms of all of the evaluation metrics in both datasets. For example, in the hotel dataset, the value of *PRM based Freer* is 35 % higher than that achieved by *LRM based Freer* w.r.t. H@15; it is 37 % higher w.r.t. MRR@15. Furthermore, when users' context-dependent preferences are integrated, the PRM-based variations (i.e., PRM based *Pre-filter*, PRM based *Simpler*, PRM + MI/IG/CHI) are significantly superior to those based on LRM in terms of most evaluation metrics. All of these results demonstrate that PRM is better at deriving repeated users' context-independent preferences, which may be because it incorporates prior knowledge into the model.

5.4.3 Evaluation of combination parameter

The results from the above two parts lead to the following conclusions: for both types of users, i.e., new users and repeated users, context-independent preferences are better estimated by applying the probabilistic regression model (PRM), and context-dependent preferences are better obtained through the contextual weighting method based on Chi-square statistic. In this part, we focus on investigating the effectiveness of our proposed stochastic gradient descent learning algorithm, which is aimed at automatically determining a set of combination parameters when generating recommendations (see Sect. 4.4). Specifically, we seek to learn the parameter for each $\langle \text{aspect}, \text{context} \rangle$ pair when combining the context-independent and context-dependent preferences via Equation 13. Therefore, there are $K \times C$ parameters to be learned, i.e., $\{\alpha_{k,c} \mid 1 \leq k \leq K, 1 \leq c \leq C\}$. In addition, we implement some variations of the learning method that combine the two types of preferences at different levels (i.e., holistic-level and aspect-level):

Table 10 Experiment results of the combination parameter identification

Dataset	Method	H@5	H@10	H@15	MRR@5	MRR@10	MRR@15
Hotel	<i>Holistic</i>	0.0457	0.0694	0.0892	0.0217	0.0315	0.0348
	<i>Aspect</i>	0.0512	0.0860*	0.1286*	0.0348*	0.0433*	0.0505*
	<i>Aspect-Context</i>	0.0677*	0.1184*	0.1457*	0.0411*	0.0523*	0.0599*
Restaurant	<i>Holistic</i>	0.0432	0.0883	0.1197	0.0209	0.0329	0.0453
	<i>Aspect</i>	0.0592*	0.1123*	0.1298*	0.0314*	0.0367*	0.0510*
	<i>Aspect-Context</i>	0.0753*	0.1378*	0.1601*	0.0389*	0.0466*	0.0628*

Results marked with * are significantly better than the method being compared ($p < 0.001$ by Student's t test). Here, the significance values are calculated between *Aspect* and *Holistic*, and between *Aspect-Context* and *Aspect*

- **Holistic learning** searches for a holistic-level parameter α manually through experimental trials, as described in previous study (Coy et al. 2001). In other words, the combination parameter is neither aspect-specific nor context-specific, thus it cannot be adaptive to a user's needs for different aspects of an item in different contexts. $\alpha_{k,c}$ in Eq. 13 is replaced with a fixed parameter α in this method. We denote it as *Holistic*.
- **Aspect-level learning** involves K parameters, i.e., $\alpha = \langle \alpha_1, \dots, \alpha_K \rangle$, in which α_k ($1 \leq k \leq K$) represents the combination parameter for aspect k in all of the contexts. For this learning, Eq. 18 is modified as $\alpha_k \leftarrow \alpha_k + \lambda L(|I^+ \cup I^-|/N)$ and Eq. 19 is modified as $\alpha_k \leftarrow \mathcal{H}\alpha_k/\|\alpha\|$. In Eq. 13, α_k replaces $\alpha_{k,c}$ for computing a review's score. This method does not consider that users' context-independent and context-dependent preferences for the same aspect can be combined in different ways in different contexts. We denote it as *Aspect*.
- **Aspect-context-level learning** learns a parameter $\alpha_{k,c}$ for each $\langle \text{aspect}, \text{context} \rangle$ pair, as described in Sect. 4.4. We denote it as *Aspect-Context*.

The experiment results are shown in Table 10. There are two important findings: (1) *Aspect* is significantly superior to *Holistic* in most conditions, which demonstrates the effectiveness of the proposed learning algorithm in terms of combining two types of user preferences via learning the combination parameter at a more fine-grained level, i.e., learning a parameter for each aspect. For instance, the H@10 achieved by *Aspect* is 0.0860 in the hotel dataset, which is 24 % higher than that achieved by *Holistic*. As for the restaurant dataset, the performance of *Aspect* is 27 % higher than that achieved by *Holistic* with respect to H@10 (i.e., 0.1123 vs. 0.0883); and (2) *Aspect-Context* further defeats *Aspect*. The average improvement brought by *Aspect-Context* is up to 28 % (relative to *Aspect*) and it is up to 61 % (relative to *Holistic*) in the hotel dataset in terms of metrics H@N ($N = 5, 10, 15$), and the improvements are respectively 24 and 55 % in the restaurant dataset. This proves our hypothesis that users' aspect-level preferences can be influenced by contexts, and that our proposed learning algorithm is capable of capturing such influences.

6 Discussion

6.1 Summary of experiment results

All of the above results lead to three main conclusions. (1) The probabilistic regression model (PRM) is suitable for deriving not only new users' but also repeated users' context-independent preferences; this can be mainly attributed to its Bayesian learning process and the prior knowledge that it can draw on when deriving such preferences. (2) For detecting users' context-dependent preferences, the contextual weighting method based on CHI defeats not only the baselines, but also the other two weighting methods respectively based on MI and IG under most circumstances. Its advantage can be attributed to two main properties of CHI: (a) CHI considers all possible combinations of the statuses (i.e., "*presence*" and "*absence*") of an aspect-related term in relation to a context value; and (b) CHI calculates the dependency between an aspect-related term and a context value by directly measuring their co-occurrence frequency. (3) The stochastic gradient descent learning method can automatically learn the combination parameters for fusing the two types of user preferences, and hence enhance the recommendation accuracy.

6.2 Practical implications to recommendation in ubiquitous computing

In our view, this research brings several practical implications to recommendation in ubiquitous computing. (1) With the aid of advanced mobile devices (e.g., smart phones, Google glass), users' current contexts (such as location, motion, time of day) can be automatically sensed (Carmichael et al. 2005; Zimmermann et al. 2005; Cheverst et al. 2005; Hammer et al. 2015); and more importantly, such contexts can be matched to their contextual preferences that are inferred from their item reviews for system to provide accurate recommendations in real time. (2) In particular, we have found that reviews can be used to model users' preferences at fine-grained aspect level, which are then linked to contexts for capturing their multi-faceted nature of contextual needs for items. (3) Moreover, through experiment on hotel and restaurant datasets, we have demonstrated the actual merit of our recommendation algorithm in mobile tourism, which is a typical application scenario of ubiquitous computing (Hatala and Wakkary 2005; Petrelli and Not 2005).

6.3 Limitations of our current work

Our current work still has several limitations. (1) The experiment was conducted on only two datasets, which limits the generalizability and applicability of our findings to broader product domains. Moreover, the practical usefulness of our method in real life is not tested, as the experiment was designed as an offline simulation and the approach has not been validated as effective for online users. (2) In the experiment, we excluded short reviews and items with few reviews to ensure that each review possesses sufficient opinions and that each item has received sufficient reviews. However, since they may also contain some valuable information, our method should be improved to

accommodate their special characteristics. (3) In our collected reviews, we observed that sentences like “*This place is a wonderful choice for family or friends to gather, but not for a couple*” offer both positive and negative opinions about an aspect in different contexts. Using our current aspect-context relation identification method (see Sect. 4.1—step four), which correlates an aspect-level opinion with all of the contexts expressed in a sentence, we cannot identify the negative opinion about “place” in the context *couple* in the above example. In addition, considering that adverbs can be used as intensifiers to strengthen or soften opinions, they could be treated in a different way from adjectives in the process of determining opinion orientation. (4) Our current recommendation algorithm simply averages all of an item’s reviews’ scores to predict the item’s interest score for the target user, which is irrespective of the number of reviews being aggregated.

7 Conclusions and future work

In this paper, we seek to enhance service recommender systems by leveraging users’ aspect-level contextual preferences (i.e., *context-dependent preference*). For this purpose, we have investigated three variations of contextual weighting methods which are based on different text feature selection strategies: MI, IG and CHI. All of the three strategies aim to analyze the relation between an aspect’s frequency (based on aspect-related terms’ relative importance) and a context value. We further derive users’ *context-independent preferences* from reviews. Particularly, to support both new users and repeated users, we have investigated two regression models for deriving context-independent preferences: the linear regression model (LRM) and the probabilistic regression model (PRM). Then, we proposed a linear-regression-based algorithm that uses a stochastic gradient descent learning procedure to automatically fuse the two types of preferences into the process of generating recommendations.

We tested the proposed method on two real-life service datasets and demonstrated that our method outperforms related techniques in terms of recommendation accuracy. In summary, we have found that (1) it is helpful to correlate users’ opinions with contextual factors by performing contextual review analysis; (2) the accuracy of a user’s profile can be increased by combining both context-dependent and context-independent preferences; and (3) aspect-related terms are important for discriminating users’ aspect-level preferences in different contexts. Thus, our work highlights the merit of deriving users’ aspect-level contextual preferences from reviews, and the effect of our proposed linear-regression-based algorithm on improving the recommendation accuracy. As mentioned above (Sect. 6.2), we believe that our algorithm can be beneficial to recommender systems in ubiquitous computing. In this scenario, the system can automatically sense a user’s current contexts through her/his mobile devices and then match the contexts to her/his aspect-level contextual preferences (as derived from her/his reviews to items such as hotels, restaurants) for the production of accurate recommendations in real time.

In the future, we will continue to improve our approach as follows. (1) We will conduct user evaluations to empirically validate the practical benefits of our recommendation algorithm to online users. (2) We will address the limitations of our method

(as discussed in Section 6.3). For instance, we may combine a matrix factorization model with LDA (McAuley and Leskovec 2013) for processing items with few reviews. We may improve the accuracy of aspect-context relation identification through adopting some specific linguistic rules (Ding et al. 2008). We will also take into account the number of reviews when aggregating them to compute an item's prediction score. (3) It will be interesting to investigate reviewers' aspect-level comparative opinions, such as "*The bed was comfortable but not as good as that in the Four Seasons Hotel*" (Zhang et al. 2010). Intuitively, comparative opinions can reveal users' preferences for one item over others with regard to some aspects; this motivates us to combine them with contextual opinions for further improving our recommendation algorithm.

Acknowledgments The reported work was supported by Hong Kong Research Grants Council (no. ECS/HKBU211912) and China National Natural Science Foundation (no. 61272365).

References

- Abowd, G.D., Dey, A.K., Brown, P.J., Davies, N., Smith, M., Steggles, P.: Towards a better understanding of context and context-awareness. In: HUC '99, Proceedings of the First International Symposium on Handheld and Ubiquitous Computing, Springer, Karlsruhe, pp. 304–307 (1999). <http://dl.acm.org/citation.cfm?id=647985.743843>
- Acar, E., Dunlavy, D.M., Kolda, T.G., Mørup, M.: Scalable tensor factorizations for incomplete data. *Chemom. Intell. Lab. Syst.* **106**(1), 41–56 (2011)
- Adomavicius, G., Kwon, Y.: New recommendation techniques for multicriteria rating systems. *IEEE Intell. Syst.* **22**(3), 48–55 (2007). doi:[10.1109/MIS.2007.58](https://doi.org/10.1109/MIS.2007.58)
- Adomavicius, G., Tuzhilin, A.: Context-aware recommender systems. In: Ricci, F., Rokach, L., Shapira, B., Kantor, P.B. (eds.) *Recommender Systems Handbook*, pp. 217–253. Springer, New York (2011). doi:[10.1007/978-0-387-85820-3-7](https://doi.org/10.1007/978-0-387-85820-3-7)
- Adomavicius, G., Sankaranarayanan, R., Sen, S., Tuzhilin, A.: Incorporating contextual information in recommender systems using a multidimensional approach. *ACM Trans. Inf. Syst.* **23**(1), 103–145 (2005). doi:[10.1145/1055709.1055714](https://doi.org/10.1145/1055709.1055714)
- Adomavicius, G., Manouselis, N., Kwon, Y.: Multi-criteria recommender systems. In: Ricci, F., Rokach, L., Shapira, B., Kantor, P.B. (eds.) *Recommender Systems Handbook*, pp. 769–803. Springer, New York (2011). doi:[10.1007/978-0-387-85820-3-24](https://doi.org/10.1007/978-0-387-85820-3-24)
- Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. *J. Mach. Learn. Res.* **3**, 993–1022 (2003). <http://dl.acm.org/citation.cfm?id=944919.944937>
- Carmichael, D., Kay, J., Kummerfeld, B.: Consistent modelling of users, devices and sensors in a ubiquitous computing environment. *User Model. User-Adapt. Interact.* **15**(3–4), 197–234 (2005). doi:[10.1007/s11257-005-0001-z](https://doi.org/10.1007/s11257-005-0001-z)
- Carter, S., Chen, F., Muralidharan, A.S., Pickens, J.: Dig: a task-based approach to product search. In: IUT' 11, Proceedings of the Sixteenth International Conference on Intelligent User Interfaces, pp. 303–306. ACM, Palo Alto (2011). doi:[10.1145/1943403.1943451](https://doi.org/10.1145/1943403.1943451)
- Chen, G., Chen, L.: Recommendation based on contextual opinions. In: Houben, G.-J., et al. (eds.) *User Modeling, Adaptation, and Personalization. Lecture Notes in Computer Science*, vol. 8538, pp. 61–73. Springer International Publishing, Aalborg (2014). doi:[10.1007/978-3-319-08786-3-6](https://doi.org/10.1007/978-3-319-08786-3-6)
- Chen, L., Wang, F.: Preference-based clustering reviews for augmenting e-commerce recommendation. *Knowl. Based Syst.* **50**, 44–59 (2013). doi:[10.1016/j.knosys.2013.05.006](https://doi.org/10.1016/j.knosys.2013.05.006)
- Cheverst, K., Byun, H., Fitton, D., Sas, C., Kray, C., Villar, N.: Exploring issues of user model transparency and proactive behaviour in an office environment control system. *User Model. User-Adapt. Interact.* **15**(3–4), 235–273 (2005). doi:[10.1007/s11257-005-1269-8](https://doi.org/10.1007/s11257-005-1269-8)
- Codina, V., Ricci, F., Ceccaroni, L.: Exploiting the semantic similarity of contextual situations for pre-filtering recommendation. In: Carberry, S., et al. (eds.) *User Modeling, Adaptation, and Personalization. Lecture Notes in Computer Science*, vol. 7899, pp. 165–177. Springer, Berlin (2013). doi:[10.1007/978-3-642-38844-6-14](https://doi.org/10.1007/978-3-642-38844-6-14)

- Coy, S., Golden, B., Runger, G., Wasil, E.: Using experimental design to find effective parameter settings for heuristics. *J. Heuristics* **7**(1), 77–97 (2001). doi:[10.1023/A:1026569813391](https://doi.org/10.1023/A:1026569813391)
- Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the EM algorithm. *J. R. Stat. Soc.* **39**(1), 1–38 (1977)
- Deshpande, M., Karypis, G.: Item-based top-n recommendation algorithms. *ACM Trans. Inf. Syst.* **22**(1), 143–177 (2004). doi:[10.1145/963770.963776](https://doi.org/10.1145/963770.963776)
- Ding, X., Liu, B., Yu, P.S.: A holistic lexicon-based approach to opinion mining. In: *WSDM '08, Proceedings of the 2008 International Conference on Web Search and Data Mining*, pp. 231–240. ACM, Palo Alto (2008) doi:[10.1145/1341531.1341561](https://doi.org/10.1145/1341531.1341561)
- Dong, R., O'Mahony, M.P., Schaal, M., McCarthy, K., Smyth, B.: Sentimental product recommendation. In: *RecSys '13, Proceedings of the 7th ACM Conference on Recommender Systems*, pp. 411–414. ACM, Hong Kong (2013a). doi:[10.1145/2507157.2507199](https://doi.org/10.1145/2507157.2507199)
- Dong, R., Schaal, M., O'Mahony, M., McCarthy, K., Smyth, B.: Opinionated product recommendation. In: Delany, S., Ontanon, S. (eds.) *Case-Based Reasoning Research and Development. Lecture Notes in Computer Science*, vol. 7969, pp. 44–58. Springer, Berlin (2013b). doi:[10.1007/978-3-642-39056-2-4](https://doi.org/10.1007/978-3-642-39056-2-4)
- Franklin, J.: The elements of statistical learning: data mining, inference and prediction. *Math. Intell.* **27**(2), 83–85 (2005). doi:[10.1007/BF02985802](https://doi.org/10.1007/BF02985802)
- Fuchs, M., Zanker, M.: Multi-criteria ratings for recommender systems: an empirical analysis in the tourism domain. In: Huemer, C., Lops, P. (eds.) *E-Commerce and Web Technologies. Lecture Notes in Business Information Processing*, vol. 123, pp. 100–111. Springer, Berlin (2012). doi:[10.1007/978-3-642-32273-0-9](https://doi.org/10.1007/978-3-642-32273-0-9)
- Ganu, G., Kakodkar, Y., Marian, A.: Improving the quality of predictions using textual information in online user reviews. *Inf. Syst.* **38**(1), 1–15 (2013). doi:[10.1016/j.is.2012.03.001](https://doi.org/10.1016/j.is.2012.03.001)
- Gunawardana, A., Shani, G.: A survey of accuracy evaluation metrics of recommendation tasks. *J. Mach. Learn. Res.* 10:2935–2962 (2009). <http://dl.acm.org/citation.cfm?id=1577069.1755883>
- Hammer, S., Wißner, M., André.: Trust-based decision-making for smart and adaptive environments. *User Model. User-Adapt. Interact* **25**, 3 (2015)
- Hariri, N., Mobasher, B., Burke, R., Zheng, Y.: Context-aware recommendation based on review mining. In: *Proceedings of the Ninth International Workshop on Intelligent Techniques for Web Personalization and Recommender Systems (ITWP)*, International Joint Conferences on Artificial Intelligence (IJCAI), pp. 30–36. Barcelona (2011)
- Hariri, N., Mobasher, B., Burke, R.: Context-aware music recommendation based on latent topic sequential patterns. In: *RecSys '12, Proceedings of the Sixth ACM Conference on Recommender Systems*, pp. 131–138. ACM, Dublin (2012) doi:[10.1145/2365952.2365979](https://doi.org/10.1145/2365952.2365979)
- Hatala, M., Wakkary, R.: Ontology-based user modeling in an augmented audio reality system for museums. *User Model. User-Adapt. Interact.* **15**(3–4), 339–380 (2005). doi:[10.1007/s11257-005-2304-5](https://doi.org/10.1007/s11257-005-2304-5)
- Hu, M., Liu, B.: Mining and summarizing customer reviews. In: *KDD '04, Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 168–177. ACM, Seattle (2004). doi:[10.1145/1014052.1014073](https://doi.org/10.1145/1014052.1014073)
- Jakob, N., Weber, S.H., Müller, M.C., Gurevych, I.: Beyond the stars: exploiting free-text user reviews to improve the accuracy of movie recommendations. In: *TSA '09, Proceedings of the First International CIKM Workshop on Topic-sentiment Analysis for Mass Opinion*, pp. 57–64. ACM, Hong Kong (2009). doi:[10.1145/1651461.1651473](https://doi.org/10.1145/1651461.1651473)
- Jamali, M., Ester, M.: Trustwalker: a random walk model for combining trust-based and item-based recommendation. In: *KDD '09, Proceedings of the Fifteenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 397–406. ACM, Paris (2009). doi:[10.1145/1557019.1557067](https://doi.org/10.1145/1557019.1557067)
- Jameson, A., Krgler, A.: Preface to the special issue on user modeling in ubiquitous computing. *User Model. User-Adapt. Interact.* **15**(3–4), 193–195 (2005). doi:[10.1007/s11257-005-2335-y](https://doi.org/10.1007/s11257-005-2335-y)
- Jannach, D., Karakaya, Z., Gedikli, F.: Accuracy improvements for multi-criteria recommender systems. In: *EC '12, Proceedings of the Thirteenth ACM Conference on Electronic Commerce*, pp. 674–689. ACM, Valencia (2012). doi:[10.1145/2229012.2229065](https://doi.org/10.1145/2229012.2229065)
- Jason, W., Chong, W., Ron, W., Adam, B.: Latent collaborative retrieval. *ICML '12. Proceedings of the Twenty-ninth International Conference on Machine Learning*, pp. 9–16. Omnipress, Edinburgh (2012)

- Karatzoglou, A., Amatriain, X., Baltrunas, L., Oliver, N.: Multiverse recommendation: n-dimensional tensor factorization for context-aware collaborative filtering. In: *RecSys '10, Proceedings of the Fourth ACM Conference on Recommender Systems*, pp. 79–86. ACM, Barcelona (2010). doi:[10.1145/1864708.1864727](https://doi.org/10.1145/1864708.1864727)
- Leung, C.W., Chan, S.C., Chung, F.-L.: Integrating collaborative filtering and sentiment analysis: a rating inference approach. In: *Proceedings of the ECAI 2006 Workshop on Recommender Systems*, pp. 62–66. Citeseer, Riva del Garda (2006)
- Levi, A., Mokryn, O., Diot, C., Taft, N.: Finding a needle in a haystack of reviews: cold start context-based hotel recommender system. In: *RecSys '12, Proceedings of the Sixth ACM Conference on Recommender Systems*, pp. 115–122. ACM, Dublin (2012). doi:[10.1145/2365952.2365977](https://doi.org/10.1145/2365952.2365977)
- Li, Y., Nie, J., Zhang, Y., Wang, B., Yan, B., Weng, F.: Contextual recommendation based on text mining. In: *COLING '10, Proceedings of the Twenty-third International Conference on Computational Linguistics*, Association for Computational Linguistics, pp. 692–700. Beijing (2010). <http://dl.acm.org/citation.cfm?id=1944566.1944645>
- Liu, L., Mehandjiev, N., Xu, D.-L.: Multi-criteria service recommendation based on user criteria preferences. In: *RecSys '11, Proceedings of the Fifth ACM Conference on Recommender Systems*, pp. 77–84. ACM, Chicago (2011). doi:[10.1145/2043932.2043950](https://doi.org/10.1145/2043932.2043950)
- Massa, P., Avesani, P.: Trust-aware recommender systems. In: *RecSys '07, Proceedings of the First ACM Conference on Recommender Systems*, pp. 17–24. ACM, Minneapolis (2007). doi:[10.1145/1297231.1297235](https://doi.org/10.1145/1297231.1297235)
- McAuley, J., Leskovec, J.: Hidden factors and hidden topics: understanding rating dimensions with review text. In: *RecSys '13, Proceedings of the 7th ACM Conference on Recommender Systems*, pp. 165–172. ACM, Hong Kong (2013). doi:[10.1145/2507157.2507163](https://doi.org/10.1145/2507157.2507163)
- Moghaddam, S., Ester, M.: Aspect-based opinion mining from product reviews. In: *SIGIR '12, Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 1184–1184. ACM, Portland (2012). doi:[10.1145/2348283.2348533](https://doi.org/10.1145/2348283.2348533)
- Panniello, U., Tuzhilin, A., Gorgoglione, M., Palmisano, C., Pedone, A.: Experimental comparison of pre- vs. post-filtering approaches in context-aware recommender systems. In: *RecSys '09, Proceedings of the Third ACM Conference on Recommender Systems*, pp. 265–268. ACM (2009). doi:[10.1145/1639714.1639764](https://doi.org/10.1145/1639714.1639764)
- Park, H.-S., Yoo, J.-O., Cho, S.-B.: A context-aware music recommendation system using fuzzy bayesian networks with utility theory. In: Wang, L. (ed.) *Fuzzy Systems and Knowledge Discovery. Lecture Notes in Computer Science*, vol. 4223, pp. 970–979. Springer, Berlin (2006)
- Pero, S., Horvth, T.: Opinion-driven matrix factorization for rating prediction. In: Carberry, S., Weibelzahl, S., Micarelli, A., Semeraro, G. (eds.) *User Modeling, Adaptation, and Personalization. Lecture Notes in Computer Science*, vol. 7899, pp. 1–13. Springer, Berlin (2013). doi:[10.1007/978-3-642-38844-6-1](https://doi.org/10.1007/978-3-642-38844-6-1)
- Petrelli, D., Not, E.: User-centred design of flexible hypermedia for a mobile guide: reflections on the hyperaudio experience. *User Model. User-Adapt. Interact.* **15**(3–4), 303–338 (2005). doi:[10.1007/s11257-005-8816-1](https://doi.org/10.1007/s11257-005-8816-1)
- Poirier, D., Tellier, I., Fessant, F., Schluth, J.: Towards text-based recommendations. In: *RIAO '10, Adaptivity, Personalization and Fusion of Heterogeneous Information, LE CENTRE DE HAUTES ETUDES INTERNATIONALES D'INFORMATIQUE DOCUMENTAIRE*, pp. 136–137. Paris (2010). <http://dl.acm.org/citation.cfm?id=1937055.1937089>
- Shani, G., Gunawardana, A.: Evaluating recommendation systems. In: Ricci, F., Rokach, L., Shapira, B., Kantor, P.B. (eds.) *Recommender Systems Handbook*, pp. 257–297. Springer, New York (2011). doi:[10.1007/978-0-387-85820-3-8](https://doi.org/10.1007/978-0-387-85820-3-8)
- Smucker, M.D., Allan, J., Carterette, B.: A comparison of statistical significance tests for information retrieval evaluation. In: *CIKM '07, Proceedings of the Sixteenth ACM Conference on Conference on Information and Knowledge Management*, pp. 623–632. ACM, Lisbon (2007). doi:[10.1145/1321440.1321528](https://doi.org/10.1145/1321440.1321528)
- Wang, F., Chen, L.: Recommending inexperienced products via learning from consumer reviews. In: *2012 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology (WI-IAT)*, vol. 1, pp. 596–603. Macau (2012). doi:[10.1109/WI-IAT.2012.209](https://doi.org/10.1109/WI-IAT.2012.209)
- Wang, H., Lu, Y., Zhai, C.: Latent aspect rating analysis on review text data: a rating regression approach. In: *KDD '10, Proceedings of the Sixteenth ACM SIGKDD International Conference on Knowledge*

- Discovery and Data Mining, pp. 783–792. ACM, Washington, DC (2010). doi:[10.1145/1835804.1835903](https://doi.org/10.1145/1835804.1835903)
- Wang, Y., Liu, Y., Yu, X.: Collaborative filtering with aspect-based opinion mining: a tensor factorization approach. In: ICDM '12, Proceedings of the Twelfth International Conference on Data Mining, pp. 1152–1157. IEEE Computer Society, Washington, DC (2012). doi:[10.1109/ICDM.2012.76](https://doi.org/10.1109/ICDM.2012.76)
- Weston, J., Bengio, S., Usunier, N.: Wsabie: scaling up to large vocabulary image annotation. In: IJCAI'11, Proceedings of the Twenty-second International Joint Conference on Artificial Intelligence, pp. 2764–2770. AAAI Press, Barcelona (2011). doi:[10.5591/978-1-57735-516-8/IJCAI11-460](https://doi.org/10.5591/978-1-57735-516-8/IJCAI11-460)
- Wilson, T., Wiebe, J., Hoffmann, P.: Recognizing contextual polarity in phrase-level sentiment analysis. In: HLT '05, Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing, Association for Computational Linguistics, pp. 347–354. Vancouver (2005). doi:[10.3115/1220575.1220619](https://doi.org/10.3115/1220575.1220619)
- Yang, Y., Pedersen, J.O.: A comparative study on feature selection in text categorization. In: ICML '97, Proceedings of the Fourteenth International Conference on Machine Learning, pp. 412–420. Morgan Kaufmann Publishers Inc., San Francisco (1997). <http://dl.acm.org/citation.cfm?id=645526.657137>
- Yu, J., Zha, Z.-J., Wang, M., Chua, T.-S.: Aspect ranking: identifying important product aspects from online consumer reviews. In: HLT '11, Proceedings of the Forty-ninth Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, pp. 1496–1505. Portland (2011). <http://dl.acm.org/citation.cfm?id=2002472.2002654>
- Yu, Z., Zhou, X., Zhang, D., Chin, C.-Y., Wang, X., men, J.: Supporting context-aware media recommendations for smart phones. IEEE Pervasive Comput. 5(3), 68–75 (2006). doi:[10.1109/MPRV.2006.61](https://doi.org/10.1109/MPRV.2006.61)
- Zhang, K., Narayanan, R., Choudhary, A.: Voice of the customers: mining online customer reviews for product feature-based ranking. In: WOSN'10, Proceedings of the Third Wconference on Online Social Networks, pp. 11–11. USENIX Association, Boston (2010). <http://dl.acm.org/citation.cfm?id=1863190.1863201>
- Zhang, W., Ding, G., Chen, L., Li, C., Zhang, C.: Generating virtual ratings from chinese reviews to augment online recommendations. ACM Trans. Intell. Syst. Technol. 4(1), 9:1–9:17 (2013). doi:[10.1145/2414425.2414434](https://doi.org/10.1145/2414425.2414434)
- Zhang, Y., Zhuang, Y., Wu, J., Zhang, L.: Applying probabilistic latent semantic analysis to multi-criteria recommender system. AI Commun. 22(2), 97–107 (2009). <http://dl.acm.org/citation.cfm?id=1574514.1574517>
- Zheng, Y., Burke, R., Mobasher, B.: Recommendation with differential context weighting. In: Carberry, S., et al. (eds.) User Modeling, Adaptation, and Personalization, vol. 7899, pp. 152–164. Springer, Berlin (2013). doi:[10.1007/978-3-642-38844-6-13](https://doi.org/10.1007/978-3-642-38844-6-13)
- Zimmermann, A., Specht, M., Lorenz, A.: Personalization and context management. User Model. User-Adapt. Interact. 15(3–4), 275–302 (2005). doi:[10.1007/s11257-005-1092-2](https://doi.org/10.1007/s11257-005-1092-2)

Guanliang Chen received his B.E. and M.E. degrees in Software Engineering from South China University of Technology. He had been an exchange research student at Hong Kong Baptist University, under the supervision of Dr. Li Chen, from May 2013 to Jan. 2014. His primary research interests lie in the areas of personalization and recommender systems, social network, data mining, web intelligence, personalized E-learning, and human computer interaction.

Li Chen is Assistant Professor of Computer Science at Hong Kong Baptist University. Dr. Chen received her bachelor and master degrees in Computer Science from Peking University, China, and Ph.D. degree in Computer Science from Swiss Federal Institute of Technology in Lausanne (EPFL). Her primary interests lie in the areas of user modeling, web personalization, recommender systems, human-computer interaction, and data mining applications. She has co-authored over 70 technical papers and has co-edited several special issues in ACM transactions.