



Article

Annotation-Assisted Clustering of Player Profiles in Cultural Games: A Case for Tensor Analytics in Julia

Georgios Drakopoulos * , Yorghos Voutos and Phivos Mylonas

Humanistic and Social Informatics Lab, Department of Informatics, Ionian University, 49100 Corfu, Greece; c16vout@ionio.gr (Y.V.); fmylonas@ionio.gr (P.M.)

* Correspondence: c16drak@ionio.gr

Received: 7 October 2020; Accepted: 7 December 2020; Published: 12 December 2020



Abstract: Computer games play an increasingly important role in cultural heritage preservation. They keep tradition alive in the digital domain, reflect public perception about historical events, and make history, and even legends, vivid, through means such as advanced storytelling and alternative timelines. In this context, understanding the respective underlying player base is a major success factor as different game elements elicit various emotional responses across players. To this end, player profiles are often built from a combination of low- and high-level attributes. The former pertain to ordinary activity, such as collecting points or badges, whereas the latter to the outcome of strategic decisions, such as participation in in-game events such as tournaments and auctions. When available, annotations about in-game items or player activity supplement these profiles. In this article, we describe how such annotations may be integrated into different player profile clustering schemes derived from a template Simon–Ando iterative process. As a concrete example, the proposed methodology was applied to a custom benchmark dataset comprising the player base of a cultural game. The findings are interpreted in the light of Bartle taxonomy, one of the most prominent player categorization. Moreover, the clustering quality is based on intra-cluster distance and cluster compactness. Based on these results, recommendations in an affective context for maximizing engagement are proposed for the particular game player base composition.

Keywords: gamification; cultural gaming; cultural heritage preservation; player annotations; data enrichment; Bartle taxonomy; Simon–Ando clustering; tensor algebra; multilinear distance; Julia

MSC: 68T05; 68Q32; 82C32; 91E40; 92B20

1. Introduction

Cultural heritage preservation has been a persistent topic in most societies. Computer games can contribute by offering unhindered access to cultural content in a fun and vivid way [1]. Games promoting cultural heritage preservation rely on affective learning, namely the human ability to understand and internalize complex concepts through intense emotions and joyful activities, to successfully introduce its player base to cultural content. The latter may well include famed monuments, works of art such as paintings and films, and even anecdotal stories and myths. Each of these cultural items, material or not, cause positive or even negative sentimental reactions. It has long been the objective of numerous private and public research initiatives to create such games [2]. The gaming industry has also been involved. One recent example is *Assassin's Creed Origins* which takes place in Egypt near the end of the Ptolemaic period (BC 49–44) and represents this era with great historical accuracy and making it accessible to a much greater audience.

There has been a trend towards the development of more complex, serious games, which are informed by both pedagogical and game-like, fun elements. Under this framework, the ANTIKLEIA

project introduces the implementation of a specific gamified module application [3] applied on real-life content mostly collected from the Europeana repository [4]. More specifically, the interactive platform of ANTIKLEIA employs a collection of cultural content files [5] in order to be further modified by individual users, as well as groups of users, through its software components, while making them directly available to the general public. Furthermore, metadata from existing files and related collections support a cultural gamified experience, allowing large collections to be restored and managed through coordinated individual or collective efforts. This kind of (semi-)automatic enrichment can be beneficial for activating recovery, even across many languages, and adding a conceptual framework to the resources that will be accessible through its platform.

In order to keep player interest unabated, cultural games often depend on eliciting affective responses from players on two distinct levels. Low-level activity relates to simple decisions such as using in-game items. In contrast, high-level activity relies heavily on the outcome of conscious strategic decisions ranging from behavior in in-game tournaments to how game connections to the real world are exploited. Since both attributes ultimately describe player behavior from different perspectives, it makes perfect sense to combine them in player profiles. Once such profiles are created, the game player base can be better understood under Bartle taxonomy or any other player classification for that matter. To achieve that, profiles have to be clustered, as ground truth regarding player types is typically unavailable. This is essentially the principal motivation behind this work.

The primary research objective of this article is twofold. First, a template Simon–Ando iterative scheme for clustering the player profiles of a cultural game based on the Bartle player taxonomy is developed. Second, the effect of including user annotations about in-game items or player activity to the above scheme is evaluated. As a secondary objective, practical recommendations for selecting game elements based on maximizing the affective potential are given. The above were implemented in Julia and differentiate this work from previous ones. Moreover, the core of the proposed methodology will be incorporated in the aforementioned ANTIKLEIA project framework.

The remainder of this work is structured as follows. The recent scientific literature is briefly reviewed in Section 2. In Section 3, the Bartle player taxonomy and the low- and high-level attributes are presented. The proposed tensor based methodology is the focus of Section 4. The experimental setup, the results, and their analysis are given in Section 5, while the recommendations coming from this analysis are discussed in Section 6. Section 7 concludes this article by recapitulating the main findings and delineating future research directions. Tensors are represented by capital calligraphic, matrices by boldface capital, and vectors by boldface small letters. Each technical abbreviation is defined the first time it is met in the text. Finally, the notation of this article is summarized in Table 1.

Table 1. Notation of this article.

Symbol	Meaning	Introduced in
\triangleq	Definition or equality by definition	Equation (1)
$\{s_1, \dots, s_n\}$	Set with elements s_1, \dots, s_n	Equation (1)
(t_1, \dots, t_n)	Tuple with elements t_1, \dots, t_n	Equation (16)
$ S $	Cardinality of set or tuple S	Equation (18)
\times_k	Tensor multiplication along dimension k	Equation (11)
\mathbf{I}_n	$n \times n$ identity matrix	Equation (8)
\mathbf{J}_n	$n \times n$ column reversal matrix	Equation (14)

2. Previous Work

Games have been proven to be excellent tools for recreation and learning [6,7]. Their design is based on properties such as immersion and engagement [8]. To this end, the elements of points, leaderboard, and badges (collectively known as PBL) take advantage of the player engagement loop [9,10]. Games designed for cultural heritage preservation are explored in [11,12]. The Bartle taxonomy is examined in [13]. The mechanisms behind leaderboard operation based on personality

traits are explored in [14], whereas personality patterns can be discovered through gaming [11,15]. Affective learning can be applied to gaming [12,16] in conjunction with big data [17] and machine learning (ML) techniques [18,19]. Clustering can be applied to emotional and physiological states [20]. Finally, if properly processed, voice can be a major indicator of human emotional state [21].

Tensor algebra extends linear algebra beyond two dimensions as explained in [22]. Tensor operations such as Tucker decomposition [23], Kruskal factorization [24], and higher order singular value decomposition (HOSVD) [25] naturally discover the multilinear interplay between a number of factors in the same way the singular value decomposition (SVD) can reveal linear dependencies between two vector spaces [26]. Tensor stack networks (TSNs) rely on neural network stacking in order to perform classification tasks [27], evaluate graph resiliency [28] and discover higher-order graph structures [29], and learn large vocabularies [30,31]. TSNs have also been applied to image compression as shown in [32] and in discovering geo-linguistic communities in Twitter [33]. TensorFlow is an open source low level framework for tensor operations including tensor eigenvectors and higher order SVD (HOSVD) [34,35]. Finally, in [36], a toolkit with extensive TSN functionality is described.

3. Players

3.1. Bartle Taxonomy

Bartle taxonomy describes four fundamental player types according to their objectives and how they accomplish them, the interactions with other players, and their relationship with the in-game world [13]. The four fundamental player types according to the Bartle taxonomy and potential factors behind the interest of each such type in cultural games are the following:

- **Explorers:** Every new game is a new world and they are determined to discover it. They enjoy visiting and recording every aspect of the game world, especially Easter eggs, loot boxes, cultural references, one-time items, and even game bugs. Since cultural games are frequently built around vast—and often past—worlds abounding with items, they are literally treasure houses for them.
- **Socializers:** Their gaming experience essentially comes down to intricate interaction with others and exploiting every game mechanism to achieve that. Cultural games offer an excellent chance for initiating conversations about a plethora of topics and for interaction through text chats, voice messages, and writing in in-game items such as chalk boards, portraits, mirrors, and books.
- **Achievers:** Working tirelessly towards accomplishing game objectives and ultimately achieving them, preferably first, is why they signed up. Appearing on leaderboards adds greatly to their gaming experience. Cultural games are ideal as there is a multitude of tournaments to participate in, myriad badges and one-time items to collect, and thousands of points to accumulate.
- **Killers:** As their name suggests, they seek to eliminate others, preferably accomplished players. They will relate if a game recreates military campaigns such as Cæsar’s Gallic Wars (BC 58–51) or it is alternative history themed such as an open-ended American Civil War (1861–1865).

The above qualitative taxonomy can be seen from another view. This is comprised of two axes, each with two points representing the two possible options in a fundamental decision, namely:

- **Action vs. Interaction:** The degree the actions of a player are one- or two-way.
- **Environment vs. Players:** The degree a player prefers the game environment or other players.

Figure 1 shows how each player category fits in this two dimensional (2D) space set forth by the above axes. This representation contains more semantic content in comparison to a single categorical scale. According to this systematic view, there exists the following two pairs of opposite categories:

- **Achievers and socializers:** Achievers are highly competitive players since they often race against both the clock and other achievers of comparable or even superior skills to fulfill a set of objectives. On the contrary, socializers are very cooperative and seek harmonic and mutually beneficial coexistence with other players usually in a more relaxed style.

- **Killers and explorers:** Explorers aim at learning whatever is possible to be known about the game and even some more. In that sense, they are the least invasive player category as they tend to observe and not act upon the game world. On the other hand, killers do change the game world, especially if they act en masse, in numerous ways.

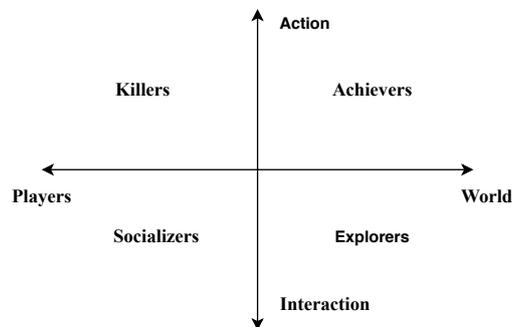


Figure 1. Bartle taxonomy in two dimensions (Source: Authors).

When designing game mechanics, the behavior of the player types and their expectations from the game should be taken into consideration in order for their interest to remain unabated. Player categorization is not static. Instead, changes occur for one or more of the following reasons:

- Seeking different goals after gaining enough experience from the cultural game.
- Joining a team with a culture of explicit or implicit peer pressure.
- Cooperating with or competing against influential players or teams.

Detecting player categorization changes can be done with an LMS-like algorithm monitoring tensor gradient [37]. Such a scheme tracks separately each factor of the higher-order dynamics of the player profile. This is impossible for a matrix method, as it mixes these factors in a gradient vector.

3.2. Player Types and Game Elements

The relationship between the player types and certain low- and high-level game elements is critical for the cultural game success. Both element categories, taken from the literature, are shown in Table 2 and explained here. They belong to one of the three primary axes of world immersion, reward system, and engagement loop, namely affective mechanisms closely related to learning [9].

A very common player engagement system applying to all player types is the triplet of points, badges, and leaderboards (PBL). This system can be broken down to its constituent parts as follows:

- Points can be found in the overwhelming majority of games. Players learn to devise strategies for point maximization. Such is the case of arcade games where points are collected in large numbers.
- Badges indicate special achievements which may well be unique for every player. They can be seen approximately as the digital counterparts of real world military medals or sports cups.
- Leaderboards are highly advertised player rankings. A very high score hints at an alternative ending path. Thus, players finishing with a lower score were indirectly invited to play again.

Loot boxes, namely stashes with special rewards, are very common in cultural gaming. Easter eggs, namely humorous references to other games or real events, are also cases of special interest. Their use is perhaps most highlighted in the film of the universe Ready Player One [38]. Moreover, one-time items, typically available during holidays or anniversaries, increase player interest and many have been reported to be sold in both in-game and out-of-game auctions [17]. Such rewards are valuable for explorers, achievers, and conditionally for killers if they contain special equipment [16]. Writable objects such as mirrors and books are new item types where players may write something visible to others, acting thus as a local in-game chat and as a focal point for explorers and socializers.

Table 2. Emotions triggered by low-(upper half) and high-level (lower half) elements (Source: See text).

Element	Explorers	Socializers	Achievers	Killers
Points	Neutral	Neutral	Joy	Neutral
Badges	Anticipation	Neutral	Anticipation	Joy
Leaderboards	Neutral	Anticipation	Joy	Joy
One-time items	Surprise	Neutral	Joy	Joy
Easter eggs	Joy	Neutral	Joy	Neutral
Loot boxes	Joy	Joy	Joy	Neutral
Secret rooms	Anticipation	Neutral	Joy	Neutral
Writable objects	Neutral	Anticipation	Neutral	Neutral
Cultural references	Joy	Anticipation	Neutral	Neutral
Interaction with players	Joy	Trust	Neutral	Anticipation
Interaction with NPCs	Neutral	Anticipation	Neutral	Anticipation
In-game tournaments	Surprise	Neutral	Surprise	Neutral
Cooperation in tournaments	Neutral	Joy	Neutral	Anger
Competition in tournaments	Neutral	Anger	Neutral	Joy
Alternative timelines	Anticipation	Joy	Anticipation	Anticipation
Crossovers	Joy	Neutral	Neutral	Neutral
Linear storytelling	Neutral	Neutral	Neutral	Joy
In media res storytelling	Joy	Neutral	Neutral	Anticipation
Open world	Joy	Joy	Anticipation	Joy
Open universe	Anticipation	Anticipation	Joy	Joy
Connection to physical world	Anticipation	Joy	Neutral	Neutral

In-game events such as auctions and tournaments are also an integral part of storytelling in cultural games attracting mainly explorers, achievers, and socializers [14]. In auctions literally everything can be sold. Prime examples include classical, renaissance, or Victorian monuments, inscriptions, paintings, statues, and decorated columns, as well as handcraft objects including jewellery, vessels, books, and ordinary household belongings. Tournaments may also be held on a regular or sporadic basis. There, killers have plenty of opportunities to compete with each other and socializers to cooperate [9]. Carefully designed tournaments may well also be special cases of an inducement prize contest, benefiting ultimately the player, typically an achiever or killer, giving the prize.

In an open world, player characters are free to roam in a huge digital world, which is particularly appealing to explorers and killers [9]. An expanding world is in fact a strong motive for players not only to keep playing, but also to return once they have completed the game. Depending on its theme, a cultural game may well have secret rooms, namely bonus areas filled with rewards.

An open universe allows the plot to be expanded in a number of worlds or for a character to be developed across various game installments. Perhaps the most well-known example is the Wing Commander space opera series [39], which is famed for its original and effective immersion mechanisms. It was possible to transfer the same player character along six game installments [40,41] allowing the racking of an impressive score and resulting in a more continuous and coherent story. Such a universe is appealing to all four fundamental player types for different reasons.

Despite coming from the digital realm, a cultural game may have extensions to the outside world. This can be carried out in many ways including scanning quick response (QR) codes, taking pictures, or recording street noise. Moreover, tangible rewards such as material badges are strong motives for players, especially for achievers and explorers. More recently augmented reality (AR), virtual reality (VR), and haptic systems bridge the gap between human senses and the game world, making games even more immersive. Explorers and socializers are typically fond of this type of features [6].

The role of artificial intelligence (AI) or non-playable characters (NPCs) is instrumental in most in-game worlds. Interacting with NPCs adds ways explorers can learn about the game and well designed NPCs may well capture the attention of socializers, especially if they have advanced AI. Moreover, they may provide critical hints to achievers and serve as cannon fodder for killers.

A significant design principle is that of the uncanny valley stating that NPCs should look convincingly human in order to be realistic but not *too* human as in this case they may look disturbing [42].

Storytelling techniques are essential to the game world evolution and all elements must have a place in that story, especially since it influences all four player types [9,14]. The classical Aristotelian structure is considered appropriate when simplicity is sought or when a clear message is to be given. On the contrary, the *in media res* storytelling of the Homeric sagas is considered suitable for games with open ended worlds or many installments with prequels and sequels. Other techniques such as the traditional Japanese storytelling [43] may be instrumental in games with experimental or specialized mechanics. Occasionally, alternate timelines or crossovers may make a linear story more interesting.

The affective reaction of the four player categories to these elements is important for cultural game mechanics if player interest is to be stimulated. Said reaction will be examined in terms of the emotion wheel model of Figure 2. The basic emotions under this model are anger, anticipation, joy, trust, fear, surprise, sadness, disgust, and the neutral emotional state.

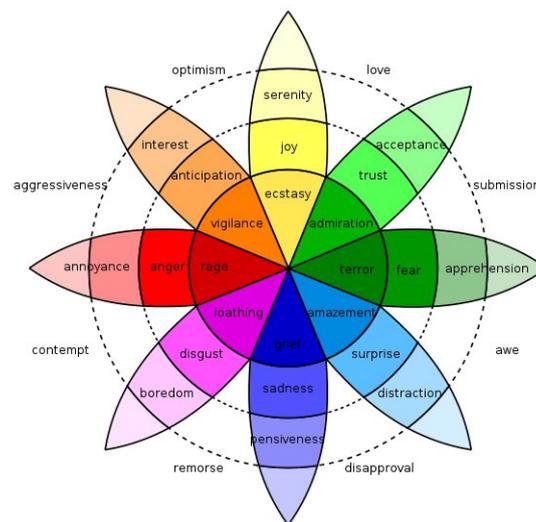


Figure 2. Emotion wheel (Source: Wikipedia).

Table 2 contains the emotions the most common gaming elements are most likely to trigger to the four player categories. Note that this table has been compiled under a statistical approach from reports and works (e.g., [14,16,44]). The upper half of the table has low level attributes, whereas the lower half has the high level ones. Observe the rich variability of emotional reactions.

3.3. Low Level Attributes

Low level attributes examined in this work pertain primarily to the way players interact with in-game items. Since they do not involve interaction with other players or with the broader in-game world, they are classified as low-level in the sense that no strategic thinking has to take place prior to or during interaction. Nonetheless, these kind of features are indicative of player psychology as they relate to unconscious and almost instinctive decisions or decisions with minor cognitive effort [6]. These features have been selected on literature recommendations [6,15,45] and are shown in Table 3.

Table 3. Elements and mnemonic names for the low-level player profile (Source: [6,15,45]).

Element	Mnemonic	Quantile Meaning	Element	Mnemonic	Quantile Meaning
$p_L^{(i)}[1]$	points	Points accumulated	$p_L^{(i)}[5]$	eggs	Easter eggs found
$p_L^{(i)}[2]$	badges	Badges collected	$p_L^{(i)}[6]$	loot	Loot boxes found
$p_L^{(i)}[3]$	board	Leaderboard position	$p_L^{(i)}[7]$	rooms	Secret rooms found
$p_L^{(i)}[4]$	items	One-time items collected	$p_L^{(i)}[Q_0]$	writeable	Writable objects used

The low-level player profile for the i th player is the numerical vector $\mathbf{p}_L^{(i)}$ of length $Q_0 = 8$ with the structure of Equation (1). The latter also contains symbolic names for clarity and readability in Algorithms 1 and 2. The proposed methodology can be extended to any number of attributes.

$$\mathbf{p}_L^{(i)} \triangleq [p_L^{(i)}[1], \dots, p_L^{(i)}[Q_0]]^T \in V_0^{Q_0 \times 1} \tag{1}$$

In Equation (1), the value set V_0 is $\{1/5, 2/5, 3/5, 4/5, 1\}$. Because of its structure, the symbolic five star scale of Table 4 is used throughout the text to make it more legible and intuitive-friendly. To enhance legibility, symbolic names consist of only one word without any modifiers.

Table 4. Symbolic names for the numerical attribute scale (Source: Authors).

Numerical	1/5	2/5	3/5	4/5	1
Symbolic	Weak	Low	Medium	High	Strong

These symbolic values are also linearly ordered not only because they represent actual numerical values where comparisons make perfect sense, but also because of their very context they represent distinct ranking levels. Thus, for instance, Weak < High or Medium > Low are valid comparisons.

The values of V_0 have been chosen in order to address the following reasons:

- The large variance of players and items involved may not always lead to numerical scales.
- Ranking ensures that attribute values remain at the same scale.
- The ranking indicates how the community is going as opposed to individual player performance.
- Ranking can be applied at various stages of the game or even for different player subsets.

Notice that the number of attributes Q_0 in this work is the number of low-level elements found in the recent scientific literature. This number by no means limits the generality of the proposed methodology. In fact, the only actual hard constraint regarding how many attributes may be used is that the low- and high-level player profiles have the same number of components.

The distance between two low level profiles is defined as in Equation (2):

$$l(\mathbf{p}_L^{(i)}, \mathbf{p}_L^{(j)}) \triangleq l(i, j) \triangleq \exp\left(-\frac{\|\mathbf{p}_L^{(i)} - \mathbf{p}_L^{(j)}\|_2^2}{2Q_0}\right) \tag{2}$$

3.4. High Level Attributes

High level game attributes pertain to player actions with higher semantic content as well as with conscious decisions. They mainly involve interaction with other players, strategic decisions, and dealing with the in-game world. As with the previous case, the high level profile of the i th player is a numerical vector $\mathbf{p}_H^{(i)}$ with the structure of Equation (3) and with the same length as $\mathbf{p}_L^{(i)}$:

$$\mathbf{p}_H^{(i)} \triangleq [p_H^{(i)}[1], \dots, p_H^{(i)}[Q_0]]^T \in V_0^{Q_0 \times 1} \tag{3}$$

The components of each profile vector $\mathbf{p}_H^{(i)}$ along with the respective mnemonic names, added for enhanced readability of Algorithm 2, are shown in Table 5. Observe that the high level attributes can add more semantic context since they directly reflect decisions which require at least some cognitive effort and they can be described with annotations or a restricted vocabulary as explained in [46]. Moreover, they are indicative of the social intelligence of the players, especially in the way they choose to compete against or cooperate with each other [47]. The same quantile-based scheme of the previous case and the same five star scale are used to obtain the respective values for each player for consistency.

The similarity metric between two high level profiles is the Gaussian kernel of Equation (4):

$$h(\mathbf{p}_H^{(i)}, \mathbf{p}_H^{(j)}) \triangleq h(i, j) \triangleq \exp\left(-\frac{\|\mathbf{p}_H^{(i)} - \mathbf{p}_H^{(j)}\|_2^2}{2Q_0}\right) \quad (4)$$

Table 5. Elements and mnemonic names for the high-level player profile (Source: [7,8,14]).

Element	Mnemonic	Quantile Meaning	Element	Mnemonic	Quantile Meaning
$p_H^{(i)}[1]$	players	Interaction with players	$p_H^{(i)}[5]$	competition	Tournament competition
$p_H^{(i)}[2]$	npcs	Interaction with NPCs	$p_H^{(i)}[6]$	timelines	Participation to timelines
$p_H^{(i)}[3]$	tours	Tournament participation	$p_H^{(i)}[7]$	crossovers	Participation to crossovers
$p_H^{(i)}[4]$	cooperation	Tournament cooperation	$p_H^{(i)}[Q_0]$	world	Fraction of world explored

3.5. First- and Higher-Order Player Profiles

From the attributes described earlier it is possible to classify players according to the Bartle taxonomy. This is a first order player classification as it is based solely on features of a single player. Although this approach certainly has merit as is based on ground truth directly deriving from player activity, higher-order methodologies tend to systematically yield more robust player classifications as they aggregate not only local ground truth states but also information implicitly encoded in similarity matrices constructed from pairwise profile distance metrics such as those of Equations (1) and (3). Among the reasons favoring higher-order approaches are the following [14,15]:

- They rely on significantly more information regarding player profiles. As such, they can mine latent patterns, encoded for instance as profile similarities, player interactions, or player clusters.
- Higher-order methods tend to ignore the effect of outliers, typically expressed as unusual or missing attribute values. Properly designed methods can provide estimates for erroneous values.
- Most games create their own environment through rules or storytelling and players adapt to it. Thus, higher-order methodologies tend to reveal player types relative to the that environment.
- Gaming is mainly a social activity. Thus, although players have their own style, they may copy individual elements. For large player bases this leads to cluster formation over time.

The mapping of profiles to the Bartle taxonomy types will take two different forms depending whether only low-level attributes are available. This is necessitated by the form of some of the higher-order methods. When only the low-level attributes are available, Algorithm 1 applies.

Notice that in both Algorithms 1 and 2 the inequality symbols \geq and \leq can be applied to the symbolic values of Table 4 reserved for the profile attributes. In these cases, they are to be interpreted taking into consideration the linear order of the symbolic values. Therefore, for instance, the inequality points \geq High is true when points has the values Strong or High and it is false otherwise.

Algorithm 1 relies on a number of observations about player activity as reflected in the low-level attributes. Achievers care for points and badges, which eventually will bring them to a prominent position in the leaderboard [14]. Explorers seek to find loot boxes, one-time boxes, secret rooms, or Easter eggs with the latter three being more important. This implies that explorers may well

accumulate significant points in the process, so points alone cannot distinguish between these player types [14,44]. Socializers can be recognized by their extended use of writeable objects in order to communicate with other players as well as their drive to find secret rooms with the hope of finding more players there. These factors may result in an increased number of badges related to player activity [15]. Killers are less easy to define through a set of rigid rules since their objective may be simple, but in order to achieve it they frequently resort in activities which are also common in the other player types. For instance, they may seek secret rooms as socializers and explorers do, they may collect badges similar to achievers and socializers, they want one-time items as explorers do, and they share with achievers the excitement for points as well as for a prominent leaderboard position [6].

Algorithm 1 Mapping of low-level attributes to Bartle taxonomy.

Require: Low-level profile as described in Equation (1).

Ensure: A first-order mapping to one of the four player types of the Bartle taxonomy.

```

1: if [points  $\geq$  High] and [at least one of badges, points  $\geq$  High] then
2:   type = achiever
3: else if [loot  $\geq$  High] and [at least two of rooms, loot, eggs  $\geq$  High] then
4:   type = explorer
5: else if [writeable  $\geq$  High] and [at least one of badges, rooms  $\geq$  Medium] then
6:   type = socializer
7: else
8:   type = killer
9: end if
10: return type

```

When both low- and high-level attributes are available in player profiles, then the more complex set of rules outlined in Algorithm 2 is used instead for the mapping to the Bartle taxonomy.

Algorithm 2 Mapping of high-level attributes to Bartle taxonomy.

Require: High-level profiles as in Equation (3).

Ensure: A first-order mapping to one of the four player types of the Bartle taxonomy.

```

1: if [tours  $\geq$  High] and [competition  $\leq$  High and cooperation  $\geq$  Weak] then
2:   type = achiever
3: else if [at least one of timelines, crossovers, world  $\geq$  High] and [tours  $\geq$  Weak or npcs  $\geq$  Medium]
   then
4:   type = explorer
5: else if [at least one of players, npcs  $\geq$  High] and [cooperation  $\geq$  Medium] then
6:   type = socializer
7: else
8:   type = killer
9: end if
10: return type

```

Algorithm 2 connects high-level attributes to player types. Achievers are more likely to participate to in-game tournaments in order to accomplish game-wide objectives, possibly with the help of other players [8]. Explorers tend to investigate both the in-game world and its extensions such as crossovers and alternative timelines. In addition, they may occasionally take part in tournaments in order to find one-time boxes, loot items, or other prizes [14]. Socializers interact with other players or NPCs, and they may tend to help others in in-game events [7]. As in the previous case, killers are difficult to

discern from other players as killers may join in-game events like achievers. Some killers search the game world as explorers. In addition, killers and socializers interact with other players and NPCs.

Higher-order methods may well use first-order profile classifications as starting points since the latter are obtained from ground truth data. Depending on the data representations, options include vector clustering, string matching, decision trees, or graph matching. Since here profiles are represented as numerical vectors, clustering techniques are more appropriate. In particular, iterative schemes coming from a template Simon–Ando scheme were selected and tested, as explained below.

4. Proposed Clustering Methodology

4.1. Template Simon–Ando Clustering

The algorithmic cornerstone for clustering player profiles is the Simon–Ando iterative scheme, which is based on the power method. The latter estimates the primary eigenvector \mathbf{g} of a matrix $\mathbf{M} \in \mathbb{R}^{n \times n}$ through a matrix-vector computation and normalization cycle, as shown in Algorithm 3.

Algorithm 3 The power method.

Require: Matrix \mathbf{M} and numerical tolerance ζ_0

Ensure: The primary eigenvector \mathbf{g} of \mathbf{M} is computed

- 1: initialize vector $\mathbf{g}^{[0]}$; normalize $\mathbf{g}^{[0]} \leftarrow \mathbf{g}^{[0]} / \|\mathbf{g}^{[0]}\|_1$
 - 2: **repeat**
 - 3: compute $\mathbf{g}^{[k+1]} \leftarrow \mathbf{M}\mathbf{g}^{[k]}$; normalize $\mathbf{g}^{[k+1]} \leftarrow \mathbf{g}^{[k+1]} / \|\mathbf{g}^{[k+1]}\|_1$
 - 4: **until** $\|\mathbf{g}^{[k+1]} - \mathbf{g}^{[k]}\|_1 \leq \zeta_0$
 - 5: **return** \mathbf{g}
-

The main parameter of the power iteration is the initialization of vector $\mathbf{g}^{[0]}$. In the general case, initialization takes place with random elements or with information extracted from matrix \mathbf{M} itself. In the former case, it is advisable that Algorithm 3 be executed many times as a random vector many not contain the direction of \mathbf{g} , especially when the dimension of the column space of \mathbf{M} is large.

The Simon–Ando iteration is similar to the power method with one major difference: it terminates when the elements of $\mathbf{g}^{[k]}$ are clustered. To this end, a different termination criterion τ_0 is necessary. Analysis of the power method indicates that it undergoes the stages described below [45,48]:

- For few iterations, the elements of $\mathbf{g}^{[k]}$ remain close to the random starting points of $\mathbf{g}^{[0]}$.
- As the iteration progresses, the elements of $\mathbf{g}^{[k]}$ move from their original positions.
- For a narrow window of iterations, the Simon–Ando phase, the elements of $\mathbf{g}^{[k]}$ are clustered.
- After the Simon–Ando phase, the elements of $\mathbf{g}^{[k]}$ are driven away from this clustering.
- The power method converges to \mathbf{g} and terminates.

Algorithm 4 is the template Simon–Ando scheme from which the three iterative schemes are derived. It is a matrix free algorithm and multiplications can be seen as passing the vector $\mathbf{v}^{[k]}$ to a kernel $T[\cdot]$ and retrieving the result $\mathbf{v}^{[k+1]}$. In this work, selecting the particular form of $T[\cdot]$ uniquely determines the iterative scheme. Additionally in the template $\mathbf{v}^{[k+1]}$ is normalized by a generic norm $\|\cdot\|_H$ in a Hilbert space, but the specific norm depends on the selection of $T[\cdot]$ as well.

Algorithm 4 The template Simon–Ando scheme.

Require: Functions of Equations (2) and (4).

Ensure: Players are clustered based on the basic Bartle types

- 1: initialize vector $\mathbf{g}^{[0]}$; normalize $\mathbf{g}^{[0]}$
 - 2: initialize matrices \mathbf{L} and \mathbf{H} as in Equation (9)
 - 3: **if** annotations are used **then**
 - 4: scale matrix \mathbf{H} once as $\mathbf{H} \leftarrow \mathbf{H}\mathbf{N}$
 - 5: **end if**
 - 6: compute kernel $T[\cdot]$ as a function of \mathbf{L} and \mathbf{H}
 - 7: **repeat**
 - 8: compute $\mathbf{g}^{k+1} \leftarrow T[\mathbf{g}^k]$; normalize $\mathbf{g}^{k+1} \leftarrow \mathbf{g}^{k+1} / \|\mathbf{g}^{k+1}\|_H$
 - 9: **until** termination criterion τ_0 is **true**
 - 10: **return**
-

The termination criterion τ_0 is a key component of Algorithm 4 since it determines clustering quality. The critical requirement for τ_0 is to detect the Simon–Ando phase before it is over. One way to achieve this is to compute the elementwise harmonic mean of the second order difference between three successive versions $\mathbf{g}^{[k-1]}$, $\mathbf{g}^{[k]}$, and $\mathbf{g}^{[k+1]}$ as shown in Equation (5), assuming their length is n :

$$\tau_0[k] \triangleq \frac{n}{\sum_{j=1}^n \frac{1}{|\mathbf{g}^{[k+1]}[j] - 2\mathbf{g}^{[k]}[j] + \mathbf{g}^{[k-1]}[j]|}} \geq \eta_0 \quad (5)$$

The harmonic mean has been selected in this work for the following reasons:

- **Robustness:** The harmonic mean in sharp contrast to the geometric mean is immune to erroneous values close to zero or to outliers in general.
- **Reliability:** For stochastic input, even for a relatively few observations and for a wide array of distributions, the harmonic mean converges to its true value.

To understand why the eigenstructure of \mathbf{M} appears in the Simon–Ando clustering, consider the following: Let $\mathbf{g}[i]$ be a fixed element of \mathbf{g} . For the very small number of iterations of the Simon–Ando phase, the multiplication with \mathbf{M} should at most perturb the clustering of the elements of \mathbf{g} . By construction, the elements of the former are weighted linear combinations of the elements of the latter, as shown in Equation (6). It follows then that for each of the n elements of $\mathbf{g}^{[k]}$ it holds that:

$$\mathbf{g}^{[k+1]}[i] = \sum_{j=1}^n \mathbf{M}[i, j] \mathbf{g}^{[k]}[j], \quad 1 \leq i \leq n \quad (6)$$

If some $\mathbf{g}^{[k]}[j]$ does not appear in (6), namely it is zero, then it can be added into both sides. Then, by stacking the n equations and casting them in an equivalent matrix notation leads to Equation (7):

$$\begin{bmatrix} \mathbf{g}^{[k+1]}[1] \\ \mathbf{g}^{[k+1]}[2] \\ \vdots \\ \mathbf{g}^{[k+1]}[n] \end{bmatrix} = \begin{bmatrix} \mathbf{M}[1, 1] & \mathbf{M}[1, 2] & \dots & \mathbf{M}[1, n] \\ \mathbf{M}[2, 1] & \mathbf{M}[2, 2] & \dots & \mathbf{M}[2, n] \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{M}[n, 1] & \mathbf{M}[n, 2] & \dots & \mathbf{M}[n, n] \end{bmatrix} \begin{bmatrix} \mathbf{g}^{[k]}[1] \\ \mathbf{g}^{[k]}[2] \\ \vdots \\ \mathbf{g}^{[k]}[n] \end{bmatrix} \quad (7)$$

Under the assumption that for two successive steps in the clustering phase of the Simon–Ando $\mathbf{g}^{[k+1]} \approx \mathbf{g}^{[k]}$ Equation (7) can be cast as a linear equation or as an eigenvalue problem as in (8):

$$\mathbf{g}^{[k]} = \mathbf{A}\mathbf{g}^{[k]} \Leftrightarrow (\mathbf{A} - \mathbf{I}_n)\mathbf{g}^{[k]} = \mathbf{0} \quad (8)$$

Matrices **L** and **H** in (9) contain the low- and high-level attribute distance metrics, respectively, as:

$$\mathbf{L} \triangleq \begin{bmatrix} 1 & l(1,2) & \dots & l(1,P_0) \\ l(2,1) & 1 & \dots & l(2,P_0) \\ \vdots & \vdots & \ddots & \vdots \\ l(P_0,1) & l(P_0,2) & \dots & 1 \end{bmatrix} \quad \mathbf{H} \triangleq \begin{bmatrix} 1 & h(1,2) & \dots & h(1,P_0) \\ h(2,1) & 1 & \dots & h(2,P_0) \\ \vdots & \vdots & \ddots & \vdots \\ h(P_0,1) & h(P_0,2) & \dots & 1 \end{bmatrix} \quad (9)$$

In Equation (9), P_0 is the total number of players. Because of the form of the Gaussian kernel both distance matrices **L** and **H** are symmetric and their diagonal elements equal one.

The user annotation weight matrix **N** is defined elementwise as in Equation (10). It is based on the annotations of Table 6. As a sign of the pairwise joint player activity between players i and j , the harmonic mean of the ratio of the annotation references to either player for the three annotation categories of Table 6 to the respective maximum is computed. This choice yields a real symmetric **N**.

$$\mathbf{N}[i,j] \triangleq \min \left[\mu_0, \frac{3}{\frac{\sum_{k=1}^{P_0} (f_k^1 + f_k^2)}{(f_i^1 + f_j^1) + (f_i^2 + f_j^2)} + \frac{\sum_{k=1}^{P_0} f_k^3}{f_i^3 + f_j^3} + \frac{\sum_{k=1}^{P_0} (f_k^4 + f_k^5)}{(f_i^4 + f_j^4) + (f_i^5 + f_j^5)}} \right] \quad 1 \leq i, j \leq P_0 \quad (10)$$

Table 6. Annotations description (Source: See text).

Annotation	Symbol	Category	Meaning
Player	a_1	Interaction	A player has interacted with another (not in tournament)
NPC	a_2	Interaction	A player has interacted with an NPC (not in tournament)
Timelines	a_3	Game world	A player participates to an in-game alternative timeline
Cooperation	a_4	Tournament activity	A player has cooperated with another player
Competition	a_5	Tournament activity	A player has competed against another player

In the above equation, f_i^k is the frequency of annotation a_k regarding player i in the dataset and μ_0 is a small positive constant. The rationale behind the selection of the particular weight of Equation (10) is that joint player activity should be high when they are of the same type since they have similar objectives and in-game behavior. Additionally, this choice allows the separate treatment of different annotation categories. Notice that annotations a^4 and a^5 refer to opposing behaviors, namely cooperation and competition in tournaments. In this case, for most players, either one of the frequencies f_i^4 and f_i^5 will be high but not both. Of course both can be down as well. There are many ways of selecting the constant μ_0 . Since it represents the minimum amount of player interaction in the game, here it was set to be the minimum non-zero value of Equation (10).

The scheme named *matrix* is derived from the Simon–Ando template by selecting the kernel $T[\cdot]$ to be matrix **L** and the scheme named *comb* is the half-sum of matrices **L** and **H**. Therefore, the former iteration relies only on low-level player attributes, whereas the latter exploits both low- and high-level ones. If the matrix **N** is used, then it multiplies **H** from right once before iteration starts. This is called scheme *comb-a* (see Table 7 for an overview of the clustering methods).

Table 7. Overview of iterative clustering schemes (Source: Authors).

Method	Kernel	Explicit	Attributes	Annotations
<i>matrix</i>	Matrix	Yes	Low	No
<i>comb</i>	Matrix	Yes	Low+High	No
<i>comb-a</i>	Matrix	Yes	Low+High	Yes
<i>tensor</i>	Tensor	No	Low+High	No
<i>tensor-a</i>	Tensor	No	Low+High	Yes

4.2. Tensor-Based Clustering

Tensors allow similarity metrics modeling simultaneous linear dependencies between sets of variables. Although from a programming perspective a tensor is a multidimensional array indexed by an array of p integers, where p is the tensor order, the formal tensor definition is the following:

Definition 1. A p th order tensor \mathcal{T} , where $p \in \mathbb{Z}^*$, is a linear mapping coupling p non necessarily distinct vector spaces $\mathbb{S}_k, 1 \leq k \leq p$. If $\mathbb{S}_k = \mathbb{R}^{I_k}$, then $\mathcal{T} \in \mathbb{R}^{I_1 \times \dots \times I_p}$.

Tensor multiplication along the k th dimension $\mathcal{G} = \mathcal{X} \times_k \mathcal{Y}$ between tensors \mathcal{X} of order p and \mathcal{Y} of order q can occur if both tensors have the same number of entries I_k in the k th dimension.

Definition 2 (Tensor multiplication). The multiplication of \mathcal{X} and \mathcal{Y} along the k th dimension is defined as:

$$(\mathcal{X} \times_k \mathcal{Y})[i_1, \dots, i_p, j_1, \dots, j_{k-1}, j_{k+1}, \dots, j_q] \triangleq \sum_{i_k=1}^{I_k} \mathcal{X}[i_1, \dots, i_k, \dots, i_p] \mathcal{Y}[j_1, \dots, i_k, \dots, j_q] \quad (11)$$

Observe that, as a special case, a tensor-vector product $\mathcal{X} \times_k \mathbf{v}$ along the k th dimension where $\mathcal{X} \in \mathbb{R}^{I_1 \times \dots \times I_p}$ is a p th order tensor and $\mathbf{v} \in \mathbb{R}^{I_k}$ is a vector is defined elementwise as:

$$(\mathcal{X} \times_k \mathbf{v})[i_1, \dots, i_{k-1}, i_{k+1}, \dots, i_p] \triangleq \sum_{i_k=1}^{I_k} \mathcal{X}[i_1, \dots, i_k, \dots, i_p] \mathbf{v}[i_k] \quad (12)$$

The result is a tensor of order $p - 1$. Therefore, for a third order tensor the result is a matrix.

Definition 3 (Frobenius norm). The Frobenius norm of a p th order tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times \dots \times I_p}$ is:

$$\|\mathcal{X}\|_F \triangleq \left(\sum_{i_1=1}^{I_1} \dots \sum_{i_p=1}^{I_p} \mathcal{X}[i_1, \dots, i_p]^2 \right)^{\frac{1}{2}} \quad (13)$$

Given the above, the iterations named *tensor* and *tensor-a* in the experiments are built around the third-order tensor $\mathcal{T} \in \mathbb{P}_\varphi \times \mathbb{P}_\varphi \times \mathbb{L}_\varphi$, where L_0 is the available level of attributes, namely two in this case. If more levels were available, then the proposed approach could be extended to use them by adding one layer per level. This demonstrates the generality as well as the simplicity of this particular method. Notice that each level of \mathcal{T} is a proper matrix by itself. The geometric insight for it can be seen as stacking matrices \mathbf{L} and \mathbf{H} along the third dimension, as shown in Figure 3.

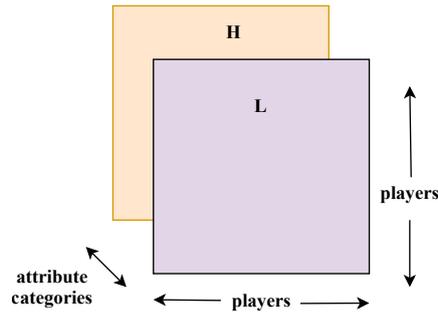


Figure 3. Tensor structure (Source: Authors).

The computation part of the iteration loop has the block form of Equation (14). When the annotation weight matrix \mathbf{N} is used, \mathbf{H} is multiplied from the right by it once during initialization.

$$\mathbf{H} = \mathbf{HN}, \quad \mathbf{G}^{[0]} \triangleq \begin{bmatrix} \mathbf{g}_L^{[0]} & \mathbf{g}_H^{[0]} \end{bmatrix}, \quad \mathbf{Y}^{[k+1]} \triangleq \mathcal{T} \times_1 \mathbf{G}^{[k]}, \quad \mathbf{G}^{[k+1]} \triangleq \mathbf{J}_2 \mathbf{Y}^{[k+1]} \quad (14)$$

The above iteration is based on running the low- and high-level cluster schemes separately and then combining them. An optional weight deriving from annotations for the high-level attributes can be added. The iteration steps of the computation part of the loop of Algorithm 4 are as follows:

- The two cluster schemes for $\mathbf{g}_L^{[0]}$ and $\mathbf{g}_H^{[0]}$ are initialized with random entries.
- If user annotations are used, then \mathbf{H} is multiplied by \mathbf{N} once during initialization.
- Tensor \mathcal{T} multiplies matrix $\mathbf{G}^{[k]}$, essentially filtering $\mathbf{g}_L^{[k]}$ and $\mathbf{g}_H^{[k]}$ through \mathbf{L} and \mathbf{H} , respectively.
- At the next step, the two columns of matrix $\mathbf{Y}^{[k+1]}$ are swapped through matrix \mathbf{J}_2 .

In this case, the kernel $T[\cdot]$ of Algorithm 4 is an implicit function of \mathbf{L} , \mathbf{H} , and \mathbf{J}_2 . Moreover, at the end of the computation part of the loop, normalization is done with the Frobenius norm, which is valid since matrices are tensors of order two. This happens as now iterations are about matrices.

Table 7 offers an overview of the iterative clustering schemes deriving from the template of Algorithm 4. It is based on applicable standard criteria for classifying iterative schemes [37].

5. Results

5.1. Setup

Table 8 shows the experimental setup for evaluating the proposed methodology. It contains those parameters indirectly influencing clustering, but they have to be manually inserted by the developer.

Table 8. Experimental setup (Source: Authors).

Parameter	Value	Parameter	Value
Number of attributes Q_0	8	Attribute groups P_0	2 (low and high)
Maximum vector size	1 MB	Processor	Intel Core i5 3210M @ 1.6 GHz
Maximum matrix size	8 MB	Main memory size	16 GB
Maximum tensor size	16 MB	L1/L2/L3 cache size	1/3/4 MB
Raw dataset rows L_0	175 K	Processed dataset rows L'_0	64 K
Number of annotators N_0	8617	Annotation options E_0	11
Number of players P_0	1024	Termination threshold η_0	0.2
Number of runs R_0	1000	Maximum number of iterations η_1	1024

The experiments are designed to answer the following questions:

- **First vs. higher order:** These tests examine the clustering quality achieved by the first-order mappings of Algorithms 1 and 2 compared to that of the higher-order iterative clustering methods. Recall that the latter aggregate local ground truth to reveal global properties.

- **Tensor vs. matrix:** From the iteration template of Algorithm 4, matrix- and tensor-based clustering schemes are derived as outlined in Table 7. The objective is to determine whether switching to the latter can offer an advantage concerning player clustering ignoring annotations.
- **Effect of annotations:** These experiments aim at evaluating the effect of user annotations. To this end, the methods relying on user annotations to obtain weights for the high-level attributes are compared against their counterparts without these weights.

5.2. Dataset Synopsis

The dataset serving as benchmark to test the proposed methodology as well as the effect of user annotations was obtained from Kaggle. It comprises of L_0 rows pertaining to an anonymized fantasy-style cultural game involving a number of late Roman and medieval historical elements including gladiatorial style combats, castle building, and knight quests, with jousting tournaments being one of the main in-game events. These rows had the following fields:

- **Annotator:** The user creating the annotation. This field is independent of the player(s) involved in the activity. As such, it was ignored as it did not contribute information to the experiments.
- **Timestamp:** Automatically generated date and time of the annotation in ISO 8601 format. This field was also ignored as it was not relevant to the analysis conducted here.
- **Category:** It is one of the possible categories: Interaction of a player with a player or with an NPC, presence of a player in the alternative timelines, and interaction with players in a tournament.
- **Subject:** The player who does or initiates an action, as described in the respective field. To keep annotations simple, even in multiplayer events only elementary actions were recorded.
- **Object:** The player or NPC who is the target of the action, where applicable. It has the same format with the previous field but its semantics are much different from an analysis perspective.
- **Action:** Although there are many actions, here stand out the cooperation and the competition in tournaments and the participation to timelines. Everything else is listed as generic interaction.

The above annotations come from N_0 distinct players, they refer to the in-game events, and they were drawn from a restricted list with a total of E_0 options. The latter means that annotating users had to choose only from a pre-specified set of annotations intended for the game designers to understand how certain game elements were understood by the player base. Therefore, there were no missing or erroneous values in the various fields of the dataset and the semantics were well defined. Observe that the number of annotators is significantly larger than the number of players the annotations are about. A possible explanation is that annotators chose to provide information about players with considerable in-game activity. Additionally, the relatively small number and format of options E_0 in the annotations list made them easy to remember, therefore making easy their creation. In turn, this is an important incentive for creating a large number of them over a short amount of time.

Table 6 has the meaning of each user annotation as well as the category it belongs to. It resulted from processing the raw dataset and extracting player as well as action information. The latter was the grouping factor for the processed L'_0 rows in order to form the annotations categories a_1 to a_5 .

Table 9 contains the distribution of each of the five categories in the resulting dataset. Observe that this distribution is rather balanced, therefore greatly facilitating analysis. Notice there is only one category probability, namely that of generic interaction, is much larger than the other ones.

Table 9. Category distribution in the processed dataset (Source: Authors)

Player	NPC	Timelines	Cooperation	Competition
24.25%	19.5%	18.25%	20%	18%

5.3. Number of Iterations and Floating Point Operations

The primary figure of merit for each iterative algorithm is the number of iterations. Table 10 presents for the methodologies of Table 7 the number of iterations required to achieve the same level

of convergence η_0 . Since the starting point is random, the number of iterations is a stochastic quantity. Therefore, for each scheme was run R_0 and the mean and variance were computed. Moreover, as an additional safeguard, a maximum of η_1 iterations was included.

Table 10. Iterations for each method (Source: Authors).

	Matrix	Comb	Comb-a	Tensor	Tensor-a
mean	262.6178	234.1333	224.3334	205.9500	193.0333
var	44.1445	42.1133	42.4467	41.3345	40.9987

The termination criterion τ_0 used to achieve the number of iterations of the above figure was that of Equation (5) with a parameter of η_0 selected, as shown in Table 8. This value is low enough to achieve convergence without allowing the power method proceeding too far. From the results in Table 10, the following can be inferred about the number of iterations for each of the clustering schemes:

- The matrix based methods systematically yield a higher number of iterations. Thus, their convergence to the same level is slower compared to the tensor based ones.
- The annotation based methods achieve systematically lower number of iterations from their counterparts. This is a clear indication that they contribute to the overall clustering quality.
- The combination of tensor representation with the user annotations resulted in the lowest number of iterations. Therefore, they contribute individually to mining knowledge from the dataset.

Table 11 has the average number of floating point operations (flops) for each method. Again, since the clustering schemes are stochastic, the procedure of the previous case was applied.

Table 11. Flops for each method (Source: Authors).

	Matrix	Comb	Comb-a	Tensor	Tensor-a
mean	414.867	409.556	407.332	397.403	394.309
var	37.497	38.683	40.803	40.6711	38.445

From the results in Table 11, the following conclusions can be drawn regarding the scaling:

- The tensor based methods achieve a somewhat lower number of flops although they contain more expensive operations of linear algebra. This can be attributed to the lower number of iterations.
- The cost for incorporating annotations is negligible, as can be seen from the entries for flops. Thus, it pays off to have them in the scheme as part of the mining strategy.

5.4. Cluster Distance

As ground truth is unavailable for the processed dataset, the following quality metrics were used. They rely solely on the results of the experiments and do not require tuning or hyperparameters.

- The average inter-cluster \bar{d} distance of Equation (16)
- The maximum inter-cluster d^* distance of Equation (17)
- The average intra-cluster distance \bar{D} of Equation (19)

Table 12 shows the normalized values of \bar{d} , d^* , and \bar{D} for each clustering scheme. The reason for this is that a relative indicator of clustering quality sheds more light to the performance of each algorithm. Each clustering scheme was normalized to its respective minimum.

Table 12. Clustering metrics for each scheme (Source: Authors).

	Matrix	Comb	Comb-a	Tensor	Tensor-a
\bar{d}	1	1.2276	1.9937	2.2816	3.0145
d^*	1	1.3333	2.4423	2.8497	3.9973
\bar{D}	2.8936	2.3444	2.0025	1.1215	1

Let C_i and C_j be the i th and j th cluster and C_0 the total number of available clusters. Then,

$$d_{i,j} \triangleq \frac{1}{|C_i||C_j|} \sum_{\mathbf{w} \in C_i} \sum_{\mathbf{w}' \in C_j} d(\mathbf{w}, \mathbf{w}') \quad (15)$$

In (15), $|C_i|$ and $|C_j|$ are the number of data points of clusters C_i and C_j . Then,

$$\bar{d} \triangleq \frac{1}{\binom{C_0}{2}} \sum_{(i,j)} d_{i,j} = \frac{2}{C_0(C_0 - 1)} \sum_{(i,j)} d_{i,j} \quad (16)$$

Along a similar line of reasoning, d^* is the maximum over all pairwise distances $d_{i,j}$:

$$d^* \triangleq \max [d_{i,j}] \quad (17)$$

Another figure of merit is how compact clusters are. This is measured by the average distance \bar{D} between any two points in a cluster. The intra-cluster distance D_i for C_i is defined as in Equation (18):

$$D_i \triangleq \frac{1}{|C_i|(|C_i| - 1)} \sum_{(\mathbf{w}, \mathbf{w}')} d(\mathbf{w}, \mathbf{w}') \quad (18)$$

Metric \bar{D} is obtained by averaging the intra-cluster distances as in Equation (19):

$$\bar{D} \triangleq \frac{1}{C_0} \sum_{i=1}^{C_0} D_i \quad (19)$$

From the entries of Table 12, the following observations can be made:

- The tensor based methods consistently result in better separated clusters both in the average distance case and in the maximum distance as indicated respectively by the high values of \bar{d} and d^* . This means that bounds between clusters contains fewer data points and are more clear.
- The tensor based methods yield more compact clusters compared to the matrix based ones, as indicated by the low values of \bar{D} . This complements the findings for \bar{d} and d^* , as more distant clusters “push” the same number of data points to smaller regions, resulting in higher density.
- Both the above hold even more when the annotation-based methods are compared against their counterparts. This is a clear indication that the inclusion of user annotations improves the overall clustering process. Combined with the tensor representation, this is even more enhanced.

5.5. Player Type Distribution

Once profile clustering is complete, two player type distributions are of interest:

- **First-order distributions:** It is the player type distribution as obtained by Algorithms 1 and 2. Therefore, the mapping from the player profile to the Bartle taxonomy is based on the former.
- **Higher-order distributions:** Once a clustering scheme is complete, clusters are formed, profiles are mapped as before, and then each player receives the majority type of the respective cluster.

From the entries of Table 13, the following can be said:

- All methods yield approximately the same percentage for achievers. This also almost holds for socializers. This is an indication these player categories may have distinct behavior which can be almost directly translated to both low- and high-level attributes.
- The first-order distributions and the *matrix* method give a very high number for killers which is inconsistent with the very low number obtained from the remaining methods. A possible explanation is they tend to treat as killers every profile not fulfilling the criteria for the other types.

- The tensor-based methods yield almost identical distributions. This may be an indication that despite the different starting points they both eventually converge to the same distribution exploiting higher order patterns with annotations accelerating this convergence.
- The above is also true but to a lesser extent for the results from the *comb* and the *comb-a* iterative schemes. This can be attributed to the fact that annotations help the clustering process but do not suffice by themselves to uncover all the patterns in the processed dataset.

Table 13. Player type distributions (Source: Authors).

	Algo. 1	Algo. 2	Matrix	Comb	Comb-a	Tensor	Tensor-a
Achievers	33.6666%	32.7500%	25.7500%	36.7500%	33.3334%	36.2534%	36.5000%
Explorers	11.5000%	32.5000%	30.3334%	37.1250%	34.6666%	32.7066%	32.5000%
Socializers	18.5000%	8.5000%	21.2500%	23.2500%	25.2500%	26.0400%	24.3366%
Killers	26.3334%	26.2500%	22.6666%	2.875%	6.7500%	5.0000%	6.6634%

5.6. Discussion

The results obtained earlier agree with these reported elsewhere in the recent scientific bibliography. In particular, the combination of low- and high-level player attributes to improve player experience has been proposed [19]. Moreover, in [14], it is maintained that players can easier fit the Bartle taxonomy when their collective behavior is taken into consideration. The need for an advanced player clustering scheme is highlighted in [17]. Tensor clustering fulfills these requirements.

Based on the experiment results the inclusion of user annotations to the clustering scheme seem to make a difference both in clustering quality and in scalability. In addition, it should be highlighted that the tensor representation for player profiles and the user annotations are two different factors contributing in their own way to clustering. The results presented here are consistent with the findings of Yang *et al.* [49] where high quality labeling rules for reducing annotation cost were derived by crowdsourcing. In [50], these rules are augmented with ones mined with ML. Large-scale data annotation is indispensable for other type of games including serious games [51]. Annotations have also been proposed as a supplementary mechanism for understanding player actions in affective games [52]. To this end, games intended for cultural preservation may well benefit from including a dedicated module designed for collecting and processing them such as the one described in [53]. Since human activity patterns are to be extracted from gaming activity, it makes perfect sense that mining algorithms may rely on human assistance, even a partial one like annotations.

Regarding the role of user annotations in general, if they are utilized properly, they can be instrumental in the disambiguation of player activity [52]. As such, they can significantly boost the performance of mining algorithms by providing the initial information and ontological structure they can start from. The primary reasons for this happening are listed below [17]:

- Humans can almost immediately understand both the complex semantics inherent in player actions and the possible ultimate objectives of other players, especially in the context of the game.
- Moreover, dedicated or even casual players can draw on considerable information from their respective gaming experience in order to interpret the actions of other players.

As a specific example for the above, we plan to incorporate the strategy proposed in this work within the gamification module of the ANTIKLEIA project in order to exploit the user annotations. Through a suite of open markup technologies, such as the extensible markup language (XML), the resource description framework (RDF), and JavaScript Object Notation (JSON), as well as open protocols such as *Dublin core*, annotations can be driven as ground truth data from the dedicated user interface (UI) component to the data management one. Meta-data have been known to boost the performance of mining algorithms in terms of accuracy and robustness [54].

In this framework, Figure 4 depicts how annotations fit in the software architecture of ANTIKLEIA. Once the raw annotations are extracted, they are first locally stored as JSON documents. The underlying

storage could very well be a document database, such as MongoDB, but a relational database is suitable as well. Then, the following analysis takes place in parallel:

- Annotations, as explained above, are driven to the data mining algorithms for interpreting the actions of the players. In turn, this leads to a better understanding of the player base.
- Annotations are also analyzed in terms of the players who gave them. This gives insight into the segment of players who are willing to improve the game, participating thus in a deeper level.
- The value of the available cultural objects for the player base is estimated through analytics based on the annotations. This is an important cultural preservation function.

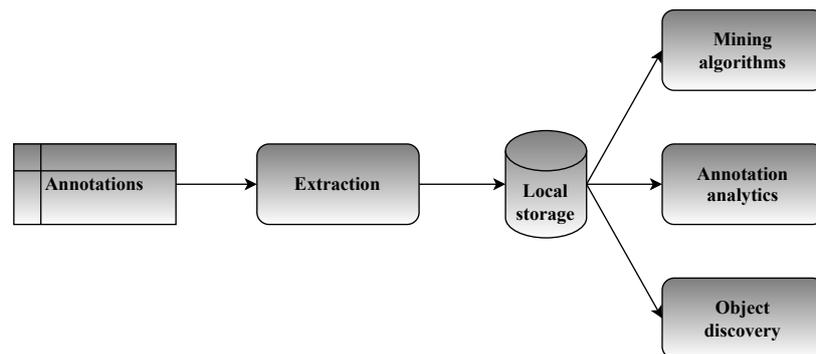


Figure 4. Annotations and the architecture of project ANTIKLEIA (Source: Authors).

Besides game designers, the above analytics may be of use to cultural enthusiasts, cultural professionals, and independent developers alike, as further described in the ANTIKLEIA use cases [3]. User annotations can even contribute to the data-driven construction of massive ontologies for cultural items [55]. The latter can reveal how the player base sees these items based on diverse criteria that clearly go beyond the scope of the present work.

6. Recommendations

With knowledge of the particular player base composition certain recommendations can be stated for game designers and practitioners with the explicit aim of keeping player interest unabated from an affective perspective. It is a long running tenet of the gaming industry that a successful game should be played more than once [11]. Several criteria for this purpose have been proposed in the literature with engagement, replayability, and immersion being among them [14,20]. The strategy in this work will be to maximize engagement, which will be achieved based on the following data:

- Table 2 will serve as a guide for the relationship between game elements and the basic emotions they elicit across player types. However, this is only a statistical approach.
- The last column of Table 13, namely the results of clustering with the *tensor-a* methodology, will be considered as the true player base distribution.

The analysis will be based on two complementary factors. First, emphasis will be placed on those gaming elements which attract the majority of player types. Conversely, elements eliciting negative emotions to the majority of players will be in general ignored. To this end, the following analysis will be used: Each basic emotion of Figure 2 is assigned the value ± 1 depending on its polarity, whereas the neutral state is assigned the value zero. For each element, the score of Equation (20) is computed:

$$\sigma_i \triangleq \sum_t e_{i,t} p_t \quad (20)$$

In Equation (20), $e_{i,t}$ is the statistical emotional response of players of category t to the i th game element, while t ranges over the four player types. Given the above, Table 14 is generated for the

elements used in the low- and high-level player profiles in Tables 3 and 5, respectively. This analysis can be conducted for the remaining elements of Table 2 or any other game elements for that matter.

Table 14. Scores for the game elements (Source: Authors).

Element	Score	Element	Score	Element	Score	Element	Score
Points	0.3650	Eggs	0.6900	Players	0.6350	Competition	−0.1766
Badges	0.7567	Boxes	0.9333	NPCs	0.3100	Alternative	1
Leaderboards	0.6750	Rooms	0.6900	Tournaments	0.6900	Crossovers	0.3250
One-time	0.7567	Writable	0.24333	Cooperation	0.1766	World	1

Given the values of Table 14, for the specific player base, priority should be given to widening the in-game world, creating alternative timelines, and including loot boxes. In addition, tournaments should be cooperative instead of competitive. On the contrary, writeable objects and the inclusion of more NPCs will offer little to the in-game experience as they attract only a small number of players.

7. Conclusions and Future Work

Games in the digital era promote cultural heritage by relying on several affective capabilities. This article has a twofold focus. First, a template Simon–Ando iterative scheme is developed for clustering player profiles consisting of attributes with affective content. Certain schemes deriving from this template are evaluated based on inter- and intra-clustering distances as well as on the number of iterations and the number of floating point operations required. The former are indicators of clustering quality, while the latter of scalability. The results are interpreted in light of the Bartle taxonomy. Understanding the player base in this way can lead to recommendations about placing emphasis on specific game elements. Second, the role of user annotations on clustering is evaluated by deriving versions using weight matrices based on them. Experiment results clearly show that in every case the inclusion of user annotations had a distinct positive effect.

Concerning future research directions, the algorithmic aspect of the proposed methodology can be improved by examining alternative combinations of attributes and tensors utilizing them. Moreover, the stability properties as well as its complexity should be carefully evaluated through extensive simulations. Another possible research direction lies in constructing more datasets from diverse player audience. This will lead to better evaluation of alternative player clustering methodologies.

Author Contributions: G.D. did the coding and developed the algorithmic strategy; Y.V. contributed to the algorithmic approach and did the bibliographic search; and P.M. provided guidance and oversight. All authors have read and agreed to the published version of the manuscript.

Funding: This research was co-financed by the European Union and Greek national funds through the Competitiveness, Entrepreneurship and Innovation Operational Programme, under the Call “Research-Create-Innovate”; project title: “Development of technologies and methods for cultural inventory data interoperability—ANTIKLEIA”; project code: T1EDK-01728; MIS code: 5030954.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Squire, K. Cultural framing of computer/video games. *Game Stud.* **2002**, *2*, 1–13.
2. Mortara, M.; Catalano, C.E.; Bellotti, F.; Fiucci, G.; Houry-Panchetti, M.; Petridis, P. Learning cultural heritage by serious games. *J. Cult. Herit.* **2014**, *15*, 318–325. [[CrossRef](#)]
3. ANTIKLEIA Greek Web Site. Available online: <https://www.antikleia.gr> (accessed on 20 November 2020).
4. Europeana Web Site. Available online: <https://www.europeana.eu/en> (accessed on 20 November 2020).
5. ANTIKLEIA Web Site. Available online: <http://antikleia.website/> (accessed on 20 November 2020).
6. Lumsden, J.; Edwards, E.A.; Lawrence, N.S.; Coyle, D.; Munafò, M.R. Gamification of cognitive assessment and cognitive training: A systematic review of applications and efficacy. *JMIR Ser. Games* **2016**, *4*, e11. [[CrossRef](#)] [[PubMed](#)]

7. Lister, M. Gamification: The effect on student motivation and performance at the post-secondary level. *Issues Trends Educ. Technol.* **2015**, *3*, 112–120. [[CrossRef](#)]
8. Streukens, S.; van Riel, A.; Novikova, D.; Leroi-Werelds, S. Boosting customer engagement through gamification: A customer engagement marketing approach. In *Handbook of Research on Customer Engagement*; Edward Elgar Publishing: London, UK, 2019.
9. Landers, R.N.; Bauer, K.N.; Callan, R.C. Gamification of task performance with leaderboards: A goal setting experiment. *Comput. Hum. Behav.* **2017**, *71*, 508–515. [[CrossRef](#)]
10. Huang, B.; Hew, K.F. Do points, badges and leaderboard increase learning and activity: A quasi-experiment on the effects of gamification. In Proceedings of the 23rd International Conference on Computers in Education, Ishikawa, Japan, 30 November–4 December 2015; pp. 275–280.
11. Seaborn, K.; Fels, D.I. Gamification in theory and action: A survey. *Int. J. Hum.-Comput. Stud.* **2015**, *74*, 14–31. [[CrossRef](#)]
12. Triantoro, T.; Gopal, R.; Benbunan-Fich, R.; Lang, G. Would you like to play? A comparison of a gamified survey with a traditional online survey method. *Int. J. Inf. Manag.* **2019**, *49*, 242–252. [[CrossRef](#)]
13. Menéndez, H.D.; Vindel, R.; Camacho, D. Combining time series and clustering to extract gamer profile evolution. In *International Conference on Computational Collective Intelligence*; Springer: Heidelberg, Germany, 2014; pp. 262–271.
14. Jia, Y.; Liu, Y.; Yu, X.; Voids, S. Designing leaderboards for gamification: Perceived differences based on user ranking, application domain, and personality traits. In Proceedings of the Conference on Human Factors in Computing Systems, Denver, CO, USA, 6–11 May 2017; pp. 1949–1960.
15. Codish, D.; Ravid, G. Detecting playfulness in educational gamification through behavior patterns. *IBM J. Res. Dev.* **2015**, *59*, 1–6. [[CrossRef](#)]
16. Conejo, F. Loyalty 3.0: How to revolutionize customer and employee engagement with big data and gamification. *J. Consum. Mark.* **2014**, *31*, 86–87.
17. Kim, B. The Popularity of Gamification in the Mobile and Social Era. *Libr. Technol. Rep.* **2015**, *51*, 5–9.
18. Kyriazidou, I.; Drakopoulos, G.; Kanavos, A.; Makris, C.; Mylonas, P. Towards Predicting Mentions to Verified Twitter Accounts: Building Prediction Models over MongoDB with keras. In *WEBIST*; ScitePress: Setubal, Portugal, 2019; pp. 25–33.
19. Tekofsky, S.; Van Den Herik, J.; Spronck, P.; Plaats, A. PsyOps: Personality assessment through gaming behavior. In Proceedings of the International Conference on the Foundations of Digital Games, SASDG, Crete, Greece, 14–17 May 2013.
20. Yang, W.; Rifqi, M.; Marsala, C.; Pinna, A. Physiological-based emotion detection and recognition in a video game context. In Proceedings of the IJCNN, Rio de Janeiro, Brazil, 8–13 July 2018; pp. 1–8.
21. Drakopoulos, G.; Pikrammenos, G.; Spyrou, E.D.; Perantonis, S.J. Emotion Recognition from Speech: A Survey. In Proceedings of the 2014 International Conference on Circuits, Systems, Communication and Information Technology Applications (CSCITA), Mumbai, India, 4–5 April 2019; pp. 432–439.
22. De Lathauwer, L. Signal Processing Based on Multilinear Algebra. Ph.D. Thesis, KU Leuven, Leuven, Belgium, 1997.
23. De Lathauwer, L.; De Moor, B. From matrix to tensor: Multilinear algebra and signal processing. In *Institute of Mathematics and Its Applications Conference Series*; Oxford University Press: Oxford, MA, USA, 1998; Volume 67, pp. 1–16.
24. Papalexakis, E.E.; Pelechrinis, K.; Faloutsos, C. Spotting misbehaviors in location-based social networks using tensors. In Proceedings of the WWW 14: 23rd International World Wide Web Conference, Seoul, Korea, 7–11 April 2014; pp. 551–552. [[CrossRef](#)]
25. Papalexakis, E.E.; Faloutsos, C. Fast efficient and scalable core consistency diagnostic for the PARAFAC decomposition for big sparse tensors. In Proceedings of the ICASSP 2015, Brisbane, Australia, 19–24 April 2015; pp. 5441–5445.
26. Alexopoulos, A.; Drakopoulos, G.; Kanavos, A.; Mylonas, P.; Vonitsanos, G. Two-Step Classification with SVD Preprocessing of Distributed Massive Datasets in Apache Spark. *Algorithms* **2020**, *13*, 71. [[CrossRef](#)]
27. Bao, Y.T.; Chien, J.T. Tensor classification network. In *MLSP*; IEEE: Piscataway, NJ, USA, 2015; pp. 1–6.
28. Drakopoulos, G.; Mylonas, P. Evaluating graph resilience with tensor stack networks: A keras implementation. *Neural Comput. Appl.* **2020**, *32*, 4161–4176. [[CrossRef](#)]

29. Benson, A.R.; Gleich, D.F.; Leskovec, J. Tensor spectral clustering for partitioning higher-order network structures. In *ICDM*; SIAM: New Delhi, India, 2015; pp. 118–126.
30. Yu, D.; Deng, L.; Seide, F. The deep tensor neural network with applications to large vocabulary speech recognition. *IEEE Trans. Audio Speech Lang. Process.* **2012**, *21*, 388–396. [[CrossRef](#)]
31. Yu, D.; Deng, L.; Seide, F. Large vocabulary speech recognition using deep tensor neural networks. In Proceedings of the INTERSPEECH 2012 ISCA's 13th Annual Conference, Portland, OR, USA, 9–13 September 2012.
32. Ma, J.; Liu, X.Y.; Shou, Z.; Yuan, X. Deep tensor admm-net for snapshot compressive imaging. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea, 27 October–2 November 2019; pp. 10223–10232.
33. Drakopoulos, G.; Stathopoulou, F.; Kanavos, A.; Paraskevas, M.; Tzimas, G.; Mylonas, P.; Iliadis, L. A genetic algorithm for spatio-social tensor clustering: Exploiting TensorFlow potential. *Evol. Syst.* **2019**. [[CrossRef](#)]
34. Abadi, M.; Barham, P.; Chen, J.; Chen, Z.; Davis, A.; Dean, J.; Devin, M.; Ghemawat, S.; Irving, G.; Isard, M.; et al. TensorFlow: A system for large-scale machine learning. In *OSDI'16*; USENIX Association: Berkeley, CA, USA, 2016; pp. 265–283.
35. Abadi, M. TensorFlow: Learning functions at scale. In *SIGPLAN International Conference on Functional Programming*; ACM: New York, NY, USA, 2016; pp. 1–8.
36. Palzer, D.; Hutchinson, B. The tensor deep stacking network toolkit. In *IJCNN*; IEEE: Piscataway, NJ, USA, 2015; pp. 1–5.
37. Drakopoulos, G.; Mylonas, P.; Sioutas, S. A Case of Adaptive Nonlinear System Identification with Third Order Tensors in TensorFlow. In *INISTA*; IEEE: Piscataway, NJ, USA, 2019; pp. 1–6. [[CrossRef](#)]
38. imdb. Ready Player One. Available online: [imdb.com/title/tt16777720](https://www.imdb.com/title/tt16777720) (accessed on 28 September 2020).
39. Star Citizen. Wing Commander I 25th Anniversary Gameplay Video. Available online: <https://www.youtube.com/watch?v=ADrl5uWckJw> (accessed on 28 September 2020).
40. imdb. Wing Commander. Available online: [imdb.com/title/tt0245563](https://www.imdb.com/title/tt0245563) (accessed on 28 September 2020).
41. WC CIC. Wing Commander. Available online: www.wcnews.com (accessed on 28 September 2020).
42. Mori, M.; MacDorman, K.F.; Kageki, N. The uncanny valley [from the field]. *IEEE Robot. Autom. Mag.* **2012**, *19*, 98–100. [[CrossRef](#)]
43. Bryan, J.I. Japanese Story-Telling. *Lotus Mag.* **1914**, *5*, 407–412.
44. Chou, Y.K. *Actionable Gamification: Beyond Points, Badges, and Leaderboards*; Packt Publishing Ltd.: Birmingham, UK, 2019.
45. Levin, S.A. The architecture of robustness. In *Global Challenges, Governance, and Complexity*; Edward Elgar Publishing: Cheltenham, UK, 2019.
46. Vallet, D.; Fernandez, M.; Castells, P.; Mylonas, P.; Avrithis, Y. A contextual personalization approach based on ontological knowledge. *Contexts Ontol. Theory Pract. Appl.* **2006**, *2006*, 35.
47. Diplaris, S.; Sonnenbichler, A.; Kaczanowski, T.; Mylonas, P.; Scherp, A.; Janik, M.; Papadopoulos, S.; Ovelgoenne, M.; Kompatsiaris, Y. Emerging, collective intelligence for personal, organisational and social use. In *Next Generation Data Technologies for Collective Computational Intelligence*; Springer: Berlin, Germany, 2011; pp. 527–573.
48. Hartfiel, D. Proof of the Simon-Ando theorem. *Proc. Am. Math. Soc.* **1996**, *124*, 67–74. [[CrossRef](#)]
49. Yang, J.; Fan, J.; Wei, Z.; Li, G.; Liu, T.; Du, X. Cost-effective data annotation using game-based crowdsourcing. *PVLDB* **2018**, *12*, 57–70. [[CrossRef](#)]
50. Yang, J.; Fan, J.; Wei, Z.; Li, G.; Liu, T.; Du, X. A game-based framework for crowdsourced data labeling. *VLDB J.* **2020**, *29*, 1311–1336. [[CrossRef](#)]
51. Seneviratne, L.; Izquierdo, E. An interactive framework for image annotation through gaming. In Proceedings of the International Conference on Multimedia Information Retrieval, Philadelphia, PA, USA, 29–31 March 2010; pp. 517–526.
52. Yannakakis, G.N.; Paiva, A. Emotion in games. In *Handbook on Affective Computing*; Springer: Berlin/Heidelberg, Germany, 2014; pp. 459–471.
53. Liu, T.; Yang, J.; Fan, J.; Wei, Z.; Li, G.; Du, X. CrowdGame: A Game-Based Crowdsourcing System for Cost-Effective Data Labeling. In Proceedings of the International Conference on Management of Data, Amsterdam, The Netherlands, 30 June–5 July 2019; pp. 1957–1960.

54. Voutos, Y.; Drakopoulos, G.; Mylonas, P. Metadata-enriched Discovery of Aspect Similarities Between Cultural Objects. In *SMAP*; IEEE: Piscataway, NJ, USA, 2020. [[CrossRef](#)]
55. Drakopoulos, G.; Voutos, Y.; Mylonas, P. Recent Advances On Ontology Similarity Metrics: A Survey. In *SEEDA-CECNSM*; IEEE: Piscataway, NJ, USA, 2020. [[CrossRef](#)]

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).