A Model of Adaptive Gamification in Collaborative Location-Based Collecting Systems

María Dalponte Ayastuy[1,2](✉), Diego Torres[1,2], and Alejandro Fernández[2]

[1] Depto CyT, Universidad Nacional de Quilmes, Bernal, Argentina
{mdalponte,diego.torres}@unq.edu.ar
[2] LIFIA, CICPBA-Facultad de Informática, Universidad Nacional de La Plata, La Plata, Argentina
{mdalponte,diego.torres,alejandro.fernandez}@lifia.info.unlp.edu.ar

Abstract. Gamification is a widely used resource to engage and retain users. It is about the use of game elements and mechanics in systems and domains that are not naturally games. Nevertheless, the usage of gamification does not always achieve the expected results due to the too much generalized approach that makes invisible the different motivations, characteristics and playing styles among the players. Currently, research on adaptive gamification deals with the gamification that each particular user needs at a particular moment, adapting gamification to users and contexts. Collaborative location-based collecting systems (CLCS) are a particular case of collaborative systems where a community of users collaboratively collect geo-referenced data. This article proposes an adapted gamification approach for CLCS, through the automatic game challenge generation. Particularly a model of user profile considering the space-time behavior and challenge completion, a model for the different types of challenges applicable in CLCS, a model for the CLCS objectives and coverage, and a strategy for the application of Machine Learning techniques for adaptation.

Keywords: Adaptive gamification · Collaborative location-based collecting systems · Game challenge

1 Introduction

Collaborative location-based collecting systems (CLCS) are a particular case of collaborative systems where a community of users collaboratively collect geo-referenced data [5]. CLCS bundle data records into datasets following a specific data schema that typically includes geographic coverage, submission date, creator, and data quality requirements [14]. CLCS frequently require the user to visit specific location to fulfill the sampling, and consequently, they are implemented with mobile applications.

H. Degen and S. Ntoa (Eds.): HCII 2022, LNAI 13336, pp. 201–216, 2022.
https://doi.org/10.1007/978-3-031-05643-7_13

Gamification (i.e., the usage of game elements in non-game contexts [6]) can be applied in CLCS as a strategy to attract participants, to sustain participation, and to motivate desired behaviours. However, gamification cannot be generalized to all users because of the different users' profiles, preferences, and playing contexts so it needs to be tailored to each one. [3,7,12,17]. Moreover, player engagement tends to decrease as the playing time passes [8], and so the desired behaviors must be reinforced [19].

The gamification of CLCS applies game elements related to space-time aspects to reach the project space-time objectives. This means to motivate the user community to collect data at certain times and places and to sustain that motivation over time. For example, the AppEar [4] citizen science project aims to survey the coasts of rivers, lakes and estuaries. For AppEar, it is important to ensure that the community visits certain geographic areas at certain times and do so with sufficient redundancy. Similarly, the iNaturalist project defines the "City campaign challenge" to promote in a specific city the collection of biodiversity data [11,16] within a certain period of time. This means that the sampling task has spatial and temporal conditions. Moreover, each different project should have specific criteria about the quality of sampling task. In some cases it may be important to achieve a certain sampling density and in other cases it is necessary to achieve a certain level of coverage.

One of the most used game elements in gamified collaborative systems is challenges [2]. A game challenge is a task or problem whose difficulty depends on the user's skills, abilities, motivation, and knowledge [9]. The player completes challenges for different reasons, but mainly because challenges allow to progress in the game and to get results (reach new levels, earn points, etc.). Although the skills, abilities, motivation of users vary, challenges (as game elements) are not frequently tailored.

There is a wide range of types of challenges detailed in the literature [21]. Particularly, those that require endurance or those that require commitment and rhythm can be mentioned, meaning that a space-time constraint must be set. Therefore, a strategy for adapting gamification in CLCS is to build game challenges tailored both to the player's space-time behavior as well to the needs of the CLCS.

The approach of Khoshkangini et al. [10] proposes a mechanism for generating personalized game challenges for each player at all times, based on their individual historical performance but also that of the community. They propose an automatic generation of game challenges using machine learning techniques that is personalized for the history and habits of each player and contextualized to their game state. Indeed, Khoshkangini et al. model and use it as input for adapting the policies or objectives intended to promote.

However, Khoshkangini et al. approach does not take into account the space-time behavior of the users, that is, when and where they interact with the CLCS. The work in [5] proposes a strategy to model the space-time behavior as a time series of behavioral atoms, where each behavioral atom synthesizes in a categorical value the intensity of gameplay based on the elicited frequency of interactions and the movement pattern within a time frame. The sequence of atoms potentially allows to identify space-time behavioral patterns shared between people and thus determine a criterion of similarity between them.

This article proposes an approach for adapting game challenges to the user and the game's objectives in a CLCS using Machine Learning strategies. This approach consists of four elements: 1) a representation of user profile considering the space-time behavior, and the challenge completion, 2) a model for the different types of challenges applicable in CLCS, 3) a model for the CLCS objectives and coverage, and 4) a strategy for the application of Machine Learning techniques for adaptation.

Also, this article presents an extension proposal of the approach in [10] specifically for CLCS, with two traits. On the one hand, using space-time goals, focusing in collecting tasks, and on the other hand modelling static as well dynamic project's objectives. A static objective is represented by goals' weights, and the dynamic objectives are computed goals priorities, considering a set of quality criteria over the sampling task. For instance, the quality criteria can be expressed in terms of collecting density (how many samples must an area have) or collecting coverage (how many different areas need to be sampled).

This article is organized as follows. In next section, a review of related work is presented. In Sect. 3 a concepts background is developed. The approach of CLCS game challenges recommendation is detailed in Sect. 4: and finally the Sect. 5 and 6 shares conclusions and future work.

2 Related Work

Research has been done on adapting gamification elements to the users, and it has been observed that users do not like repetition or uniformity, so dynamic content generation is considered to adapt the gaming experience based on the profiling of users' characteristics [15].

Among the research works that propose gamification adaptation mechanisms, it is possible to find approaches where the player profile (or the playing style) is statically modeled, or where an idea of a dynamic profile is implicitly derived and automatically adjusted over time. Among this first group the works in [10,13] can be included. On the other hand, there are studies that build a player profile based on a classification, usually determined by a questionnaire, to associate them with game preferences (game dynamics). For example, the work of [18] considers user's characteristic by means of Hexad scale [20] and relates it to the optimal game element from the known relationship between each player archetype and the game mechanics.

Contemporary video games frequently use procedural content generation to dynamically create new game elements and consequently grow the game diversity and the gaming experience, that can keep players engaged with the game. This can be used to dynamically adapt the game to player's preferences, skills and playing style [15].

3 Background

As was previously introduced, in a CLCS the community of users collaboratively collect geo-referenced data in a domain-specific data structure that includes, at least: geographic coordinates, submission date, creator, and sample data.

Some objectives of the CLCS can be: to summon a significant number of people/volunteers, to motivate certain behavior in people (for example, to make them travel in a certain manner, to make them participate in a sustained way), to reach a certain sampling quality level, get different samples of different people over the same area (to have different points of view), among others. Beyond the fact that these systems may have other objectives, those mentioned above are related to the space-time aspect, and particularly the present work focuses on the objective of reaching a target sampling quality level.

To specify the quality criteria that each CLCS needs to apply, a set of geographic areas and a set of temporal restrictions are defined to group the sampled data. This allows to express the collecting coverage or collecting density (presented in Sect. 1) in terms of the number of samples required for a given area, which meet a given time constraint. As an example, consider the Fig. 1, where areas are described as black boxes, the green dots describe weekend samples and yellow dots describe weekdays samples.

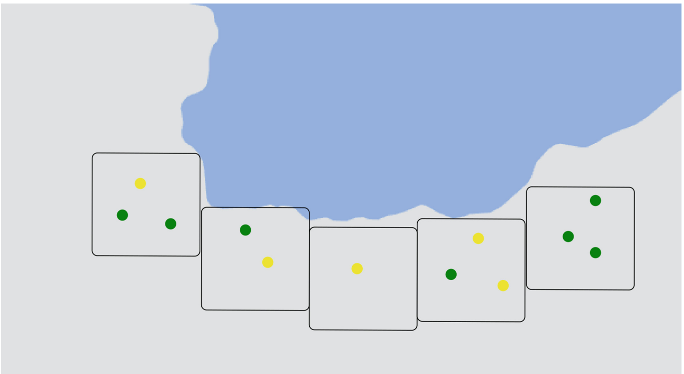

Fig. 1. Sampling example (Color figure online)

In this example, a coverage quality criteria could be to have five yellow samples and five green samples on each area. It can be understood observing the Fig. 1 that in the most left area four weekday samples and 3 weekend samples are still needed.

This section includes definitions that are necessary to understand the approach that is addressed in Sect. 4. Firstly, the definitions related to *Space-time Behaviour* are presented. Later the *Game Challenge* data structure is defined.

3.1 Space-Time Behaviour Definitions

Definition 1 (Sample). *The sample data of a CLCS is a tuple:*

$$SD = <u, T, LL, D>$$

where `u` *is de user,* `T` *is the sample timestamp,* `LL` *is the sample geographic coordinates (Latitude and longitude), and* `D` *is the domain-specific data (i.e., the content of the sample).*

The work in [5] presents an approach to the users' space-time profiling. As a first step to describe the space-time behavior of the user, the samples are grouped within timeframes, and based on the aggregation of these samples, a clustering technique is applied. This clusters represents the different interaction intensities within a time frame.

Definition 2 (Timeframe). *A time frame is identified by an integer value:*

$$t_i \in [1...n] \subset \mathbb{Z}$$

where $t_i < t_j$ *if* $i < j$

Definition 3 (Behavioural Atom). *A behavioral atom is a categorical value that describes the user's interaction with the CLCS within a time frame from a sample set [5].*

Notice that the input data is not modeled explicitly in the atom, because each possible atom value represents an abstraction of the space-time behavior in a given time period. Particularly, the clusters on a dataset of Fousquare application in New York city [1] gave rise to four atom types, read as Low, Medium, High and Max.

Definition 4 (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* $\boldsymbol{i}$*.*

3.2 Game Challenge Definitions

Goals characterize the sampling task through a spatial condition, a temporal condition, and a number of samples. Both in the spatial and temporal sense it is possible to define a discrete set of scenarios to characterize the sample. That is, define a set of geographical areas of interest to the CLCS and a set of time intervals that segment the temporal universe according to the sampling needs of the CLCS. To give an example, some system might need to break the time into weekdays vs. weekend days, and other might need smaller segments (morning, afternoon, or night), or more specific combinations. In addition, the goals define a number of samples that must be carried out fulfilling these space-time conditions. With this structure, a goal could be: "take 3 samples in zone number 1 between December 13 and 20". The objective of this number of samples field is to allow the *behavior improvement* of the users, growing this value through recommendations, based on the playing history.

Definition 5 (Goal). *The goal is a tuple:*

$$G = <SA, TR, \#S>$$

where `SA` *is the identification of an sampling area, and* `TR` *is a discrete value that describes a time restriction (interval) when the goal must be completed, and* `#S` *is the sample number that must be done in area* `SA` *and interval* `TR`*.*

A challenge goal can be, for instance:

g_1 : To complete one (`#S = 1`) sample in area 50 (`SA = 50`) on monday (`TR = monday`).
g_2 : To complete two (`#S = 2`) samples in area 6 (`SA = 6`) on a weekend day (`TR = weekend`).

Definition 6 (Game Challenge). *The Game Challenge is a tuple:*

$$GC = <u, g, d, r, w, i>$$

where `u` *is the user,* `g` *is the goal,* `d` *is the estimated difficulty category, and* `r`*,* `w` *and* `i` *are numbers that represents respectively the goal's computed reward, the weight and the percentage of improvement. The* **w** *value is the relevance that is statically configured for each goal.*

For instance, the game challenge $gc_1 = <alex, g_1, medium, 50, 8, 25>$ can be read as user Alex must complete goal g_1, which has an estimated *medium* difficulty for Alex, a reward of 50, a project relevance weight of 8 and an improvement of 25%. Another example is the game challenge $gc_2 = <chris, g_1, low, 20, 8, 50>$ that represents that the user Chris must complete goal g_1, which has an estimated *low* difficulty for Chris, a reward of 20, a project relevance weight of 8 and an improvement of 50%. With these two examples it can be seen that the same challenge goal has different difficulty, reward and improvement for two different users.

Definition 7 (Playing History). *The playing history of a user u is a list of tuples:*

$$PH_u = \{<gc, a>\}$$

where `gc` *is the game challenge and* `a` *is a real value in range $[0...1] \subset R$ that describes the challenge achievement*

Note that even though in Definition 7 the value `A` is presented as a real value in range [0...1], this allows to represent also the discrete Boolean values 0 and 1, that are useful for those goals that do not allow a partial completion (they are fully completed or not completed at all).

For instance, the playing history $ph_1 = \{<g_1, 1>, <g_2, 1>, <g_3, 0>, \}$ indicates that the user completed the first two challenges (g_1 and g_2) but not the third one (g_3).

Definition 8 (User profile). *The representation of user profile considering the traveling behaviour, the challenge preferences and completion is the following tuple:*

$$UP_u = <UTB_u, PH_u>$$

3.3 System Setup Definitions

The CLCS domain specification requires the setup of the samples areas set, the time restrictions set, the improvement scale, the static goal weights, the prize table and the area coverage requirements.

Definition 9 (Areas set). *The areas set is a set of integer values:*

$$a_i \in \mathbb{Z}$$

Definition 10 (Time restrictions). *The time restrictions are a set of categorical values:*

$$tr_i \in String$$

Definition 11 (Improvement scale). *The improvement scale is a set of percentage values:*

$$bi_i \in [0...100] \subset \mathbb{Z}$$

Definition 12 (Prize table). *A prize table is a list of tuples:*

$$prizes = \{<a, tr, d, i, p>\}$$

where ***a*** *is a sampling area,* ***tr*** *is a time restriction,* ***d*** *is the difficulty,* ***i*** *is the improvement and* ***p*** *is an integer number representing the prize for challenges with area* ***a****, time restriction* ***tr****, difficulty* ***d*** *and improvement* ***i****.*

Definition 13 (Static Weights). *The goals weights is a list of tuples:*

$$weights = \{<a, tr, w>\}$$

where ***a*** *is a sampling area,* ***tr*** *is a time restriction and* ***w*** *is the corresponding weight.*

Definition 14 (Required coverage). *The CLCS required coverage is a list of tuples:*

$$requiredCoverage = \{<a, tr, rs>\}$$

where ***a*** *is a sampling area,* ***tr*** *is a time restriction and* ***rs*** *is an integer number representing the required number of samples in area* ***a*** *within time restriction* ***tr****.*

As an example, if a coverage quality criteria is to have, in each area, five samples for *weekday* time restriction and three samples for *weekend*, then the required coverage is:

$$\{<a_1, weekday, 5>, <a_1, weekend, 3>, ..., <a_3, weekend, 3>\}$$

4 CLCS Automatic Game Challenge Recommendation

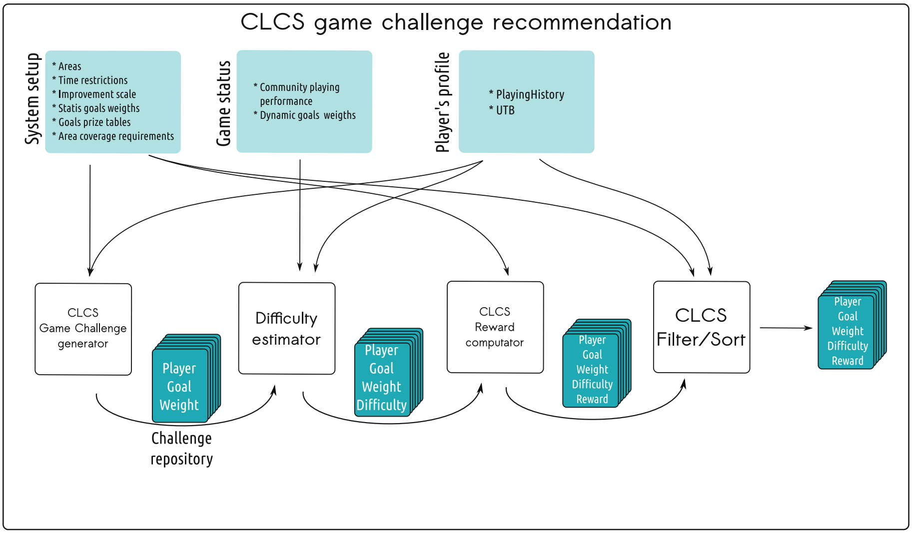

Fig. 2. Framework for CLCS game challenge recommendation

This game challenge recommendation is based on the generation of game challenges tailored to users within the restrictions and objectives presented by each CLCS. It can be seen as a pipeline process made up of 4 main processes which input is the user u_1, and the output is an ordered list of game challenges. The first step is the *challenge repository population*, when all the possible challenge goals that present to the user u_1 a behavioral improvement are built. Also, these challenges must be related to the CLCS specific requirements.

The second step is the *goal difficulty estimation* for each game challenge in the repository populated in the previous step, considering the user u_1 playing history and the community performance in relation to the game challenge goal. The third step is the *reward computation* for each game challenge, where given it's goal, difficulty value, and behavior improvement percentage, a reward amount is obtained. This is done considering the goal's prize table and the dynamically computed system priorities. Finally, the *filtering and sorting step* is done, where the all the potential game challenges are ordered by difficulty, reward and weight, to be recommended to the user u_1. This steps are depicted in Fig. 2.

As was mentioned in the introduction, the recommendation system presented in [10] can be suited to CLCS needs. Therefore this article proposes an extension that takes into account the characteristics and needs of the CLCS and incorporates the idea of space-time behavior modeled by the UTB. Specifically, it models an CLCS goal, extends the game status considering the sample quality related objectives of CLCS, and incorporates the dynamic calculation of goal weights and the users' UTB series at the moment of filtering and ordering the game challenges. To achieve this, the system setup provides a set of geographic areas and a set of time restrictions for the sampling goal.

As an example, a system setup is detailed in Table 1, with 3 geographical areas (a_i), 2 time restrictions (weekday and weekend) and 2 behavioral improvements (50% and 100%). In addition, goals are assigned with a weight that is configured in the system setup, and that allows a priority or relevance to be statically assigned to certain goals (certain areas in certain time restrictions). Also, the prizes tables for the goals $<a_1, weekday>$ and $<a_3, weekend>$ (area a_1, with *weekday* and area a_3 with *weekend* restriction) is shown.

Table 1. Example of system setup

Property	Values
Sample area (SA)	$\{a_1, a_2, a_3\}$
Time restriction (TR)	$\{weekday, weekend\}$
Improvement scale	$\{50\%, 100\%\}$

Goals weights

	a_1	a_2	a_3
Weekday	8	10	12
Weekend	6	8	9

Prizes tables

$<a_1$,weekday$>$		
dif\imp	50%	100%
Easy	100	125
Medium	133	156
Hard	166	186
Very hard	197	225

(...)

$<a_3$,weekend$>$		
dif\imp	50%	100%
Easy	111	130
Medium	144	161
Hard	177	192
Very hard	211	230

4.1 Challenge Repository Population

Table 2. Step 1.a: game challenge repository population

sa	tr	i	w
a_1	Weekday	50%	8
a_1	Weekday	100%	8
a_1	Weekend	50%	6
a_1	Weekend	100%	6
$\vdots$			
a_3	Weekend	50%	9
a_3	Weekend	100%	9

In the first step, the CLCS Game Challenge generator generates an initial repository of game challenges from all the combinations of areas and time restriction that are configured through the system setup. The game challenges must also suppose a level of improvement in the player's behavior that is forced by the $\#s$ parameter (see Definition 5), so each generated goal is replicated to be combined with each level of improvement configured by system setup. Considering the system setup example of Table 1, there are 12 combinations (see Table 2: 3 areas $\times$ 2 time restriction $\times$ 2 improvement scales). The *weight* field is filed out with the corresponding value in the system setup: $W(a_1, weekday) = 8$ (see Table 1).

The value for $\#s$ parameter is completed considering the playing history of the user. For instance, if in the last period the player could complete the goal $g_1 = <a_1, weekday, 2>$ (2 samples), then a 50% improvement means completing $\#s$ with the value 3, and a 100% improvement, the value 4. On the other hand, if the player has no previous activity on a given goal, this field is filled with 1. As an example consider that alex had solved g_1 and g_2:

$$g_1 = <a_1, weekday, 2>$$

$$g_2 = <a_2, weekend, 1>$$

With this historic input, $\#s$ parameter in the Game Challenge repository is filled as shown in Table 3). Notice that are only showed the goals based on g_1, in first and second row, with 50% and 100% respectively, and g_2 in third and fourth row, with 50% and 100%.

4.2 Challenge Difficulty Estimation

This module estimates the difficulty of each challenge c_1 in the repository for user u_1. The difficulty estimation has, as a central element, placing the performance of the user u_1 in the context of the performance of the community. For this, the

Table 3. Step 1.b: improvement application

sa	tr	i	w	#s
a_1	Weekday	50%	8	3
a_1	Weekday	100%	8	4
$\vdots$				
a_2	Weekend	50%	8	2
a_2	Weekend	100%	8	2
$\vdots$				

historical information is limited to those challenges associated with the same goal g_{c1}. Then the other users who have already solved the target challenge c_1 are identified and their distance from u_1 is quantified. Challenge's difficulty is defined as a categorical value between Easy, Medium, Hard and Very Hard, and is calculated from the distance between u_1 and the other players who had completed the target challenge c_1.

Particularly, with the aim of better adapting the challenge c_1 to capabilities or mobility style of the user u_1, he is placed in the context of a segment of the community, made up of people who have similar space-time behavior to that of u_1. The contribution of this article at this point is to narrow the community to this segment, by means of an unsupervised clustering on the UTB series of the entire community. This clustering allows the categorization of the users according to their space-time behavior as is modeled in the approach presented in [5]. Particularly, the difficulty estimation presented in [10] is adapted for the input that contains only the u_1's neighborhood. This adaptation takes into account a most specific context to tailor the challenge difficulty based on similar users with equivalent space-time behavior.

Table 4. Step 2: challenge difficulty estimation

sa	tr	i	w	#s	d	r
a_1	Weekday	50%	8	3	Medium	
a_1	Weekday	100%	8	4	Medium	
$\vdots$						

For instance, consider that user Alex has in his playing history a completed challenge with goal: $g_1 = <a_1, weekday, 2>$. And considering a 100% improvement, the target goal is: $g_2 = <a_1, weekday, 4>$. On the other hand, comparing de current performance ($\#s = 2$) with the performance of his neighbors in relation to the goal $<a_1, weekday>$, his playing history is not so far from the zone where g_2 is. This means that the estimated difficulty for this challenge is *Medium*.

4.3 Challenge Reward Computation

Beyond representing a game element, the game challenge's reward is a vehicle to motivate the desired behavior, and particularly to meet the defined quality criteria in the CLCS setup. With this objective, in the goal reward computation step, the prizes tables that are statically defined in the system configurations are combined with the current goals' coverage. To compute the reward field, the static baseline reward is weighted with a computed goal priority, as is described in the following equation:

$$r_c = prize(g_c, i_c, d_c) \times w_c \tag{1}$$

where *prize* is a function that obtains from the prizes table the statically configured value for goal g_c, improvement i_c and difficulty d_c, where g_c, i_c and d_c are the goal, improvement and difficulty of challenge c respectively.

The value w_c is the dynamically computed goal weight, defined in the Eq. 2, which considers the required coverage and the current coverage of the goal g_c.

$$w_c = \frac{reqCoverage(a_c, tr_c)}{currentCoverage(a_c, tr_c)} \tag{2}$$

where $currentCoverage(a_c, tr_c)$ is an integer value that represents the sampling status in area a_c and time restriction tr_c, and $reqCoverage(a_c, tr_c)$ is the integer value corresponding to the configured coverage requirement for area a_c and time restriction tr_c. Notice that, while $reqCoverage(a_c, tr_c)$ is greater than $currentCoverage(a_c, tr_c)$ -which is fulfilled from the beginning of the game- w_c is a value that overscales $prize(g_c, i_c, d_c)$, and when $currentCoverage(a_c, tr_c)$ reaches $reqCoverage(a_c, tr_c)$, the value w_c starts to underscale $prize(g_c, i_c, d_c)$. This means that as long as the quality level is not reached, the reward is greater to motivate the challenge to be met.

Table 5. Required coverage configuration

A	TR	Required samples
a_1	Weekday	5
a_1	Weekend	5
⋮		

As an example, consider the required coverage configuration described in Table 5, the defined prizes in Table 1, and the scenario introduced by Fig. 1, where area a_1 had one sample on a weekday (yellow dot) and two weekend samples (green dots). The computed reward for a challenge $c_1 = <alex, g_1, medium, r, 8, 50>$, bounded to goal $g_1 = <a_1, weekday>$, needs to compute the dynamic goal weight (w_{c1}) as follows:

$$w_{c1} = \frac{reqCoverage(a_1, weekday)}{currentCoverage(a_1, weekday)} = \frac{5}{1} = 5$$

Secondly, this value is used in Eq. 1:

$$r_{c1} = p(g_1, 50, medium) \times w_{c1} = 133 \times 5 = 665$$

This formula is applied to the repository as is shown in Table 6.

Table 6. Step 3: reward calculation

sa	tr	i	w	#s	d	r
a_1	Weekday	50%	8	3	Medium	665
a_1	Weekday	100%	8	4	Medium	780
⋮						

The computation presented here considers the required and current level of coverage, through the criteria described in Eqs. 1 and 2, but it is important to note that it is not the only way to incorporate the objectives of the project in the computation of the challenge reward.

4.4 Challenge Sorting

This is the last stage of the CLCS game challenge recommendation process, where the different variables of the game challenges are taken into account to present the user with the elements of the repository in a certain order. There is not a single order criterion, and in particular it is important to consider the 2 dimensions that are proposed: the objectives of the CLCS and the space-time behavior in the definition of community.

In our proposal this is modeled through different data and processes. On the one hand, in relation to the CLCS objectives, there are the static goals weights, which are represented as a variable in the game challenge tuple, and the dynamic objectives that are calculated as was described in the Eq. 2. On the other hand, in relation to the space-time behavior of the users, there are the difficulty value (which considers the UTB series of the community) and the reward value, which takes into account the difficulty and, transitively, the UTB series.

Therefore, a possible strategy is to sort the set of challenges by least difficulty, then highest reward, then static weight. Considering the repository described in Table 7, the challenges order is: $\{c_2, c_3, c_1\}$.

However, the sorting strategy's efficiency must be measured, considering user acceptance through the challenge completion rate.

Table 7. Step 4: challenge sorting

Challenge	sa	tr	i	w	#s	d	r
c_1	a_1	Weekday	50%	8	3	Medium	665
⋮							
c_2	a_3	Weekend	50%	9	1	Easy	720
c_3	a_3	Weekend	100%	9	1	Medium	805

5 Discussion

This article adapted Khoshkangini et al.'s approach for CLCS, adding the user's and community space-time behavior. Nevertheless, different and specific goal models could be proposed for other domains, and some devices in the recommendation process can be replaced by others.

An important aspect that requires greater detail is the granularity of the areas. This topic was not analyzed in this article, however their size can generate differences over the user's engagement. Particularly, regarding the required sampling quality, it could be fitted by using an area set representing a finer grain tessellation (smaller poligons). Also, due to the way they have been modeled here, the areas are independent and different, but it could be useful to model the equivalence of areas and to generate the challenges based on these equivalences. This would allow having a greater number of challenges in the analysis and help to minimize the cold start.

Another aspect that can be exploited is the calculation of atoms. The proposal presented here considers only the set of tuples with coordinates and timestamps, but could include a wider variables set, to relate these samples to the game challenge (or other game element) that had been assigned to the user.

Considering the sorting step, a different sorting strategy can be applied, and the playing history could be taken into account to consider the completion of similar challenges. Finally, the modeling of the user's motivation and objective can be used here.

Lastly, this approach was focused on the generation of challenges, but the question remains of how much of this scheme can be reused to generate other game elements? Or even more, can several types of elements be generated in parallel?

6 Conclusions and Future Work

In this article an automatic game challenge generation approach for CLCS was presented. The needs and characteristics of the CLCS are presented, such as the space-time objectives and the space-time user behavior, to later be valued during the process of automatic generation of game challenges. The contributions are a model of user profile considering the space-time behavior and challenge completion, a model for the different types of challenges applicable in CLCS, a model for the CLCS objectives and coverage, and a strategy for the application of Machine Learning techniques for adaptation.

It is still pending for future work to consider the level of completion of the game challenges at some point in the process. Also, a potential challenge generation strategy could consider area equivalence for both difficulty estimation, reward estimation, or challenge sorting. Also, the quality criterion could consider in some way this notion of equivalence of areas.

Other work scheduled for the future is to detect the player's objectives or to establish a relationship between the type of challenge and the types of player (from a space-time behavior point of view).

References

1. Dalponte Ayastuy, M., Torres, D.: Behavioral atoms for NY foursquare users in 2012 (April 2021). https://doi.org/10.5281/zenodo.4728128
2. Dalponte Ayastuy, M., Torres, D., Fernández, A.: Adaptive gamification in Collaborative systems, a systematic mapping study. Comput. Sci. Rev. **39**, 100333 (2021). https://doi.org/10.1016/j.cosrev.2020.100333. https://www.sciencedirect.com/science/article/pii/S1574013720304330
3. Böckle, M., Novak, J., Bick, M.: Towards Adaptive Gamification: a Synthesis of Current Developments. Research Papers (July 2017)
4. Cochero, J.: Appear: a citizen science mobile app to map the habitat quality of continental waterbodies. Ecología Austral. **28**(02), 467–479 (2018)
5. 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 2021. Association for Computing Machinery, New York (2021). https://doi.org/10.1145/3471391.3471431
6. Deterding, S., Dixon, D., Khaled, R., Nacke, L.: From game design elements to gamefulness: defining "gamification". In: Proceedings of the 15th International Academic MindTrek Conference: Envisioning Future Media Environments, MindTrek 2011, Tampere, Finland, pp. 9–15. ACM, New York (2011). https://doi.org/10.1145/2181037.2181040
7. Göbel, S., Wendel, V.: Personalization and adaptation. In: Dörner, R., Göbel, S., Effelsberg, W., Wiemeyer, J. (eds.) Serious Games, pp. 161–210. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-40612-1_7
8. Hamari, J., Koivisto, J., Sarsa, H.: Does gamification work? A literature review of empirical studies on gamification. In: 2014 47th Hawaii International Conference on System Sciences, pp. 3025–3034. IEEE (2014)
9. Iversen, S.: In the double grip of the game: challenge and *fallout* 3. Game Stud. **12** (2012). http://www.gamestudies.org/1202/articles/in_the_double_grip_of_the_game
10. Khoshkangini, R., Valetto, G., Marconi, A., Pistore, M.: Automatic generation and recommendation of personalized challenges for gamification. User Model. User Adap. Inter. **31**(1), 1–34 (2020). https://doi.org/10.1007/s11257-019-09255-2

11. Kishimoto, K., Kobori, H.: Covid-19 pandemic drives changes in participation in citizen science project "city nature challenge" in Tokyo. Biol. Conserv. **255**, 109001 (2021). https://doi.org/10.1016/j.biocon.2021.109001. https://www.sciencedirect.com/science/article/pii/S0006320721000537
12. Tomé Klock, A.C., da Cunha, L.F., de Carvalho, M.F., Eduardo Rosa, B., Jaqueline Anton, A., Gasparini, I.: Gamification in e-learning systems: a conceptual model to engage students and its application in an adaptive e-learning system. In: Zaphiris, P., Ioannou, A. (eds.) LCT 2015. LNCS, vol. 9192, pp. 595–607. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-20609-7_56
13. Lavoué, E., Monterrat, B., Desmarais, M., George, S.: Adaptive gamification for learning environments. IEEE Trans. Learn. Technol. **12**, 16–28 (2018). https://doi.org/10.1109/TLT.2018.2823710
14. Lemmens, R.: A conceptual model for participants and activities in citizen science projects. In: Vohland, K., et al. (eds.) The Science of Citizen Science, pp. 159–182. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-58278-4_9
15. Lopes, R., Bidarra, R.: Adaptivity challenges in games and simulations: a survey. IEEE Trans. Comput. Intell. AI Games **3**(2), 85–99 (2011). https://doi.org/10.1109/TCIAIG.2011.2152841
16. Nugent, J.: Citizen science: iNaturalist. Sci. Scope **041**(07), 12–13 (2018)
17. Orji, R., Tondello, G.F., Nacke, L.E.: Personalizing persuasive strategies in gameful systems to gamification user types. In: Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems, CHI 2018, pp. 1–14. Association for Computing Machinery, New York (2018). https://doi.org/10.1145/3173574.3174009
18. Sánchez-Anguix, V., Alberola, J.M., Julián, V.: Towards adaptive gamification in small online communities. In: Sanjurjo González, H., Pastor López, I., García Bringas, P., Quintián, H., Corchado, E. (eds.) SOCO 2021. AISC, vol. 1401, pp. 48–57. Springer, Cham (2022). https://doi.org/10.1007/978-3-030-87869-6_5
19. Thiebes, S., Lins, S., Basten, D.: Gamifying information systems - a synthesis of gamification mechanics and dynamics. In: ECIS 2014, June 2014 (2014)
20. Tondello, G., Wehbe, R., Diamond, L., Busch, M., Marczewski, A., Nacke, L.: The gamification user types hexad scale. In: CHI PLAY 2016, October 2016 (2016). https://doi.org/10.1145/2967934.2968082
21. Vahlo, J., Karhulahti, V.M.: Challenge types in gaming validation of video game challenge inventory (CHA). Int. J. Hum. Comput. Stud. **143**, 102473 (2020). https://doi.org/10.1016/j.ijhcs.2020.102473. https://www.sciencedirect.com/science/article/pii/S1071581920300756