Integrating Context-Awareness and Multi-Criteria Decision Making in Educational Learning

Conference Paper · April 2019

DOI: 10.1145/3297280.3297522

Yong Zheng
Illinois Institute of Technology

Integrating Context-Awareness and Multi-Criteria Decision Making in Educational Learning

Yong Zheng
Illinois Inst. of Technology
Chicago, Illinois, USA
yong.zheng@iit.edu

Shephalika Shekhar
Illinois Inst. of Technology
Chicago, Illinois, USA
sshekhar1@hawk.iit.edu

Alisha Anna Jose
Illinois Inst. of Technology
Chicago, Illinois, USA
ajose1@hawk.iit.edu

Sunil Kumar Rai
Illinois Inst. of Technology
Chicago, Illinois, USA
srai@hawk.iit.edu

ABSTRACT

Recommender system is a well-known information system which can capture user tastes and produce item recommendations to the end users. Context-aware recommender systems (CARS) additionally take contexts (e.g., location, time, weather, etc) into consideration, and multi-criteria recommender systems (MCRS) utilize user preferences in multiple criteria to better generate recommendations. Both CARS and MCRS have been widely applied in the real-world applications, such as tourism, movies, music and dining. However, there are no existing research which exploits the methods to integrate them together, not to mention the contributions in the area of educational learning. In this paper, we make the first attempt to integrate context-awareness and multi-criteria decision making in the recommender systems by using the educational data as a case study. Our experimental results reveal that it is able to help produce more accurate recommendations by taking advantage of these two recommendation strategies. We also perform experiments on a tourism data set to demonstrate that the proposed methods can also be generalized to other domains.

CCS CONCEPTS

• **Information systems → Recommender systems**;

KEYWORDS

recommender systems; context; multi-criteria; education; learning

1 INTRODUCTION AND MOTIVATIONS

Recommender systems (RS) is an effective solution to alleviate the problem of information overload and assist decision making. A traditional recommender may produce a list of recommendations tailored by user preferences. It has been widely applied to several domains and applications, such as online streaming (e.g., Netflix, Spotify), e-commerce (e.g., Amazon.com), social networks (e.g., Facebook), tourism (e.g., TripAdvisor), etc. Technology-enhanced learning (TEL) becomes popular recently, since it aims to design, develop and test sociotechnical innovations that will support and enhance learning practices of both individuals and the organizations [21]. RS have been applied in the education area [11, 20, 21, 25] to support TEL, especially in personalized and adaptive learning.

Several novel RS were proposed to improve the recommendations and adapt to new applications recently. Context-aware recommender systems (CARS) [4] is one of these examples. CARS leverage the value of recommendations by exploiting context information (e.g., time, location, weather, etc) that affects user preferences, since a user's taste may vary from contexts to contexts. For example, a user may choose a different type of the movie if he or she is going to watch the movie with *kids* rather than with *partner*. Or, a user may perform a different outdoor activity when it is *at weekend* rather than *the weekday*. In these two examples, "companion" and the "day of the week" are two context variables which may affect a user's decisions. Another new type of the RS is multi-criteria recommender systems (MCRS) [3] which take advantage of user preferences in multiple criteria. For example, users on the TripAdvisor.com can give ratings on multiple criteria (such as room size, cleanness, customer service, etc) in addition to the overall rating on a hotel. MCRS try to aggregate these multi-criteria preferences in order to better predict the user tastes on the items.

Both CARS and MCRS have been applied to different applications (e.g., movies [23, 27] and tourism [7, 17, 27]), but there are no existing research which integrates them together, not to mention the contributions in educational learning. In addition, CARS have been exploited for the learning area [25], but the application of MCRS in the education is still under investigation. Our contributions in this paper can be summarized as follows:

- We introduce an educational data which contain both context and multi-criteria information.
- We exploit MCRS based on the education data, and make the first attempt to integrate context-awareness and multi-criteria decision making in recommender systems.
- We compare the proposed methods with multiple baselines, and demonstrate the effectiveness of the integration.
- We additionally apply the proposed methods in a tourism data in order to demonstrates that these methods can be generalized to other domains.

2 RELATED WORK

2.1 Context-Aware Recommender Systems

We introduce the terminologies in CARS as follows. Take Table 1 for example, there is one user U_1, one movie T_1, and three context dimensions – Time (weekend or weekday), Location (at home or cinema) and Companion (alone, partner, family). In the following

SAC '19, April 8–12, 2019, Limassol, Cyprus

ACM ISBN 978-1-4503-5933-7/19/04…$15.00
https://doi.org/10.1145/3297280.3297522

discussion, we use *context dimension* to denote the contextual variable, e.g. "Location". The term *context condition* refers to a specific value in a dimension, e.g. "home" and "cinema" are two contextual conditions in "Location". The *contexts* or *context situation* is, therefore, a set of contextual conditions, e.g. {*weekend, home, family*}.

Table 1: Contextual Ratings on Movies

User	Item	Rating	Time	Location	Companion
U_1	T_1	3	weekend	home	alone
U_1	T_1	5	weekend	cinema	partner
U_1	T_1	?	weekday	home	family

The most commonly used definition for context is the one given in 1999 by G.D. Abowd, et al. "*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 applications themselves*" [1]. We provide a finer-grained definition of contexts in CARS, where context information refers to the dynamic variables which may change frequently when a user is going to perform a same activity (e.g., listening to music, watching a movie, dinning at a restaurant, etc) [26]. More specifically, they could be the dynamic attributes of a user (e.g., emotions) and/or the attributes of the activity, such as when and where a user will watch movies.

There are usually three ways to build a contextual recommender. *Contextual pre-filtering*, such as the splitting-based methods [33] will use the context information to filter out irrelevant rating profiles and apply the traditional recommendation algorithms to produce the recommendations. By contrast, *contextual post-filtering* methods [28] produce recommendations without considering contexts, and then utilize context information to adjust the predicted ratings or re-rank the items. *Contextual modeling* [5, 34] is the most complicated strategy, while contexts are directly incorporated into the predictive models. There are few research on post-filtering, while pre-filtering and contextual modeling become the most popular and effective ways to build CARS.

2.2 Multi-Criteria Recommender Systems

An example of data in MCRS can be shown by Table 2. The *rating* refers to the users' overall rating on the items. We also have users' ratings on multiple criteria, such as room, check-in and service.

Table 2: Example of Rating Data from TripAdvisor

User	Item	Rating	Room	Check-in	Service
U_1	T_1	3	3	4	3
U_2	T_2	4	4	4	5
U_3	T_1	?	?	?	?

MCRS tries to take multi-criteria ratings into account to build better recommenders, i.e., given a user U_3 and an item T_1, MCRS tries to predict U_3's overall rating on T_1 by estimating and then aggregating his or her ratings on the three criteria as shown in Table 2. One of the popular methods is the heuristic approach [2, 19] which utilize the multi-criteria ratings to better calculate user-user or item-item similarities in the collaborative filtering algorithms. Another one is the model-based approach [2, 27] which constructs a predictive model to estimate a user' overall rating on one item from the observed multi-criteria ratings.

2.3 Educational Recommender Systems

Educational recommender systems emerged as a technology-enhanced learning [21] method. It has been successfully applied to the education domain to suggest books for K-12 users [24], recommend after-school programs [8] or materials in informal learning [10], or suggest appropriate citations [14] in paper writings.

Researchers have considered context-awareness in educational learning. For example, Cui, et al. [9] introduce a language environment that adapts the recommendations or interactions to the individual learner's understanding, as well as the contextual situations, including the location and level of the noise in the location. Lonsdale, et al. [18] take advantage of location information to recommend learning partners nearby for a leaner. He, et al. [14] suggest publications to cite according to the topics associated with the sentences of paragraph in the process of paper writings.

By contrast, there are limited research in exploiting multi-criteria recommender systems in the learning area. Manouselis, et al. [22] utilizes three real-world data sets in the learning area – OEreal (477 ratings by 99 users on 345 items), EUN (2,554 ratings by 228 users on 899 learning resources), MERLOT (2,626 ratings by 18 users on 2,603 resources). Take the EUN data for example, it was collected from a teacher portal of the European Schoolnet, and there are six criteria to further indicate how a student is interested in one learning resource, such as the ease of integration in the classroom, the relevance to the teaching topics, ability to assist learning, etc. They investigate how multi-criteria recommendations can help suggest appropriate learning resources, and discover the significant effectiveness on the EUN data.

As far as we know, there are no attempts to integrate CARS and MCRS together, not to mention the contributions in the area of educational learning. In this paper, we exploit MCRS in educational learning, and investigate how to integrate context-awareness with multi-criteria decision makings in RS.

3 DATA AND PROBLEM STATEMENT

We use an educational data [31] which was collected from our Web-based learning portal that improves the process of teaching and learning for faculties and students. One of the components in our portal is project recommendations. Students are required to complete a project for some courses in which they need to find a data set and topics, define research problems, and utilize their skills to solve the proposed research problems. We pre-define the potential topics by giving a list of data sets on Kaggle.com in our learning portal. In addition, students may choose to work on the project by themselves or a team work. Therefore, this data set is also useful for investigation on group learning or recommendations [29, 30].

Each student was asked to fill the questionnaire by himself or herself. Each subject should select at least three liked and disliked topics of the projects, and provide an overall rating to them. In addition, they were asked to rate each selected project on three criteria: how interesting the application area is (i.e., App), how convenient the data processing will be (i.e., Data), how easy the whole project is (i.e., Ease). The rating scale for all ratings is 1 to 5.

We have collected the data for two years. Table 3 presents an example of our data. There are a total of 3,306 ratings given by 269 students on 70 items (i.e., the topics of the projects). In addition to the overall ratings, we have students' ratings on the three criteria (i.e., App, Data and Ease). There are three context dimensions: the type of the class (database, data analytics and data mining) semester (Spring or Fall) and year of the course (2017 or 2018).

Table 3: Example of The Educational Data

User	Item	Rating	App	Data	Ease	Class	Year	Semester
10	41	4	4	4	4	DM	2017	Fall
10	60	2	2	2	2	DA	2017	Fall
12	21	4	4	5	4	DA	2018	Spring
14	35	2	1	2	1	DM	2017	Spring
10	30	?	?	?	?	DA	2018	Fall

Note that it is difficult to collect educational data sets with both context and multi-criteria information. As mentioned by Drachsler, et al. [11], "*collecting data (in the learning area) was limited in terms of cost, time requirements, scope and authenticity of the data, as this was typically done using single groups or classes for an experiment*". Our data is even larger than the three educational data (i.e., OEreal, EUN, MERLOT) with multi-criteria ratings which were introduced in the previous section. Moreover, our data is small but dense in ratings, while each student rated at least six items.

The research problems in our paper can be summarized as follows:

- We compare traditional recommendation methods with the context-aware and multi-criteria recommendation algorithms, in order to examine the effectiveness of context-awareness and multi-criteria decision makings respectively.
- We try to figure out different ways to integrate context-awareness and multi-criteria decision making, and observe whether they can outperform the context-aware and multi-criteria recommendation algorithms respectively.
- Finally, we apply the proposed methods to a tourism data set which will be introduced later, in order to demonstrate that our methods can be generalized to other domains.

4 METHODOLOGIES

In this section, we first introduce the methods in CARS and MCRS respectively, and finally discuss how to integrate them.

4.1 Context-Aware Approaches

The rating prediction in CARS can be described as *R*: *Users* × *Items* × *Contexts* → *Ratings* [4]. In our case, we try to estimate how a user will rate a topic of the project by given the context information (i.e., the type of the class, semester and year). In our work, we select context-aware matrix factorization (CAMF) [5] as the representative for the contextual modeling method to produce context-aware recommendations, since it is considered as a popular and standard benchmark algorithm in CARS. CAMF learns rating deviations or biases in the process of matrix factorization. There are different versions of CAMF, while we use the basic one which learns the rating deviation in each context condition independently. The rating prediction in the CAMF can be shown as Equation 1.

$$\hat{r}_{uic_{k,1}c_{k,2}\ldots c_{k,L}} = \mu + b_u + b_i + \sum_{j=1}^{L} B_{c_{k,j}} + \overrightarrow{p_u} \cdot \overrightarrow{q_i} \tag{1}$$

$\hat{r}_{uic_{k,1}c_{k,2}\ldots c_{k,L}}$ denotes the predicted rating given by user u on the item i in a specific context situation c_k. Assume there are L contextual dimensions in total, $c_k = \{c_{k,1}c_{k,2}\ldots c_{k,L}\}$ is used to describe the contextual situation, where $c_{k,j}$ denotes the contextual condition in the j^{th} context dimension. b_u and b_i represent the user bias and item bias. $B_{c_{k,j}}$ is the contextual rating deviation or bias associated with the contextual condition $c_{k,j}$. A finer-grained way is to assume this context bias is different for each user or item, but we ignore this option for simplicity. μ is the global average rating in the data, while $\overrightarrow{p_u}$ and $\overrightarrow{q_i}$ represent the latent-factor vectors for user u and item i respectively. We use the stochastic gradient descent as the optimizer to learn these parameters.

4.2 Multi-Criteria Decision Making

In MCRS, we have user preferences in multiple criteria, in addition to users' overall ratings on the item. In our case, we have students' ratings in App, Data and Ease. The rating prediction task in multi-criteria decision making can be described by the Equation below, where R_0 denotes the overall rating, and $R_1, R_2, ..., R_k$ represent the rating on k criteria. The function f represents the rating prediction or aggregation function which tries to utilize the multi-criteria ratings to predict the overall rating.

$$R_0 = f(R_1, R_2, ..., R_k) \tag{2}$$

We decide to build models on top of the aggregation-based multi-criteria recommendation algorithm, since it is straightforward and popular. There are usually two steps involved in this aggregation-based strategy: *multi-criteria rating predictions* which is the process of predicting the rating on each criterion, and *rating aggregations* which refers to the process of aggregating the predicted multi-criteria ratings to estimate the overall rating. The recommended items will be produced based on these estimated overall ratings. We specifically introduce these two steps respectively as follows.

4.2.1 Multi-Criteria Rating Predictions. We can predict the multi-criteria ratings independently or dependently. In the *independent* method, we predict the individual rating on each criterion by using the rating matrix associated with each criterion. Take Table 3 for example, to predict how a user will rate an item in the criterion "App", we use the rating matrix $< User, Item, App >$ only. This method is simple but it ignores the correlations among criteria. In our work, we adopt the standard biased matrix factorization (BiasedMF) [16], as shown by Equation 3.

$$\hat{r}_{ui} = \mu + b_u + b_i + \overrightarrow{p_u} \cdot \overrightarrow{q_i} \tag{3}$$

Another way is the *dependent* method, such as criteria chains [27] which considers the correlations among multiple criteria. First of all, it uses the information gain based on the entropy values to define the sequence of the criteria as the chain. In our case, the sequence is "App - Data - Ease", since the variable "App" is the dimension with largest impurity measured by information gain, and "Ease" is the one with least impurity. Once the sequence is defined, we will predict the ratings in the dimension "App" by using

BiasedMF first. The predicted ratings in "App" will be reviewed as context to be used to predict the rating in "Data" by using CAMF. The predicted ratings in "App" and "Data" will be viewed as inputs in CAMF to finally predict the rating in "Ease". In this way, the correlation among multiple criteria will be considered, which may improve the predictive performance.

4.2.2 Rating Aggregations. Once the multi-criteria ratings have been predicted, we can aggregate them together to predict the overall rating. There are two possible methods too. One is *linear aggregation* [2] which can be described by Equation 4.

$$R_0 = w_1 * R_1 + w_2 * R_2 + ... + w_k * R_k + t \quad (4)$$

By this way, we can aggregate the predicted multi-criteria ratings by using a linear regression to learn the weights $w_1, w_2, ..., w_k$ in the model. To achieve better performance, we use support vector regression (SVR) [12] as the aggregation function.

Another way is the *conditional aggregation* which was proposed in criteria chains [27]. In this method, we view the multi-criteria preferences as contexts and utilize context-aware recommendation methods (e.g., CAMF) to predict the overall ratings. Namely, we try to estimate a user's rating on an item by considering the situation or conditions that he or she already gave a specific preference to the three criteria (i.e., App, Data, Ease). This is actually a process of context-aware predictions, but we name it as "conditional aggregation" to avoid using the term "contextual aggregation". Note that we did not use the context variables (such as class, year, semester) in this step. It is worth mentioning that the predicted ratings on each criterion could be float values with decimals. According to the instructions in criteria chains [27], we need to round these values to integers to be used in the contextual aggregations, in order to alleviate the sparsity issue and improve the quality of predictions.

4.3 Integration Methods

Based on the introductions above, we can see that there are two steps in the MCRS: multi-criteria rating predictions and the rating aggregations. For each step, we have two methods – independent and dependent methods for the multi-criteria rating predictions, and the linear and conditional aggregations for the second step. It results in four baseline approaches for multi-criteria recommendations. To integrate context information (i.e., App, Data and Ease), we can incorporate them into these two steps respectively and accordingly.

4.3.1 Context-Aware Multi-Criteria Rating Predictions. In this way, we only fuse the contexts into the process of multi-criteria predictions. Recall that the multi-criteria ratings can be predicted independently or dependently. For each rating prediction process, we can use the context-aware prediction methods. Take the independent prediction for example, to predict how a user will rate an item in the criterion "App", we use the rating matrix $< User, Item, App >$ only. By incorporating contexts into this process, we apply CAMF to the matrix $< User, Item, App, Class, Year, Semester >$. Alternatively, we can also additionally consider "Class, Year, Semester" as contexts in the criteria chains (i.e., the dependent method in multi-criteria rating predictions) to predict the multi-criteria ratings.

4.3.2 Context-Aware Rating Aggregations. Context-awareness can also be used in the step of rating aggregations. In the linear aggregation, we use SVR to aggregate the predicted multi-criteria ratings. But we can also consider the context information as categorical variables which can be the factors in the regression process. In the conditional aggregations, we already consider the predicted multi-criteria ratings as contexts, but we can additionally add the context variables (e.g, Class, Year and Semester) in the process too. As a summary, we have two methods for each step, and we can incorporate context information into each method.

5 EXPERIMENTS AND RESULTS

5.1 Baseline and Proposed Approaches

We design four types of the approaches as shown in Table 5:

- The model *BiasedMF* (i.e., M0) is the biased matrix factorization model [16] by using the rating matrix $< User, Item, Rating >$ only without considering contexts and multi-criteria ratings.
- The model *CAMF* (i.e., M13) is the context-aware recommendation approach described in Section 4.1. We use the rating matrix $< User, Item, Rating, Class, Year, Semester >$ without considering multi-criteria preferences.
- The multi-criteria recommendation models are denoted as M1 to M4. There are two methods for each step, which results in four models. These models utilize multi-criteria ratings only without considering contexts.
- The context-integrated models are represented by M5 to M12. We can incorporate context into each method in each step, which results in eight models in total for the examinations.

Table 4: Integration Methods

	Step 1. Multi-Criteria Rating Predictions	Step 2. Rating Aggregations
M1	M5	M9
M2	M6	M10
M3	M7	M11
M4	M8	M12

Furthermore, we correlate M5-M12 with M1-M4 as shown by Table 4. It shows the relationship between the integrated methods and the original multi-criteria recommendation approaches. More specifically, M1-M4 are the basic multi-criteria recommendation approaches, while M5-M12 are the models which incorporate context information to the two steps of the algorithms respectively. Take M1 for example, the table puts M5 and M9 in the two steps respectively. It infers that M5 incorporates contexts into the step of multi-criteria rating predictions, in comparison with M1. Accordingly, M9 fuses contexts to the step of rating aggregations, while it utilizes the same way to predict multi-criteria ratings as M1 does. Note that M4 is equivalent to the approach of criteria chains [27] which was demonstrated as an effective multi-criteria recommendation approaches that take advantage of correlations among multiple criteria.

5.2 Evaluation Protocols

Note that we use time of the year as context information in the data. Theoretically, we should use a time-dependent based evaluation. Namely, we should use the past data to predict the future, e.g.,

Table 5: Descriptions of Recommendation Models

Models	Descriptions
M0	We use BiasedMF by using the rating matrix <User, Item, Ratings>only without considering contexts and multi-criteria ratings.
M1	We use independent multi-criteria rating predictions and the linear aggregations.
M2	We use independent multi-criteria rating predictions and the conditional aggregations.
M3	We use dependent multi-criteria rating predictions and the linear aggregations.
M4	We use dependent multi-criteria rating predictions and the conditional aggregations.
M5	We incorporate contexts into independent multi-criteria rating predictions, and use linear aggregations afterwards.
M6	We incorporate contexts into independent multi-criteria rating predictions, and use conditional aggregations afterwards.
M7	We incorporate contexts into dependent multi-criteria rating predictions, and use linear aggregations afterwards.
M8	We incorporate contexts into dependent multi-criteria rating predictions, and use conditional aggregations afterwards.
M9	We use independent multi-criteria rating predictions and incorporate contexts into the linear aggregations.
M10	We use independent multi-criteria rating predictions and incorporate contexts into the conditional aggregations.
M11	We use dependent multi-criteria rating predictions and incorporate contexts into the linear aggregations.
M12	We use dependent multi-criteria rating predictions and incorporate contexts into the conditional aggregations.
M13	We use CAMF to produce recommendations by taking advantage of the context information only.

using the ratings in the year 2017 to predict students' tastes in 2018. However, the data is relatively small, while the records associated with the year 2017 are even fewer. In this case, we adopt the time-independent evaluations, blend the data and apply a 5-fold cross validation to produce the evaluation results.

We evaluate the performance of recommendations based on the top-N recommendation task. More specifically, we examine the performance of the top-N recommended items by given a user. We use the popular F-measure and normalized discounted cumulative gain (NDCG) [15] as the evaluation metrics. F-measure is a metric which takes precision and recall into consideration. Precision refers to the fraction of positive predictions over the top-N recommendations, while recall is the fraction of positive predictions with respect to the list of relevant items by a specific user. However, the high precision results may hurt recall sometimes. F-measure combines theses two metrics together. And we consider precision and recall have equal importance, so F-measure can be represented by Equation 5.

$$F - Measure = 2 \cdot \frac{precision \cdot recall}{precision + recall} \tag{5}$$

NDCG[1] is a ranking measure, where positions are discounted logarithmically. It is used to examine the quality of the ranks in the top-N recommendations. We evaluate the models based on these metrics for top-5 and top-10 recommendations. We only present the results based on the top-10 recommendations, since the results in top-5 recommendations show the similar patterns. We use CARSKit [35] and LibRec [13] recommendation libraries to produce intermediate results in the recommendation process.

5.3 Results and Findings

Figure 1 presents our experimental results. More specifically, Figure 1 a) shows the comparisons among our baseline approaches, while M0 does not consider context or multi-criteria ratings, M13 is CAMF which utilizes the context information only, and M1-M4 denote four multi-criteria recommenders without considering contexts. We can clearly observe that the models that consider context information only (i.e., M13) or multi-criteria ratings only (e.g., M1 and M2) can outperform M0 in both NDCG and F-Measure results. More specifically, M1 and M13 are the best performing models, while there are no significant differences between them. Not all of the multi-criteria recommendation models work well. For example, M3 failed to beat M0, and M4 becomes the worse model among the baseline approaches. Note that we use the dependent method to predict the multi-criteria ratings in M3 and M4, while we use an independent way in M1 and M2. The results reveal that there are no strong correlations among the multiple criteria in this educational data. We further look at the data in order to understand why this happens. There are three criteria in our case: App, Data, Ease. The criterion "App" can indicate a user's taste on the domain of the projects, while "Data" and "Ease" actually represent the difficulty of the projects from the perspective of the students. However, some students prefer to choose an easy project, while some others may prefer to select more challenging project. It results in some conflicting interests, and finally decrease the performance when we try to use the dependent way to predict multi-criteria ratings.

Figure 1 b) presents the F-Measure results by comparing the basic multi-criteria models (i.e., M1-M4) and the integrated methods that incorporate context information to the first step (i.e., the process of multi-criteria rating predictions). The mapping correlations can also be found in Table 4. For example, the model M5 uses the same way in rating aggregations as the one used in M1, but M5 additionally takes contexts (i.e., the type of the class, year, semester) into consideration in multi-criteria rating predictions. Based on the results shown in Figure 1 b), we can observe that the F-Measure results can be significantly improved by fusing contexts into the process of the multi-criteria rating predictions, except M7 in which we incorporate contexts into the dependent multi-criteria rating predictions, and use linear aggregations afterwards. More specifically, M6 improved M2 by 9.6%, while M5 improved M1 by 3.7%, and M8 improved M4 by 5.7% in F-Measure. Similar patterns can be found in NDCG.

The observations in Figure 1 b) reveal that context information are valuable to predict the multi-criteria ratings. To confirm this discovery, we further perform statistical analysis on the ratings associated with the three context dimensions. More specifically, we use a binary splitting method [6] to split the ratings according to the different context conditions. We have three context dimensions

[1] https://gist.github.com/bwhite/3726239

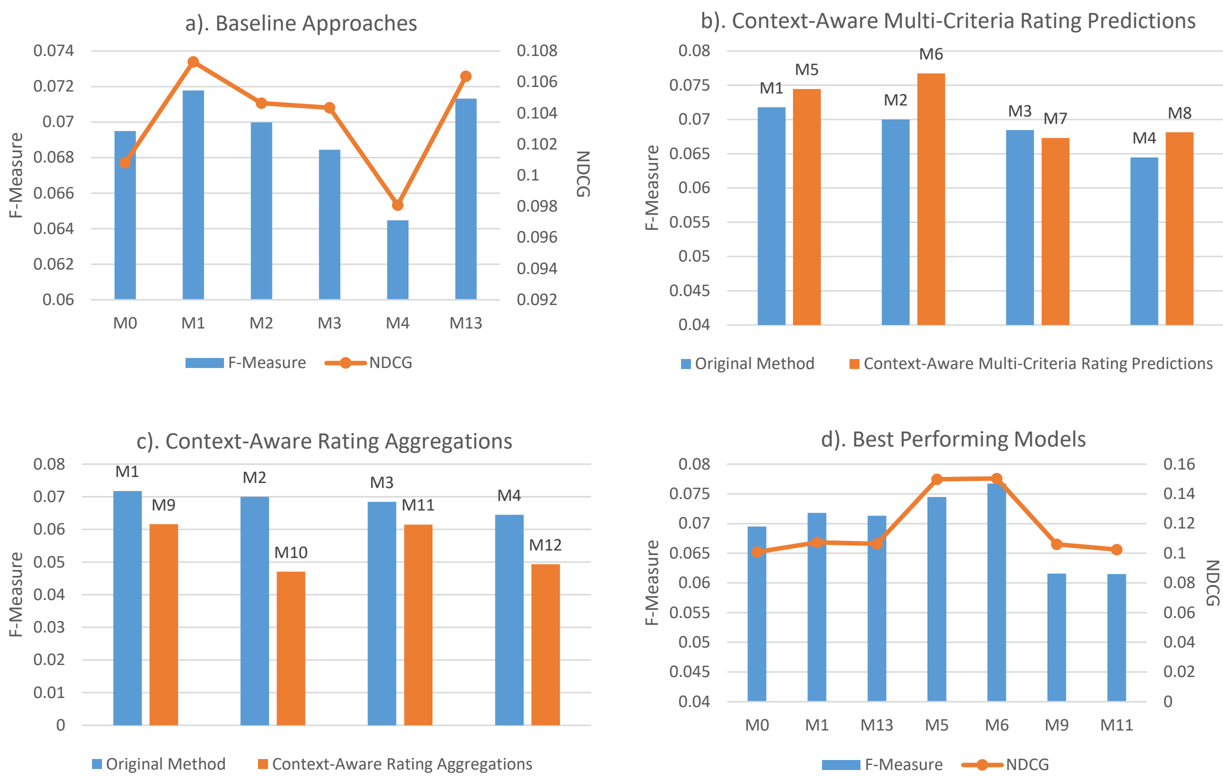

Figure 1: Experimental Results Based on The Educational Data

in our data – class, year and semester. The "year" and "semester" are two binary variables since there are only two values in these variables. Take semester for example, the two values are "Spring" and "Fall". Accordingly, it splits the ratings into two groups – ratings associated the Spring semester, and the ratings associated with the Fall semester. After that, we employ a two-independent sample hypothesis testing to examine whether the ratings in these two context conditions are significantly different at the 97% confidence level. We view the context information is useful to predict the ratings, if the significance exists based on the splitting by using a context condition. Note that, there are three conditions in the context dimension "class", we try all possible binary splits, where the results can be shown by Table 6. The cells with color green denote there is significant difference by each binary splitting for the ratings on that criterion. We can observe that all the three context variables are influential on the criterion "Ease", while only the class may be helpful to predict the criterion "App" and "Data". The contextual effects exist in these multi-criteria ratings, which helps explain why incorporating contexts into the process of multi-criteria rating predictions can improve the recommendation performance

Table 6: Results of the Significance Tests

Context Dimensions	Context Conditions	App	Data	Ease
Class	DM vs Non-DM			
	DA vs Non-DA			
	DB vs Non-DB			
Year	2017 vs 2018			
Semester	Spring vs Fall			

We can discover some interesting patterns based on the results in Table 6. For example, students may give significant and different ratings in "App" and "Data", if the course is a database class or others (i.e., data analytics and data mining classes). It makes sense since there are limited data preprocessing in the database class, while the students were required to build a relational database with well-designed data tables. One interesting finding is that students may leave different ratings in the criterion "Ease" in which all the three context variables (i.e., class, year and semester) pay an important role. The "year" and "semester" are reasonable since a new faculty with less teaching experiences taught these classes, and he or she may had different requirements from semesters to semesters.

Moreover, Figure 1 c) shows the F-Measure results by comparing the basic multi-criteria models and the integration methods that incorporate contexts into the second step only (i.e., the process of rating aggregations). The results are worse, which indicates that it is not helpful to contextualize the second step in the multi-criteria recommendations in this educational data.

Finally, Figure 1 d) compares the following best performing models – the baseline that does not consider contexts or multi-criteria ratings (i.e., M0), the best performing multi-criteria model (i.e., M1), the model that considers context only (i.e. M13), the best performing integrated methods which incorporates contexts into the process of multi-criteria rating predictions (i.e., M5 and M6), the best integrated models which fuses contexts into the process of rating aggregations (i.e., M9 and M11). We can observe that M5 and M6 are the two best performing models in these comparisons, in

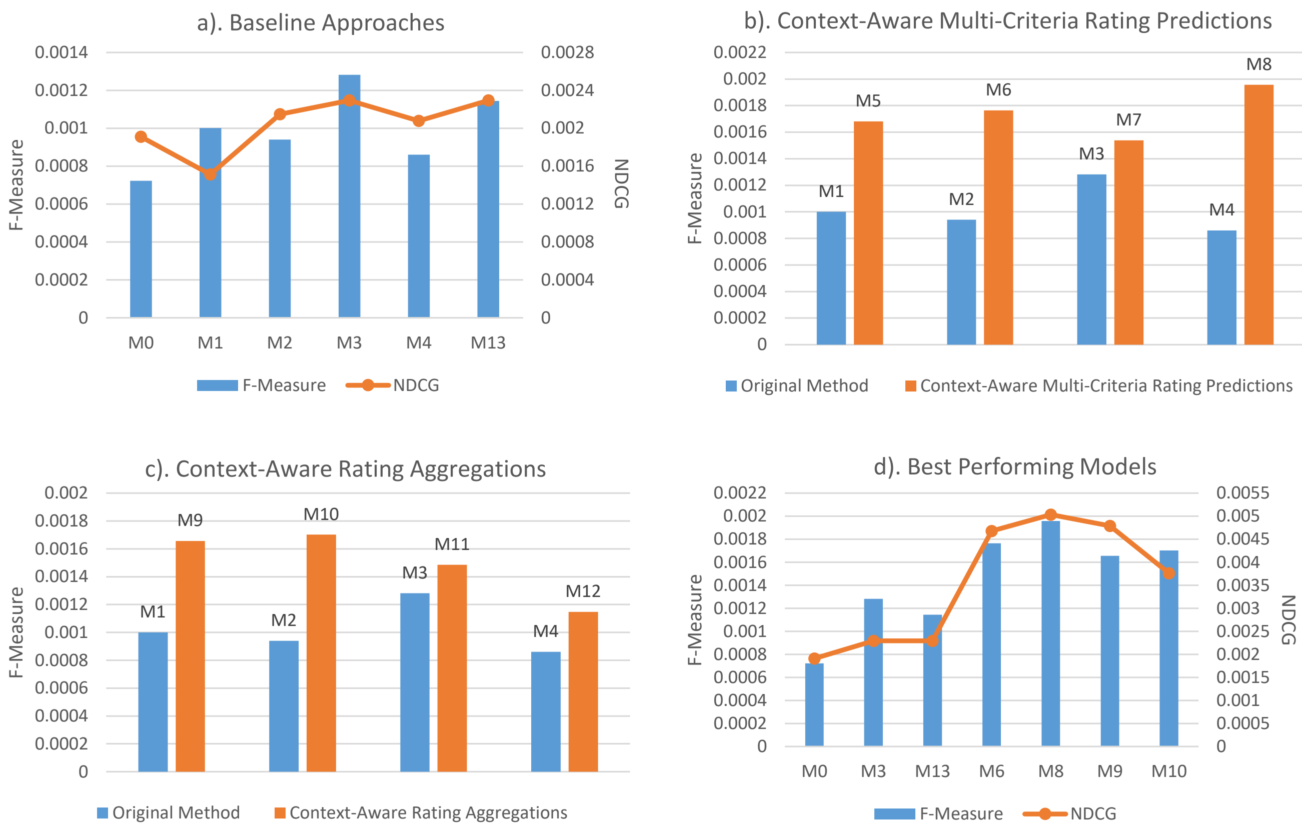

Figure 2: Experimental Results Based on The TripAdvisor Data

terms of both the F-Measure and NDCG results. Particularly, M6 improves the M0, M1 and M13 by 10.4%, 6.9% and 7.6% respectively in F-Measure, and 49.2%, 40.1% and 41.4% respectively in NDCG. We can conclude that it is able to improve the recommendation performance by integrating context-awareness and multi-criteria preferences in our educational data, while the best performing model (i.e., M6)incorporates context information (i.e., class, year and semester) into independent multi-criteria rating predictions, and use conditional aggregations afterwards.

5.4 Extensions

The proposed methods in this paper can be generalized to other domains, rather than the educational data only, though it is difficult to find a data with both contexts and multi-criteria preferences. We found a TripAdvisor data set[2] which was crawled from Tripadvisor.com. There are 3,524 ratings given by 551 users on 3,716 hotels. In addition to the overall ratings, we have ratings on six criteria – value, location, rooms, cleanliness, service and sleep quality. We have a single context variable "trip type" (e.g., business, couple, family, friends, solo). Based on the timestamp, we create additional context dimensions – year (e.g., 2000 to 2015), month (e.g., January to December), and quarter (Q1 to Q4). The sequence of the criteria in the chain is "value - service - rooms - sleep quality - cleanliness - location" according to the information gain. We apply the 5-fold cross validation and the same evaluation metrics on this data.

The experimental results based on the TripAdvisor data can be shown in Figure 2. In Figure 2 a), we can observe that both the context-aware model (i.e., M13) and all of the multi-criteria recommendation methods (i.e., M1-M4) can outperform the simple model which does not consider context or multi-criteria preferences (i.e., M0). M3 becomes the best performing multi-criteria recommendation models, in which we use dependent multi-criteria rating predictions and the linear aggregations.

Results in Figure 2 b) and c) can reveal that we are able to improve the recommendation performance by integrating context-awareness and multi-criteria recommendation approaches. M8 and M6 are the two top performing models if we incorporate context information into the step of predicting multi-criteria ratings. M9 and M10 are the top two models if we fuse contexts into the process of rating aggregations. Figure 2 c) shows different results from the ones in Figure 1 c), it is because users may give higher ratings in the multiple criteria if they like the hotels, and there are strong correlations among these criteria. But students may given conflicted ratings in "Data" and "Ease" if they like the domain of the projects and prefer to select more challenging projects.

Finally, Figure 2 d) presents the comparisons among the best performing baseline approaches (e.g., M0, M3, M13) and the integration methods (M6, M8, M9, M10) proposed in our paper. We

[2] https://www.researchgate.net/publication/308968574_TripAdvisor_Dataset

can observe that all of these integration methods can outperform the baseline approaches, while M8 becomes the best model, in which we incorporate contexts into dependent multi-criteria rating predictions, and use conditional aggregations afterwards. These results demonstrate that we are able to the effectiveness of integrating context-awareness and multi-criteria decision making in the recommender systems can be generalized to other domains.

6 CONCLUSIONS AND FUTURE WORK

In this paper, we make the first attempt to exploit different methods which integrate context-awareness and multi-criteria decision making in the recommender systems. We first evaluate these models based on the educational data, and then demonstrate that the proposed models can be generalized to other domains, such as a tourism data. We conclude that we are able to improve the recommendation performance by integrating CARS and MCRS together, but the best way to integrate them based on the characteristics of the domain and the data. For example, students may present conflicting interests in the three criteria we have in the data, which results in worse performance if we use the dependent method to predict the multi-criteria ratings.

There are plenty of the work we can explore in the future. On one hand, we plan to crawl more data from TripAdvisor in order to obtain a larger data set for the evaluation purpose. On the other hand, we are motivated to solve the issue of "conflicting interests" in our educational data, and we will try to figure out solutions to better take advantage of these multi-criteria ratings in the area of personalized learning. Furthermore, we explore how to incorporate contexts into the aggregation-based multi-criteria recommendation algorithms in this paper, while there are other ways to build multi-criteria recommendation models, such as the utility-based multi-criteria recommendations [32]. We will extend our work and investigate the methods of incorporating context-awareness into other types of multi-criteria recommender systems.

REFERENCES

[1] G. Abowd, A. Dey, P. Brown, N. Davies, M. Smith, and P. Steggles. 1999. Towards a better understanding of context and context-awareness. In *Handheld and Ubiquitous Computing*. Springer, 304–307.

[2] Gediminas Adomavicius and YoungOk Kwon. 2007. New recommendation techniques for multicriteria rating systems. *IEEE Intelligent Systems* 22, 3 (2007), 48–55.

[3] Gediminas Adomavicius and YoungOk Kwon. 2015. Multi-criteria recommender systems. In *Recommender Systems Handbook*. Springer, 847–880.

[4] Gediminas Adomavicius, Bamshad Mobasher, Francesco Ricci, and Alexander Tuzhilin. 2011. Context-Aware Recommender Systems. *AI Magazine* 32, 3 (2011), 67–80.

[5] Linas Baltrunas, Bernd Ludwig, and Francesco Ricci. 2011. Matrix factorization techniques for context aware recommendation. In *Proceedings of the fifth ACM conference on Recommender systems*. ACM, 301–304.

[6] Linas Baltrunas and Francesco Ricci. 2009. Context-based splitting of item ratings in collaborative filtering. In *Proceedings of ACM conference on Recommender systems*. 245–248.

[7] Matthias Braunhofer, Mehdi Elahi, and Francesco Ricci. 2014. STS: A Context-Aware Mobile Recommender System for Places of Interest.. In *UMAP Workshops*.

[8] Robin Burke, Yong Zheng, and Scott Riley. 2011. Experience Discovery: hybrid recommendation of student activities using social network data. In *Proceedings of the 2nd International Workshop on Information Heterogeneity and Fusion in Recommender Systems*. ACM, 49–52.

[9] Yanchun Cui and Susan Bull. 2005. Context and learned modeling for mobile foreign language learner. *Systems* 33, 2 (2005), 353–367.

[10] Hendrik Drachsler, Hans GK Hummel, and Rob Koper. 2009. Identifying the goal, user model and conditions of recommender systems for formal and informal learning. *Journal of Digital Information* 10, 2 (2009).

[11] Hendrik Drachsler, Katrien Verbert, Olga C Santos, and Nikos Manouselis. 2015. Panorama of recommender systems to support learning. In *Recommender systems handbook*. Springer, 421–451.

[12] Harris Drucker, Christopher JC Burges, Linda Kaufman, Alex J Smola, and Vladimir Vapnik. 1997. Support vector regression machines. In *Advances in neural information processing systems*. 155–161.

[13] Guibing Guo, Jie Zhang, Zhu Sun, and Neil Yorke-Smith. 2015. LibRec: A Java Library for Recommender Systems.. In *UMAP Workshops*, Vol. 4.

[14] Qi He, Jian Pei, Daniel Kifer, Prasenjit Mitra, and Lee Giles. 2010. Context-aware citation recommendation. In *Proceedings of the 19th international conference on World wide web*. ACM, 421–430.

[15] Kalervo Järvelin and Jaana Kekäläinen. 2002. Cumulated gain-based evaluation of IR techniques. *ACM Transactions on Information Systems (TOIS)* 20, 4 (2002), 422–446.

[16] Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix factorization techniques for recommender systems. *Computer* 42, 8 (2009).

[17] Asher Levi, Osnat Mokryn, Christophe Diot, and Nina Taft. 2012. Finding a needle in a haystack of reviews: cold start context-based hotel recommender system. In *Proceedings of the 6th ACM conference on Recommender systems*. 115–122.

[18] Peter Lonsdale, Chris Baber, Mike Sharples, Will Byrne, Theodoros N Arvanitis, Pat Brundell, and Russell Beale. 2005. Context awareness for MOBIlearn: creating an engaging learning experience in an art museum. *Proc. MLearn 2004: Learning Anytime, Everywhere* (2005), 115–118.

[19] Nikos Manouselis and Constantina Costopoulou. 2007. Experimental analysis of design choices in multiattribute utility collaborative filtering. *International Journal of Pattern Recognition and Artificial Intelligence* 21, 02 (2007), 311–331.

[20] Nikos Manouselis, Hendrik Drachsler, Katrien Verbert, and Erik Duval. 2013. Survey and analysis of TEL recommender systems. In *Recommender Systems for Learning*. Springer, 37–61.

[21] Nikos Manouselis, Hendrik Drachsler, Riina Vuorikari, Hans Hummel, and Rob Koper. 2011. Recommender systems in technology enhanced learning. In *Recommender systems handbook*. Springer, 387–415.

[22] Nikos Manouselis, George Kyrgiazos, and Giannis Stoitsis. 2014. Exploratory study of multi-criteria recommendation algorithms over technology enhanced learning datasets. *Journal of e-Learning and Knowledge Society* 10, 1 (2014).

[23] Ante Odic, Marko Tkalcic, Jurij F Tasic, and Andrej Košir. 2012. Relevant context in a movie recommender system: Users' opinion vs. statistical detection. In *Workshop on Context-Aware Recommender Systems at ACM RecSys*, Vol. 12.

[24] Maria Soledad Pera and Yiu-Kai Ng. 2013. What to read next?: making personalized book recommendations for K-12 users. In *Proceedings of the 7th ACM conference on Recommender systems*. ACM, 113–120.

[25] Katrien Verbert, Nikos Manouselis, Xavier Ochoa, Martin Wolpers, Hendrik Drachsler, Ivana Bosnic, and Erik Duval. 2012. Context-aware recommender systems for learning: a survey and future challenges. *IEEE Transactions on Learning Technologies* 5, 4 (2012), 318–335.

[26] Yong Zheng. 2015. A revisit to the identification of contexts in recommender systems. In *Proceedings of the Conference on Intelligent User Interfaces Companion*. ACM, 133–136.

[27] Yong Zheng. 2017. Criteria Chains: A Novel Multi-Criteria Recommendation Approach. In *Proceedings of the ACM Conference on Intelligent User Interfaces*. 29–33.

[28] Yong Zheng. 2018. Context-Aware Mobile Recommendations By A Novel Post-Filtering Approach. In *Proceedings of the 31st International Florida Artificial Intelligence Research Society Conference*. AAAI.

[29] Yong Zheng. 2018. Exploring User Roles In Group Recommendations: A Learning Approach. In *Adjunct Proceedings of the ACM Conference on User Modelling, Adaptation and Personalization: 3rd Workshop on Human Aspects in Adaptive and Personalized Interactive Environments*.

[30] Yong Zheng. 2018. Identifying Dominators and Followers In Group Decision Making Based on The Personality Traits. In *IUI'18 Workshop on Theory-Informed User Modeling for Tailoring and Personalizing Interfaces*.

[31] Yong Zheng. 2018. Personality-Aware Decision Making In Educational Learning. In *Proceedings of the ACM Conference on Intelligent User Interfaces Companion*.

[32] Yong Zheng. 2019. Utility-Based Multi-Criteria Recommender Systems. In *Proceedings of 34th ACM SIGAPP Symposium on Applied Computing*. ACM.

[33] Yong Zheng, Robin Burke, and Bamshad Mobasher. 2014. Splitting approaches for context-aware recommendation: An empirical study. In *Proceedings of the 29th Annual ACM Symposium on Applied Computing*. ACM, 274–279.

[34] Yong Zheng, Bamshad Mobasher, and Robin Burke. 2014. CSLIM: Contextual SLIM recommendation algorithms. In *Proceedings of the ACM Conference on Recommender Systems*. ACM, 301–304.

[35] Yong Zheng, Bamshad Mobasher, and Robin Burke. 2015. CARSKit: A java-based context-aware recommendation engine. In *Data Mining Workshop (ICDMW), 2015 IEEE International Conference on*. IEEE, 1668–1671.