Fits Like A Game: a Multi-criteria Adaptive Gamification for Collaborative Location-based Collecting Systems

María Dalponte Ayastuy[1,2][0000-0002-1412-5694], Alejandro Fernández[2][0000-0002-7968-6871], and Diego Torres[1,2][0000-0001-9333-0532]

[1] Depto CyT, Universidad Nacional de Quilmes, R. Saenz Peña 352, Bernal, Buenos Aires, Argentina mdalponte@unq.edu.ar

[2] LIFIA, CICPBA-Facultad de Informática, Universidad Nacional de La Plata, 50 y 120, La Plata, Buenos Aires Argentina {alejandro.fernandez,diego.torres}@lifia.info.unlp.edu.ar

Abstract. This article proposes an adaptive gamification approach based on a Multi-Criteria Recommendation System (MCRS) for Collaborative Location-based Collecting Systems, adapting the gamification to each user, taking into account her preferences and the project's objectives as a multi-criteria scenario. Specifically, the potentially recommended items are dynamically generated gamification elements, and the recommendation criteria are defined considering two points of view: user preferences and project objectives. Finally, the article includes an evaluation of the proposal and then a discussion of the results.

Keywords: Adaptive Gamification · Multi-criteria Recommender Systems · Clustering-based Collaborative Filtering

1 Introduction

Citizen science encompasses a range of methodologies that encourage and support the contributions of the volunteers to the advancement of research and monitoring. Contributions may include co-identifying research questions; co-designing/ conducting investigations; co-designing, building, and testing low-cost sensors; co-collecting and analyzing data; co-developing data applications; and collaboratively solving complex problems [22]

Gamification is a widely used strategy to engage, retain users and direct their participation. It is about using game elements and mechanics in systems and domains that are not naturally games. Despite the rapid growth of the gameful design research area and the current level of success in user engagement, these findings cannot be generalized to all domains and all users. The one-size-fits-all approach presents several limitations because of the users' different motivations, personalities, needs, and playing styles [4, 11, 3]. Currently, the research stream on adaptive gamification is considering how to dynamically adapt the game elements and mechanics each user needs in each context. Citizen science projects may apply adaptive gamification to have greater participation from the general public and reach a higher project efficiency. The gamification approach could be adapted to the community members and the project's objectives to achieve better participation and sustained user engagement.

The research on adaptive gamification reveals two main adaptation strategies of game elements. On the one hand, the adaptation approach can recommend at different moments gamification elements corresponding to different types, depending on the estimated user preferences. On the other hand, the gamified system can adapt by adjusting a single gamification aspect according to the player's performance or behavior. Adapting a game element is a change in the features or traits of the specific gamification element. Adapting the game mechanic is an adaptation of the game that generates a rule change, mainly related to difficulty adaptation (i.e., adjusting the time factor, the reward, or enabling an action). [10]

Collaborative location-based collecting systems (CLCS) are collaborative systems where the community of users collects timestamped, geotagged data; this is usually done by using a mobile application [7]. CLCS are found in citizen science projects, such as the AppEar project[5], GeoVin [6], or iNaturalist [17]. In these CLCS-supported projects, volunteers carry out survey tasks (frequently known as *check-ins*) using mobile technologies that allow images, timestamps, and spatial coordinates to be associated with the reports. Usually, CLCS have specific space-time coverage objectives. For example a project may look to maximize coverage of a certain area, or making sure that samples are collected at all hours [7]. A possible approach to specify the needed coverage is to divide the territory into static areas and define a set of time restrictions, indicating how many samples are needed in each area and time. Any attempt to gamify a CLCS should have it's objectives in mind.

To implement an adaptive gamification approach, a strategy for the dynamic generation of game elements considering the user's profile and the current project's coverage objectives is needed. If the user profile is made up of, among other data, user preferences, these can be considered a set of criteria. Similarly, the objectives of the citizen science project can derive another set of criteria. In this way, the adaptation can be addressed through a multi-criteria recommendation system. Although there have been advances on the subject, such as the one presented in [16, 1], they do not consider either the spatial-temporal aspect that affects the user's profiling or the project's objectives.

This article proposes an adaptive gamification approach based on a Multi-Criteria Recommendation System (MCRS) for CLCS-supported projects, applying clustered-based collaborative filtering [2] technique. It adapts the gamification to each user, taking into account her preferences and the project's objectives as a multi-criteria scenario. Specifically, the potentially recommended items are dynamically generated gamification elements, and the recommendation criteria are defined considering two points of view: user space-time behavior and project objectives. Finally, the article includes an evaluation of the proposal and then a discussion of the results.

2 Related Work

Since it has been observed that users do not like repetition or uniformity, research has been conducted on the adaptation of gamification elements to users. Particularly, there are approaches of dynamic content generation to adapt the gaming experience based on the profile of the users' characteristics [8], but they still lack a strategy to incorporate user feedback.

One possible strategy could be the use of recommender systems, which takes from a repository of gamified items the one that the person might like the most -using explicit or implicit information about the user-, and there have already been scientific progress on this, such as the general framework for designing adaptive systems in [20] or the proposal in [23] that personalizes activity recommendations by a player model based on activity tracking. Nevertheless, these approaches do not fully meet the needs of citizen science projects as mentioned above, related to modeling spatial and temporal collaborative activities or project objectives.

On the other hand, the application of multiple criteria during the recommendation task has been explored in [16]. However, the application of a multi-criteria recommender system requires its adaptation to the dynamics of gamification, considering aspects such as reward and difficulty level in order to maintain the flow and the user's engagement [24].

From another point of analysis, one of the most used game elements in gamified collaborative systems is game challenges [10], wich is a task or problem in which difficulty depends on the user's skills, abilities, motivation, and knowledge and count toward progress and outcomes[13]. While there is a wide range of types of challenges detailed in the literature[21], to present a gamified application for CLCS, those challenges that require endurance faculties or those that require sustaining a temporality and rhythm must be considered. To develop challenges of this type in a personalized way, it is necessary to model users based on how they interact with the CLCS in terms of how they behave spatially and temporally [7].

3 Space-time game challenges domain

An example domain is used in the following sections to explain this approach, where a CLCS is gamified through game challenges generation and recommendation.

The space-time game challenges are actions a user must fulfill within an area and a time restriction. Particularly to CLCS, these actions are the previously mentioned sampling tasks, which are registered with a geographically referenced location and a timestamp. For instance, a possible game challenge is to gather two samples in area a_1 on a weekend morning. Additionally, the game challenge is also described with a difficulty estimation and a reward that can influence user preference.

Therefore, the following criteria set C_u describes user preferences using 5 aspects of the game element:

$$C_u = \{area, time\_restriction, difficulty, reward,$$
$$sample\_number\}$$

On the other hand, project priorities can be described through the game challenge's area and time restriction.

$$C_p = \{area, time\_restriction\}$$

The criteria set C_p allows prioritization of challenges associated with specific areas or time restrictions, thus approximating the project's objectives.

An example of a scoring matrix is shown in Table 1, where it can be seen, for example, that the game element ge_0 has been rated by users u_0 and u_2. User u_0 scores 2 points for the challenge's area, 3 points for its time restriction, and 2 points for difficulty, reward, and sample number. According to the project's objectives, both the area and the time restriction of game element ge_0 received a score of 5 points, meaning that it has a great priority.

	u_0	u_1	u_2	Global
ge_0	(2 3 2 2 2)	?	(3 3 4 5 2)	(5 5)
ge_1	(1 1 2 1 1)	(5 5 4 4 3)	?	(3 5)
ge_2	?	(2 3 3 4 4)	?	(1 4)
ge_3	(0 1 3 0 0)	?	(5 5 4 2 2)	(3 4)

Table 1. Multi criteria ratings with 5 user criteria and 2 project (global) criteria

The scoring in C_u may include explicit user preferences, e.g., multi-criteria scoring for game challenges, or implicit user preferences, such as selecting a game challenge from an ordered list recommended to the user. The values in C_p (global column in Figure 1) may describe the project's criteria in terms of priority sampling areas and time restrictions.

4 Problem statement and approach

Recommendation systems are software tools and techniques providing suggestions for items that are most likely to be interesting to the user or to be relevant to her needs. In computing these item suggestions, recommendation systems try to predict what the most suitable items are, based on the user's preferences. The system collects information from users regarding their preferences which are either explicitly expressed or are inferred by interpreting the actions of the user[18]. One possible approach is collaborative filtering, which is based on the fact that if the active user matched in the past with certain users, then the new recommendations coming from these similar users should be relevant and of interest to the active user.

The problem of adapting game elements in the context of collaborative location-based collecting systems is presented in terms of a Multi-Criteria Recommendation System (MCRS) where the recommended items are game elements, such as the game challenges that were presented in Section 3. MCRS are systems that use multiple criteria to support recommendation [2]. The performance of alternatives in the game elements set is analyzed upon a set of criteria that may refer to the multiple dimensions upon with the item is being evaluated. These criteria can be related to the game element's attributes but also to other aspects that may be interesting to the user or the project. This work considers two subsets of criteria: the one that expresses the (implicit or explicit) preferences of the user, namely C_u, and the one that expresses the project's criteria, namely C_p.

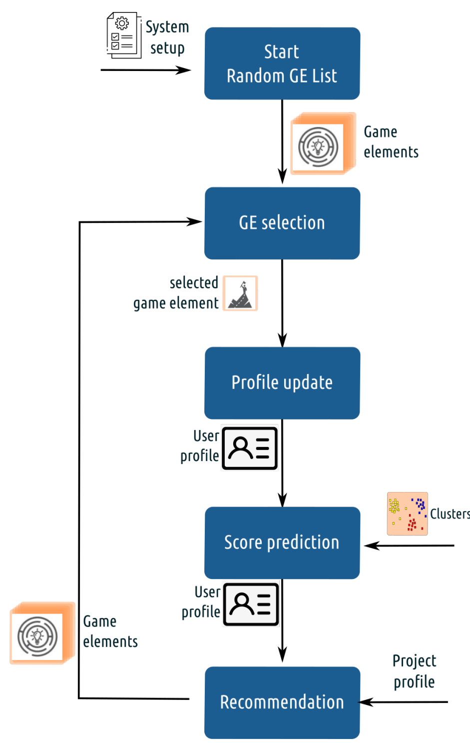

Fig. 1. Multi-criteria recommendation approach

This article presents a multi-criteria recommendation system based on a clustering based collaborative filtering technique with an item selection scoring, which is a less intrusive technique with respect to multi-criteria rating -where the user is asked to rate an item on multiple criteria-. The approach consists of 3 main steps. First, the scores of those elements that the user has not yet rated (unscored game elements) are predicted, calculating also a confidence level for each estimate that gives an idea about the quality and quantity of the available information was used to estimate. Secondly, a recommendation of the N best-scored elements is made, considering (or not) the confidence level. Finally, the user's opinion is incorporated, which is implicitly expressed by the selection of one of the recommended game elements. These steps are depicted through *score prediction*, *recommendation* and *profile update* boxes in figure 1. Notice that in the figure, the workflow begins with a random selection of game elements to present to the user a first recommendation.

The score prediction step is based on a clustering-based collaborative filtering, with space-time clusters. The recommendation step considers the previously estimated criteria scores and applies an aggregation function to have an unique overall score of each element for the target user and be able to build a game element ordered list. The profile update step uses a pairwise computation technique to update the predicted overall score given the user's opinion that is expressed by a game element selection from the ordered list.

This steps are further detailed in following sections.

4.1 Preliminar definitions

Definition 1 (Criteria score). *The score for an specific criteria c_i (for $c_i \in C_u$) in game element ge_j given by the user u_x is denoted as $r_{ijx} \in [0..5]$.*

Definition 2 (Game element score). *The score tuple for a game element ge_j given by the user u_x is defined as*

$$R_{jx} = < r_{1jx}, .., r_{kjx} >$$

where k is the size of set C_u.

To include multi-criteria rating information in the calculation of the similarity between two different users, k different similarity values are obtained by using a variation of the Manhattan distance. The overall similarity then can be computed by aggregating the individual similarities into an average function, as is explained in Definitions 3, 4 and 5.

Definition 3 (Scoring similarity). *The similarity between scoring R_{yx} and R_{yv} is defined as follows:*

$$sdist(R_{yx}, R_{yv}) = \frac{1}{k} \times \sum_{c=1}^{k} |r_{cyu} - r_{cyv}|$$

Definition 4 (Users distance). *The distance between users u and v is defined as follows:*

$$udist(u, v) = \frac{1}{|C(u, v)|} \times \sum_{g \in C(u,v)} sdist(R_{gu}, R_{gv})$$

where $C(u, v)$ is the set of common game elements between users u and v. This are the game elements that both u and v have rated.

Definition 5 (Users similarity). *The similarity between users u and v is a transformation over the notion of distance:*

$$usim(u, v) = \frac{1}{1 + udist(u, v)}$$

Definition 6 (Overall score). *The overall score of a game element given by a user is a value:*

$$O(u_x, ge_j) = f(R_{jx})$$

where f is a linear function, R_{jx} is the score for the game element ge_j given by user u_x.

It is well known that recommender systems exhibit significant user or item bias, which is explained by some users' tendency to give higher scores than others and some items to be rated higher than others [15]. Mathematically, the average rating per user can be expressed as the summation of nonzero ratings given by the user divided by the user's number of ratings. However, this formula does not consider the number of game elements the user rated. It puts on the same page users who have rated hundreds of game elements and users who rated only one game element. To correct this bias and give statistical significance, a C term is added to the denominator. The C term is called *shrimp term*, and it is a constant value chosen depending on the properties of the data.

Definition 7 (User bias). *The user bias is a description of how the user tends to score the game elements. It is computed as follows:*

$$\overline{u} = \frac{\sum_{ge \in R_u} O(u, ge)}{|R_u| + C}$$

where $O(u, ge)$ is the overall score of game element ge given by user u, R_u is the set of game elements rated by user u, C is the shrimp term.

Finally, in relation to the modeling of the user's space-time behavior, the last two definitions are needed. To synthesize the user activity within the time frame in a single value and thus be able to shape historical activity as a time series, a K-means clustering was executed in order to detect behavioral atoms [9]. The aforementioned behavioral atoms are generated from aggregating a set of sampling events in a given time interval.

Definition 8 (Behavioural Atom). *A behavioral atom is a categorical value that describes the user's space-time activity within a time frame. They are discovered through a clustering process over a sample task-derived data set.*

With these elements, the UTB (User Traveling Behaviour) series are composed, as described in the following definition.

Definition 9 (UTB). *The User Traveling Behaviour series for a user* $\boldsymbol{u}$ *is a sequence*

$$UTB_u = \{a_1, .., a_n\}$$

where each $\boldsymbol{a}_i$ *is a behavioral atom corresponding to the timeframe* $\boldsymbol{i}$*.*

These UTB time series can be grouped to give a notion of similarity between people based on their spatial and temporal behavior [7].

4.2 Game elements score prediction

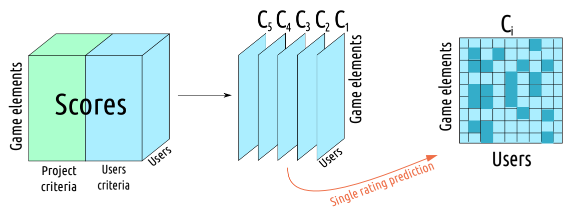

Fig. 2. Score prediction. Dark squares are the predicted scores

This multi-criteria score prediction problem can be approached as multiple predictions by taking each criterion as an independent dimension of estimation and then aggregating this data into an overall score per game element [1]. Considering the criteria in isolation allows the estimation of missing data using any one-dimensional approach (see Figure 2) and then integrating all the score matrices related to the different criteria using an aggregation strategy, as will be discussed in the recommendation step.

Using a **clustering-based collaborative filtering** approach, to estimate the score of an unknown item (here, game element) for a given user, the neighbors' scores must be considered. These neighbors are users belonging to the same cluster, which is computed based on their space-time behavior (the UTB time series introduced in Definition 9).

The Definition 10 presents the function $S()$, that estimates the individual criterium score in game element g for user u as a weighted average of the known scores of the k nearest neighbors in the cluster. The users' similarity -as defined in 5- is used to weigh the score of each of the other users.

Definition 10 (Adjusted weighted sum). *The adjusted weighted sum* S *is defined as follows*

$$S(u, i, g) = \overline{u} + [\sum_{v \in K} (r_{vig} - \overline{v}) \times usim(u, v)] \times \frac{1}{\Sigma_{v \in K} |sim(u, v)|}$$

where K *is the set of* k *nearest neighbors in the cluster,* $usim(u, v)$ *is a value of similarity between users* u *and* v*,* $\overline{u}$ *is the user bias, and* r_{vig} *is the known score or criteria* c_i *in the game element* g *for user* v*.*

Similarity matrix

	u_0	u_1	u_2
u_0	1	0.8	0.5
u_1		1	0.7
u_2			1

User biases

User	$\overline{u}$
u_0	0.75
u_1	1,76
u_2	1,56

Table 2. Sample scenario

For instance, assume the user similarities and user biases described in the table 2, and that K set is composed by u_1 and u_2. Then, the estimation of the score for criterion c_0 in the game element ge_0 for user u_0 is as follows:

$$sum = (r_{100} - \overline{u_1}) \times 0.8 + (r_{200} - \overline{u_2}) \times 0.5$$

$$S(u_0, c_0, ge_0) = \overline{u_0} + \frac{sum}{0.8 + 0.5}$$

Notice that r_{100} and r_{200} are the scores given by user u_1 and u_2 to criterium c_0 in game element ge_0, respectively.

In addition, the score corresponding to the other criteria of set C_u must be estimated. Once all the individual scores are estimated, it is possible to carry out the recommendation stage, as is described in the next section.

4.3 Game elements recommendation

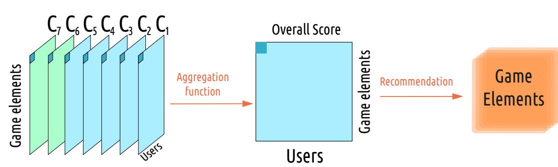

Fig. 3. Recommendation

When providing a recommendation for user u, it is necessary to estimate the value of the utility function R_{jx} for every game element ge_j for which it is undefined, and to choose game element ge_i that maximizes R_{ix}, or otherwise, to build an ordered list with top-scored game elements[14].

For this purpose, it is necessary to have an overall score from the individual scores assigned to the different criteria, which allows the establishment of a total order among the game elements. The aggregation function is the relationship between the overall rating and the underlying criteria scores $r_{0jx} = f(r_{1jx}, .., r_{kjx})$. Figure 3 highlights in dark blue color the score given by a player to each criterion within a game element.

There are many approaches to this, but in this work, the $O()$ function defined in 11 is used. It aggregates the values of the individual criteria into a linear function whose weights are approximated by linear regression. Each weight w_i is associated with criterion i and can be interpreted as the importance of this criterion in determining the overall rating.

Definition 11 (Aggregation Function).

$$O(u_x, ge_j) = [\Sigma_{i=1}^{k} w_i \times r_{ijx}] + c$$

where r_{ijx} *is the score of the criteria* i *in the game element* ge_j*, given by the user* u_x*.*

The weights w_i and constant c are estimated based on the set of known ratings through a linear regression in a batch process that is executed offline, according to the system configurations. The system could consider users who have had a certain load of participation to mark their weights outdated or 'dirty'. See section 4.5 for further details.

The output of this recommendation step is presented to the user as a descending score list of game elements, as defined below.

Definition 12 (Recommendation list).

$$L_u = [l_0..l_n]$$

where l_i *are game elements and* $O(u, l_i) > O(u, l_j)$ *for every* $i > j$*.*

The user has then the opportunity to choose one of them, thus expressing her preference and giving feedback to the system. The impact of this interaction is explained in the next section.

4.4 User feedback

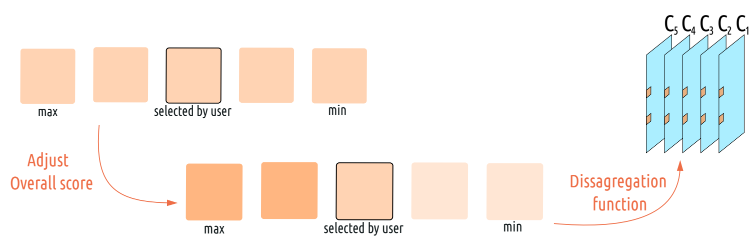

Fig. 4. User feedback: game element selection

Once the recommendation of game elements for a user has been resolved, the user can choose an element from the ordered list to express her preferences. The underlying idea is to derive the user's preferences by having her choose a game element from a list sorted by overall score in a descending order. This action makes it possible to correct if needed the estimates that established the order in the list of the recommended game elements. Since the chosen element may not be the best scored, it is possible to adjust the user's profile by considering that by such a choice the user is indicating that the previous items in the list (i.e., for which a higher score had been estimated) should be adjusted to have a lower score than the selected item. Formally, let be l_p the selected element at position p in the list. If $p > 0$ then there is a subset of elements $L' = [l_0..l_{p-1}]$ with the elements that are previous to l_p. The scores of the game elements in L' should be updated according to their relative position with respect to the selected item.

In this approach, a pairwise computation with non-transitive decomposition by difference is applied[16]. The main idea is to update the rating of the items in L' and finally disaggregating these changes into individual criteria preferences.

Asuming that the preference between an object **a** and an object **b** does not depend on the attributes in common of **a** and **b** (principle of preferential independence), the non-discriminating criteria between the selected object and an object above must be removed. The discriminating criteria (i.e. equally scored criteria) is the set defined in equation 1

$$\theta_{uij} = \{c \in C_u : r_{ciu} \neq r_{cju}\} \tag{1}$$

Listing 1.1. Pairwise computation

```
1 For prev: 0 to p
2     dcSet = discCriteria(gt,prev)
3     cNum = len(dcSet)
4     lpScore = partialScore(gt,dcSet)
5     prevScore = partialScore(prev,dcSet)
6     delta = prevScore - lpScore
7     critDelta = delta/cNum
8     alpha = 1/p
9     update_profile(prev,alpha, critDelta)
```

To update the overall score of elements in L' their distance to l_p must be computed, but only considering the discriminating criteria set (see line 2 in algorithm detailed in Listing 1.1).

With this set, a partial game element score can be computed through function O_θ defined in equation 2 and implemented in line 4 of Listing 1.1.

$$O_\theta(u,i) = \frac{\Sigma_{c\in\theta} r_{ciu}}{\#(\theta)} \tag{2}$$

In equation 3 the function $\Delta(u,i,g)$ is defined as the difference between the partial scores of game elements g_i and g_t. Notice that $\Delta(u,i,g)$ is positive in any case, given that g_i is first in the game element list than g_t.

$$\Delta(u,i,g) = O_\theta(u,i) - O_\theta(u,g) \tag{3}$$

Thus, Delta is used to obtain the score of each criterion composing the temporary vector, through the $delta()$ function defined in equation 4 (these are implemented in lines 6 and 7 of Listing 1.1).

$$\delta(u,i,g) = \frac{\Delta(u,i,g)}{\#(C_u)} \tag{4}$$

Finally, the user profile is updated as is described in equation 5. In order to maintain an efficient system -with a balanced impact- the number of pairwise computations that have been done need to be taken account. For this aim, the variable $\alpha = 1/pos(g)$ is used (lines 8 and 9 in Listing 1.1).

$$r'_{ciu} = r_{ciu} + 0,1 \times log(1 + \alpha \times \delta(u,i,g)) \tag{5}$$

Notice that α relativizes the change in the score according to the position of the chosen element. The more distant the element g_t is from the first element (which has a higher score), the smaller the *alpha* value. Indeed, a logarithmic function is used to reduce the impact of high values. A coefficient of 0.1 is used to control the impact of the new data on the system. Lastly, in order to avoid the negative values obtained with the logarithm, the value of 1 is added.

With this profile update, the system is able to make more suitable scoring predictions.

4.5 Batch processes

It is well known that the computation of user clusters as well as the distance between all users in each cluster is computationally demanding. That is why this processing must be able to be done offline to the workflow described in the Figure 1 and that this does not compromise the usability of the recommender system. The same policy is applied in the calculation of the weights that are part of the score estimation, as indicated in the Definition 11.

The situation that triggers the execution of each of these processes is an orthogonal configuration to the system that we propose here. For the case of clusters, in particular, a quality parameter such as mean square error or intra-cluster cohesion can be used.

5 Evaluation and results

In this preliminary version, while lacking a dataset with the needed characteristics related to playing behaviour and space-time information, the evaluation of this proposal had to focus on the collaborative filtering aspect. This was done using an existing dataset, for which a single cluster is considered, with the expectation that the current implementation behaves adequately according to the literature in evaluation of collaborative filtering recommender systems.

Examples of decision support metrics are **precision at k** and **recall**. The first one is the proportion of recommended items in the top-k set that are relevant, and the last one is the proportion of relevant items found in the top-k recommendations. Moreover, the F1-score measure is a harmonic mean of precision and recall.

Definition 13 (Precision at k).

$$P_k = \frac{(L_u \cap R_u)}{\#(L_u)}$$

Where L_u is the set of recommended game elements for active user u and R_u is the set of relevant game elements.

Definition 14 (Recall at k).

$$R_k = \frac{(L_u \cap R_u)}{\#(R_u)}$$

Where L_u is the set of recommended game elements for active user u and R_u is the set of relevant game elements.

Definition 15 (F1 score).

$$F1 = \frac{2 \times R_k \times P_k}{R_k + P_k}$$

Where R_k is the recall at k and P_k is the precision at k.

In the computation of precision at k, as the divisor can be a zero value, because the recommendation list can be empty, in that case the precision at k must be set to 1 [12, 19]. Similarly, when computing recall at k a similar situation can occur when the total number of relevant items is zero. In this case the recall at k is set to value 1.

Given that the development progress of this proposal is preliminary, the aspects to be evaluated are those related to accuracy and recall through offline tests using datasets with historical scoring information. The possibility of using synthetic data was discarded as it presents a significant risk of being biased to favor the algorithms and it is only recommended to use them in the performance tests of the tools [14].

To evaluate this approach, a set of test scenarios were developed, and for all of them, an adaptation of the MovieLens data set was carried out, which contains the scoring of 10681 movies given by 71567 users. The multi-criteria suitability was done replicating the score given by the user in each of the criteria of the set C_u. Relevant elements were also marked, this is, their score exceeds a certain threshold, to then separate 20 percent of the relevant records in a test set. The test scenarios are described below.

- Scenario A: For this test, N=10 iterations were performed where each time a user was taken at random and the score prediction and recommendation were performed considering k=6 neighbors within the cluster. The project criteria were not considered, i.e. the values are set to zero.
- Scenario B: Similar to A, but with random project's criteria scores.

The maximum precision at k obtained was 0.75 in scenario A and 0.9 in scenario B. The maximum recall value obtained was 0.2 in A, and 0.15 in B. The recall value can be seen as too low but is related to the evaluation parameters. As long as the size of the recommendation is limited to a few elements, it will always be much smaller than the full set of relevant game elements.

Recommendation systems have a variety of properties that may affect user experience, such as accuracy, robustness, scalability, and so forth, but the task of evaluating a recommender system must be based on the set of relevant properties for the application [19]. Particularly, numerous strategies have been proposed and used in the literature to evaluate the accuracy, including statistical accuracy metrics (e.g., mean absolute error and mean square error), as well as decision support system metrics that determine how well the recommender algorithm can predict high-relevance items (i.e., items that the user would rate highly)[12].

6 Conclusions and future work

This article presented a framework to adapt the gamification of a CLCS-supported project through a multi-criteria recommending system, with awareness of user's and project preferences/priorities. The proposed system satisfies the objective of this work, as it incorporates elements of gamification, the criteria of the CLCS project, and the possibility of incorporating space-time behavior in the prediction of preferences.

The evaluation of the approach, as was mentioned, is limited to offline test scenarios, focused in collaborative filtering performance metrics. To carry out evaluations with real users, it is necessary in the first place to implement this proposal in a citizen science project that will allow to collect space-time data.

The disaggregation function approach supports the implicit formulation of the user's preference model based on the selection of a game element from a list of recommended. This is a less intrusive technique concerning multi-criteria rating, where the user is asked to rate an item on multiple criteria. Nevertheless, the possibility of multi-criteria scoring is a future work, to allow users to provide the feedback about the game element on specific criteria. Finally, another pending but very promising work is to improve the gamification strategy, incorporating the increase of difficulty and reward as part of the project's objectives based on the user's progress on the selected challenge.

References

1. Adomavicius, G., Kwon, Y.: Multi-criteria recommender systems, pp. 847–880. Springer US (Jan 2015). https://doi.org/10.1007/978-1-4899-7637-6_25
2. Adomavicius, G., Manouselis, N., Kwon, Y.: Multi-Criteria Recommender Systems, pp. 769–803. Springer US, Boston, MA (2011). https://doi.org/10.1007/978-0-387-85820-3_24, https://doi.org/10.1007/978-0-387-85820-3_24
3. Busch, M., Mattheiss, E., Hochleitner, W., Hochleitner, C., Lankes, M., Fröhlich, P., Orji, R., Tscheligi, M.: Using player type models for personalized game design - an empirical investigation **28**, 145–163 (01 2016)

4. Böckle, M., Novak, J., Bick, M.: TOWARDS ADAPTIVE GAMIFICATION: A SYNTHESIS OF CURRENT DEVELOPMENTS. Research Papers (Jun 2017), https://aisel.aisnet.org/ecis2017_rp/11
5. Cochero, J.: Appear: A citizen science mobile app to map the habitat quality of continental waterbodies. Ecologia Austral **28**, 467–479 (08 2018)
6. Cochero, J., Pattori, L., Balsalobre, A., Ceccarelli, S., Marti, G.: A convolutional neural network to recognize chagas disease vectors using mobile phone images. Ecological Informatics **68**, 101587 (2022)
7. Dalponte Ayastuy, M., Torres, D.: Relevance of non-activity representation in traveling user behavior profiling for adaptive gamification. In: Proceedings of the XXI International Conference on Human Computer Interaction. Interacción '21, Association for Computing Machinery, New York, NY, USA (2021). https://doi.org/10.1145/3471391.3471431, https://doi.org/10.1145/3471391.3471431
8. Dalponte Ayastuy, M., Torres, D., Fernández, A.: A model of adaptive gamification in collaborative location-based collecting systems. In: Degen, H., Ntoa, S. (eds.) Artificial Intelligence in HCI. pp. 201–216. Springer International Publishing, Cham (2022)
9. Dalponte Ayastuy, M., Torres, D.: Adaptive gamification in collaborative location collecting systems: a case of traveling behavior detection. Journal of Computer Science and Technology **22**(1), e05 (Apr 2022). https://doi.org/10.24215/16666038.22.e05, https://journal.info.unlp.edu.ar/JCST/article/view/1943
10. Dalponte Ayastuy, M., Torres, D., Fernández, A.: Adaptive gamification in Collaborative systems, a systematic mapping study. Computer Science Review **39**, 100333 (2021). https://doi.org/https://doi.org/10.1016/j.cosrev.2020.100333, https://www.sciencedirect.com/science/article/pii/S1574013720304330
11. Heeter, C., Magerko, B., Medler, B., Lee, Y.H.: Impacts of forced serious game play on vulnerable subgroups. Int. J. Gaming Comput. Mediat. Simul. **3**(3), 34–53 (Jul 2011). https://doi.org/10.4018/jgcms.2011070103, https://doi.org/10.4018/jgcms.2011070103
12. Herlocker, J.L., Konstan, J.A., Terveen, L.G., Riedl, J.T.: Evaluating collaborative filtering recommender systems. ACM Transactions on Information Systems (TOIS) **22**(1), 5–53 (2004)
13. Iversen, S.: In the double grip of the game: Challenge and Fallout 3. Game Studies **12** (2012), http://www.gamestudies.org/1202/articles/in_the_double_grip_of_the_game
14. Jannach, D., Zanker, M., Felfernig, A., Friedrich, G.: Recommender systems: an introduction. Cambridge University Press (2010)
15. Koren, Y., Rendle, S., Bell, R.: Advances in collaborative filtering. Recommender systems handbook pp. 91–142 (2021)
16. Martin, A., Zarate, P., Camillieri, G.: A Multi-Criteria Recommender System Based on Users' Profile Management, pp. 83–98. Springer International Publishing, Cham (2017). https://doi.org/10.1007/978-3-319-39292-9_5, https://doi.org/10.1007/978-3-319-39292-9_5
17. Nugent, J.: inaturalist. Science Scope **41**(7), 12–13 (2018)
18. Ricci, F., Rokach, L., Shapira, B.: Recommender systems: introduction and challenges. Recommender systems handbook pp. 1–34 (2015)
19. Shani, G., Gunawardana, A.: Evaluating recommendation systems. Recommender systems handbook pp. 257–297 (2011)

20. Tondello, G.F., Orji, R., Nacke, L.E.: Recommender systems for personalized gamification. In: Adjunct publication of the 25th conference on user modeling, adaptation and personalization. pp. 425–430 (2017)
21. Vahlo, J., Karhulahti, V.M.: Challenge types in gaming validation of video game challenge inventory (CHA). International Journal of Human-Computer Studies **143**, 102473 (2020). https://doi.org/https://doi.org/10.1016/j.ijhcs.2020.102473, https://www.sciencedirect.com/science/article/pii/S1071581920300756
22. Vohland, K., Land-Zandstra, A., Ceccaroni, L., Lemmens, R., Perelló, J., Ponti, M., Samson, R., Wagenknecht, K.: The science of citizen science. Springer Nature (2021)
23. Zhao, Z., Arya, A., Orji, R., Chan, G.: Effects of a personalized fitness recommender system using gamification and continuous player modeling: System design and long-term validation study. JMIR Serious Games **8**(4), e19968 (Nov 2020). https://doi.org/10.2196/19968, http://games.jmir.org/2020/4/e19968/
24. Zichermann, G., Cunningham, C.: Gamification by design: Implementing game mechanics in web and mobile apps. " O'Reilly Media, Inc." (2011)