GameSpace: An Explorable Visualization of the Videogame Medium

6 authors, including:

James Ryan
University of California, Santa Cruz

36 PUBLICATIONS   **116** CITATIONS

Some of the authors of this publication are also working on these related projects:

Project   Situated and Pragmatic Story Generation View project

GameSpace: An Explorable Visualization of the Videogame Medium

JAMES RYAN, Expressive Intelligence Studio, University of California, Santa Cruz

ERIC KALTMAN, Expressive Intelligence Studio, University of California, Santa Cruz

ANDREW MAX FISHER, Games and Playable Media, University of California, Santa Cruz

TAYLOR OWEN-MILNER, 2K Games, Novato, California

MICHAEL MATEAS, Expressive Intelligence Studio, University of California, Santa Cruz

NOAH WARDRIP-FRUIN, Expressive Intelligence Studio, University of California, Santa Cruz

We present GameSpace, a visualization of the videogame medium as a 3D explorable space. GameSpace takes the form of a galaxy of 15,000 stars, where each star represents an existing game and stars are positioned such that related games are nearer to one another. Users may fly around the space, using conventional game controls, to explore the medium and serendipitously discover new games. This application is made possible by a combination of techniques from natural language processing and machine learning, namely latent semantic analysis and multidimensional scaling, and builds upon our earlier work in producing tools for videogame discovery. Beyond demonstrating an increase in user engagement relative to our primary existing tool, in this paper we outline our design goals, describe our development process, and discuss the project along various dimensions of intrigue. GameSpace is available online at http://gamespace.io.

1 INTRODUCTION

The Internet is replete with text about videogames. From in-depth reviews, to walkthroughs and FAQS, to general descriptions like those found on Wikipedia and MobyGames, community members and professionals constitute a perpetual wellspring of corpora that are ripe for larger, structural analysis techniques. What if we could harness the information embedded in that text for game studies analysis? What conclusions could we reach and what ways could we find to illustrate new conclusions about the game community and about the videogame medium itself?

As we outline in Section 3, there is an emerging area that *does* harness videogame text corpora, through techniques from natural language processing (NLP) and other areas. In [45], we reviewed this body of work and characterized its approach to game studies as a bottom-up one that can generate novel insights that cannot be obtained through more conventional top-down methods. Additionally, however, we noted a critical deficit in the existing work taking this new approach: none of the models reported in that literature can be

engaged beyond the publications describing them, which is troublesome given the complexity of machine learning models and the resulting difficulty of adequately describing them. As such, we called for future work in this area to make its models more understandable through visualization and interactivity.

In the spirit of this call, we have released a suite of tools and visualizations that make the models that we have built by processing videogame text corpora interactive.[1] The primary applications in this suite are GameNet, an explorable network of nearly 12,000 games that are linked to one another according to how related they are, and GameSage, an interface to GameNet that generates a new node in the network by processing a user's idea for a game.

Elsewhere, we have demonstrated a number of strengths present in our tool suite: GameNet's relatedness judgments are sound [45], GameNet and GameSage expose users to relevant games that are unfamiliar to them [40, 45], GameNet is a useful tool for game scholars [45], and GameSage is a useful one for game designers [40]. We have also learned, however, of opportunities to raise the level of engagement in these tools. In a user study evaluating GameNet and GameSage, for instance, 51% of respondents indicated that they would not be likely to use the tool again [40].

In the interest of maintaining these benefits of our model while providing an interface to it that is more engaging for users, we present GameSpace, an explorable visualization of the videogame medium built using NLP and machine learning. GameSpace takes the form of a 3D galaxy of stars, where each star represents an existing game and stars are placed in the space such that related games are nearer to one another. Users can fly around the space, using conventional game controls, to explore and learn about the medium. Each game has data attached to it—its Wikipedia article and a YouTube video demonstrating gameplay—that may be engaged without leaving the space. This application is made possible through a combination of techniques from NLP and machine learning, which we applied to a corpus of over 15,000 videogame articles extracted from Wikipedia. Further, GameSpace is notable as a *living*, publicly (if indirectly) modifiable visualization—we have set up a system that will automatically rederive GameSpace's underlying model periodically, as new descriptive videogame content is added to Wikipedia, to add new games into the space. This means that users, by describing new games on Wikipedia, may cause the set of games included in GameSpace to expand accordingly. As we articulate again in the next section, our primary goal in developing GameSpace has been to make a GameNet-like tool that is more engaging to users. In Section 6.2, we test whether this goal has been achieved by comparing user metrics automatically collected by both systems.

Generally, we hope that this project will encourage new explorations of the potential of using machine learning and interactive visualization to engender exploration and understanding of the videogame medium.

2 DESIGN GOALS

Our development of GameSpace was structured according to a number of design goals:

(1) Produce a more engaging interface to our underlying NLP model (*i.e.*, one that is more engaging than GameNet); this is our *core design goal* and the subject of our evaluation in Section 6.2.
(2) Maintain the spontaneous and serendipitous sense of discovery available in our earlier tools.
(3) Present a fun and soothing experience that is visually and aurally appealing.

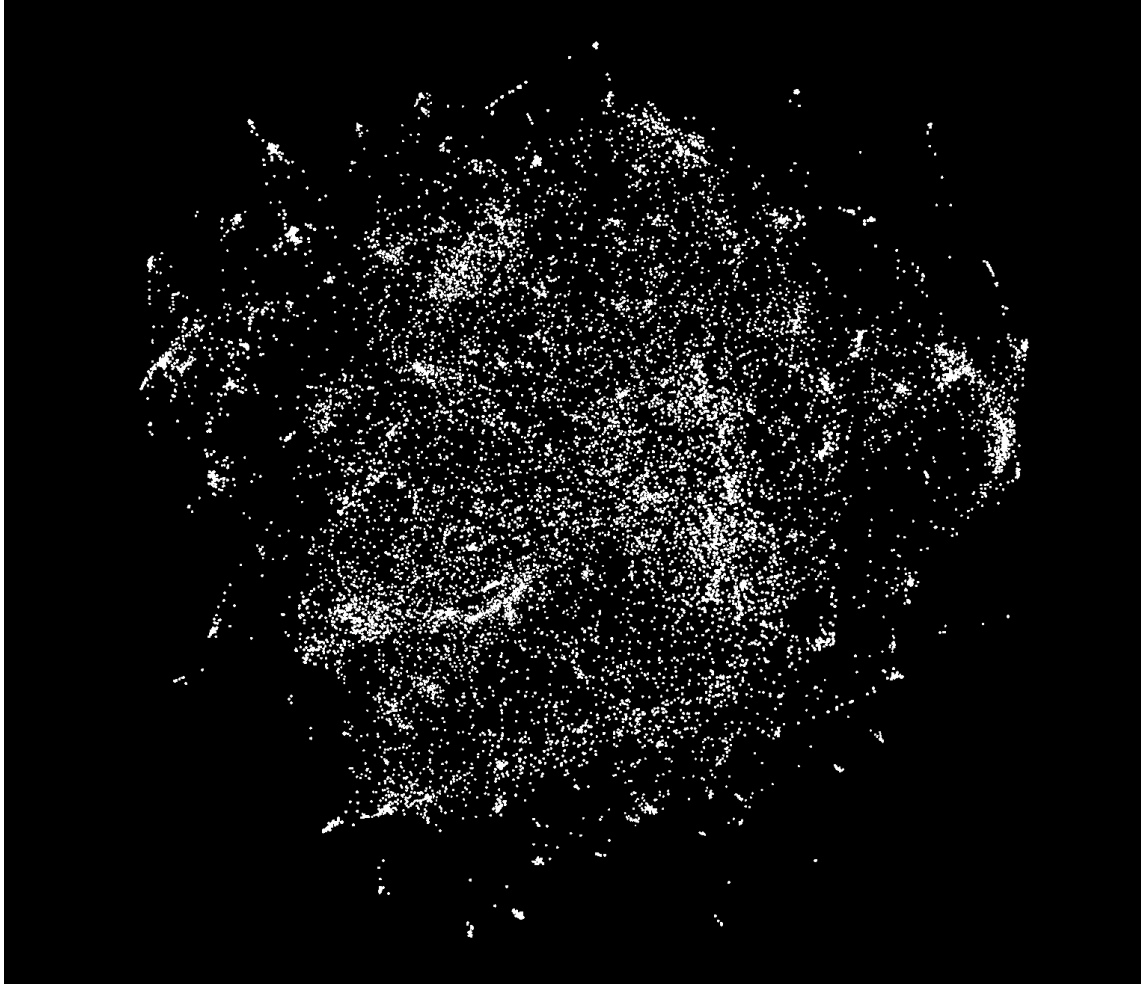[1]These tools are hosted online at http://gamecip-projects.soe.ucsc.edu.

Fig. 1. A wide view of the entire 3D space. Each star represents an existing videogame, and stars are positioned such that related games are nearby one another. As a result, noticeable constellations of related games appear.

(4) Directly integrate outside data sources, such as Wiki- pedia articles and YouTube videos, into the tool (whereas GameNet forces users to leave the system to engage these sources); this promotes a more contained and cohesive experience.

(5) Encourage users to share and discuss games that they discover in the space.

(6) Engage the Wikipedia community to provide a mechanism for mutual improvement of both our model and coverage of videogames on Wikipedia.

Throughout the paper, we cite these goals when discussing specific design decisions that we made in pursuit of them.
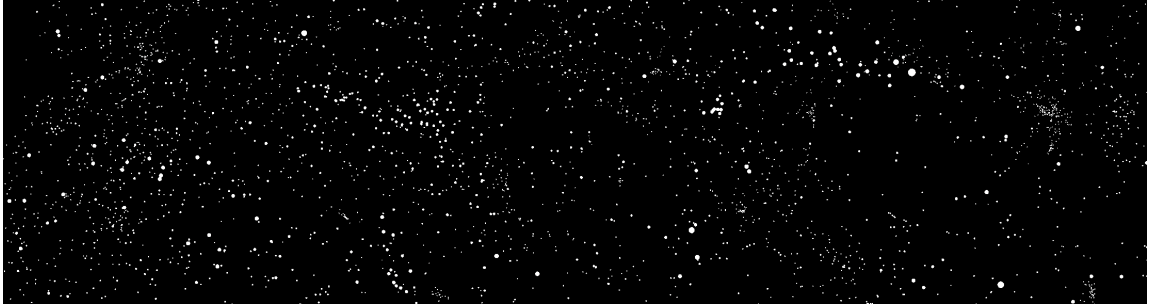
Fig. 2. A view toward the interior of GameSpace. We sought to evoke the aesthetics of stargazing.

3 RELATED WORK

Our project here extends the growing body of work in which techniques from machine learning and NLP are utilized to generate new insights about videogame phenomena. This approach was established by José Zagal and Noriko Tomuro, who applied clustering analysis and readability metrics to a corpus of 400,000 game reviews [52, 54]. Later work by others at DePaul University submitted the same corpus to sentiment analysis [37], and used the resulting model to fuel a videogame recommender system [30]. In a follow-up to the latter project, we built a game recommender system to test the intuitive notion that people tend to like related games [43]. Other work in game studies has applied *lexicometry*—investigation into the frequency of lexical items in a corpus—to game reviews [49, 53, 55], forum posts [3, 22], developer descriptions [19], walkthroughs [41], and academic publications [31]. Another recent study also processed text from academic publications about games, but using a *topic modeling* approach [8].

In the spirit of this new approach, we have released a series of tools and visualizations that are fueled by NLP techniques. The first, GameNet, is an explorable network in which nearly 12,000 games are linked to the games to which they are most related [44, 45]. Each game's node in the network also contains a summary extracted from Wikipedia, as well as links to the game's Wikipedia article and images and videos of gameplay. An add-on tool, GameSage, allows users to freely describe an idea for a game, which it processes to automatically generate a node in GameNet that links the abstract idea to related existing games [40, 44]. GameNet and GameSage are made possible by a latent semantic analysis (LSA) model trained on a corpus of Wikipedia articles about games; GameSpace is fueled by a revision to this model whose derivation we describe in Section 4. A third tool, GameGlobs, visualizes various clusterings of the 12,000 games in our original LSA model, with integration into GameNet [42]. An additional tool in this suite, GameTree, has not yet been released—it visualizes a hierarchical taxonomy of the videogame medium in the form of a radial tree built by *hierarchical agglomerative clustering* [42]. Other than our own efforts in producing these tools and visualizations, we are aware of only one other project with similar aims: Vizmo, a "videogame browser" that uses a stacked area chart to index 600 games according to visual style and mood [24]. Released just before GameNet, its approach is driven by handcrafted metadata attribution rather than machine learning (hence its smaller database of games).

At the broader intersection of data visualization and video- games, we find projects that visualize: content pools, as design support for authors [16, 47]; player behavior, to provide for player modeling [11]; broader

gameplay activity, for purposes of game analytics [13, 28, 33]; the expressive range or possibility spaces of procedural content generators [21, 48]; and even the field of games research itself [31]. Our project diverges from these efforts in that we do not visualize videogame phenomena so much as the medium itself.

While the area we just outlined visualizes videogame data, an emerging body of work is exploring the generation of games from open data [4, 7, 14, 15]. First articulated in [14], a *data game* "allows the player(s) to explore data that is derived from outside the game, by transforming the data into something that can be played with" (p. 1). In a sense, GameSpace is a data game that transforms our LSA model into something playable, though it is not especially gameful. As we discuss in Section 7.5, we are interested in the idea of *making data playable*, and we have ideas for injecting more gameful elements into the application. Relatedly, we would like to reference *Game Studies* [27], a game that provides a playable interface to the history of the academic study of games (rather than the medium itself).

In expressive visualization more broadly, we find that the journalistic repertoire has expanded to include rhetorical visualizations called *data stories* [46], and other creative practitioners are producing *explorable explanations* of real-world phenomena [50]. Though our project is less rhetorical and simulationist, GameSpace is in a sense an explorable explanation of the videogame medium.

Finally, we would like to connect our current project to studies of game genre (*e.g.*, [1, 3, 18, 26]). Whereas naturally most genre typologies partition the space of games top-down, according to gameplay concerns, we proffer an alternative bottom-up approach in which genre is a continuous notion and games are deemed alike to the degree that they are described similarly. GameSpace instantiates this notion by positioning games that are described more similarly nearer to one another in a continuous 3D space. In Section 7.1, we discuss this notion more deeply.

4   METHOD

In this section, we describe our full processing pipeline, which proceeds from the extraction of a collection of Wikipedia articles about games to the derivation of a three-dimensional space containing those games. Additionally, we include some details related to the implementation of a game engine for exploring the derived space. Broadly, our approach can be thought of as working from a simple premise: *games that are described similarly are likely, in fact, similar.*

4.1   Corpus Extraction

Our process began with the extraction of 21,456 Wikipedia articles that each pertained to an individual game. We found these by scraping Wikipedia category pages for each year in the history of the medium—*e.g.*, `Category:1977 video games`—each of which contain links to all the existing articles for games released that year. Of this full batch, 5,534 games were removed for being less than 250 words in length; in earlier work, we found that very short articles are not robust to our methods [45]. Additionally, we detected and removed 663 duplicate articles. In the end, our corpus comprised 15,259 articles, each ranging between 250 and 10,436 words, with an average length of 1,118 words.[2] In May 2014, we carried out a similar procedure to extract 11,829 usable articles [45]; this means that approximately 3,500 more usable videogame articles have appeared in the last three years. Later, in Section 4.6, we discuss our plans to automatically extract

---

[2]The longest article was for *Overwatch* (2016).

and process new articles as they appear. Finally, we will note that our entire corpus comprises some 17M total words.

4.2   Preprocessing

Our method of corpus preprocessing roughly followed the approach outlined in more detail in [45]: we converted all text to lowercase, removed punctuation and miscellaneous Wikipedia artifacts, tokenized videogame titles and platform names, removed stop words, carried out lemmatization, and removed words that appeared in only a single document. Additionally, this time around we utilized an additional preprocessing step (prior to stopword removal): using the phrase-recognition algorithm of [32] (as implemented in the `Phrases` module of the Python library `gensim` [38]), we automatically detected and tokenized multiword phrases that appeared in our corpus. This step caused, for example, all occurrences of the common videogame phrase 'first person shooter' to be aggregated into the single token `first_person_shooter`. This is critical because latent semantic analysis, which we discuss next, uses a *bag of words* approach, which means that the occurrence of a phrase like 'first person shooter' would be insufficiently captured as discrete, independent occurrences of its three component words. From informal exploration, this algorithm appears to have worked very well at automatically recognizing common videogame phrases, and it was very easy to use. We encourage scholars in game studies to try it as a cheap way of investigating how people talk about games.

4.3   Latent Semantic Analysis

After preprocessing the corpus, we built a *co-occurrence matrix* of its terms and documents, normalized the frequency counts in that matrix using *tf–idf weighting*, and then submitted the normalized matrix to *latent semantic analysis* (LSA); for more information on these steps, see [45]. LSA is a classical natural language processing method that is often used for automatically determining semantic relatedness between text documents [10]. Rather than measuring direct overlap between the collections of words appearing in each document, LSA works by factoring a co-occurrence matrix (by performing a *singular value decomposition* [17]). One of the byproducts of this procedure is a new matrix that has fewer dimensions, each specifying a weighted bundling of terms that have similar distributions in the corpus. Remarkably, this reduced matrix encodes information that may not be present in the original full matrix. For example, LSA may be able to infer that two terms that do not appear together in any document—perhaps dialectal variants that denote the same thing, like 'gas' and 'petrol'—are in fact highly semantically related [23]. By the same token, it may infer the semantic relatedness of two documents that have no terms in common. This ability to learn global associations from local co-occurrences is the hallmark of LSA and the reason for its prolonged success.

Once the reduced matrix has been derived, semantic relatedness between documents can be automatically calculated. This is typically done by taking the cosine between the documents' k-dimensional LSA vectors, where k is the number of dimensions in the reduced matrix and 'LSA vectors' simply denotes the rows in that matrix.[3] If this is not intuitive, try conceiving of an LSA model as a k-dimensional space in which each document is placed at the k-dimensional coordinates specified by its LSA vector. In this space, the semantic relatedness of two documents is reified as the distance between the documents' positions in the space—this distance is what the cosine represents. In corpora in which each document pertains to a specific

---

[3]Following an empirical investigation carried out in earlier work [45], we selected a k value of 207 for our LSA model.
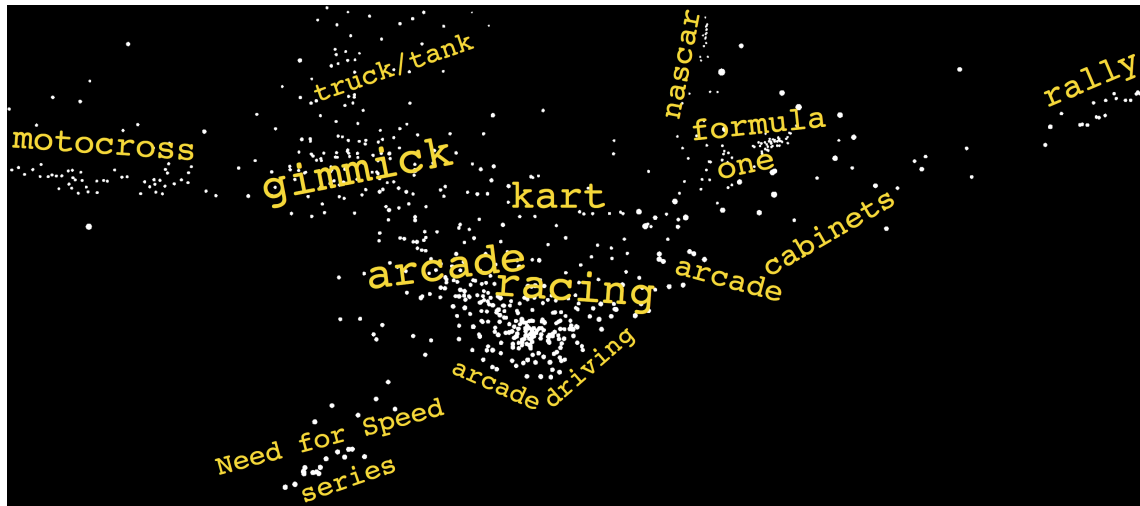
Fig. 3. GameSpace's nebula of racing games, annotated for its various component clusters. The variant of multidimensional scaling that we utilize, t-SNE, builds tight clusters of related objects and then attempts to position those clusters according to how related *they* are to one another.

individual concept, such as a corpus comprising encyclopedia entries, these relatedness scores can reasonably be utilized as a measure of the relatedness of the concepts themselves. In our project, we utilize this notion to operationalize the relatedness of documents describing games as the relatedness of the games themselves. As such, once LSA has completed, we may take any pair of games in our corpus and automatically compute a score indicating how related the games are.

In our earlier work, the processing stopped there and we made our high-dimensional LSA model explorable through GameNet [45] and GameSage [40]. As we describe next, the current project takes things a step further by deriving a lower-dimensional approximation of our LSA model, for the purposes of producing a more engaging visualization.

4.4   Multidimensional Scaling

After using LSA to derive a 207-dimensional space, we used a variant of *multidimensional scaling* to produce a three-dimensional approximation of that space. Whereas dimensionality reduction in techniques like LSA is about harnessing underlying structure that may be hidden in native data, multidimensional scaling aims to produce lower-dimensional approximations of high-dimensional spaces purely for the purpose of visualization [9]. As such, it is not uncommon to first submit data to a technique like LSA before then applying a variant of multidimensional scaling. At a high-level, multidimensional scaling tends to work something like this. First, a set of objects (in our case, the games in our corpus) are randomly distributed in a k-dimensional space (in our case, $k = 3$). The space then evolves over a series of steps where, on each step, each object is able to move in the direction that will take it closest to the objects it is related to (in our case, in terms of cosine similarity in our LSA space) and farthest from the objects it is not related to. The last step occurs when no object wishes to move anymore.
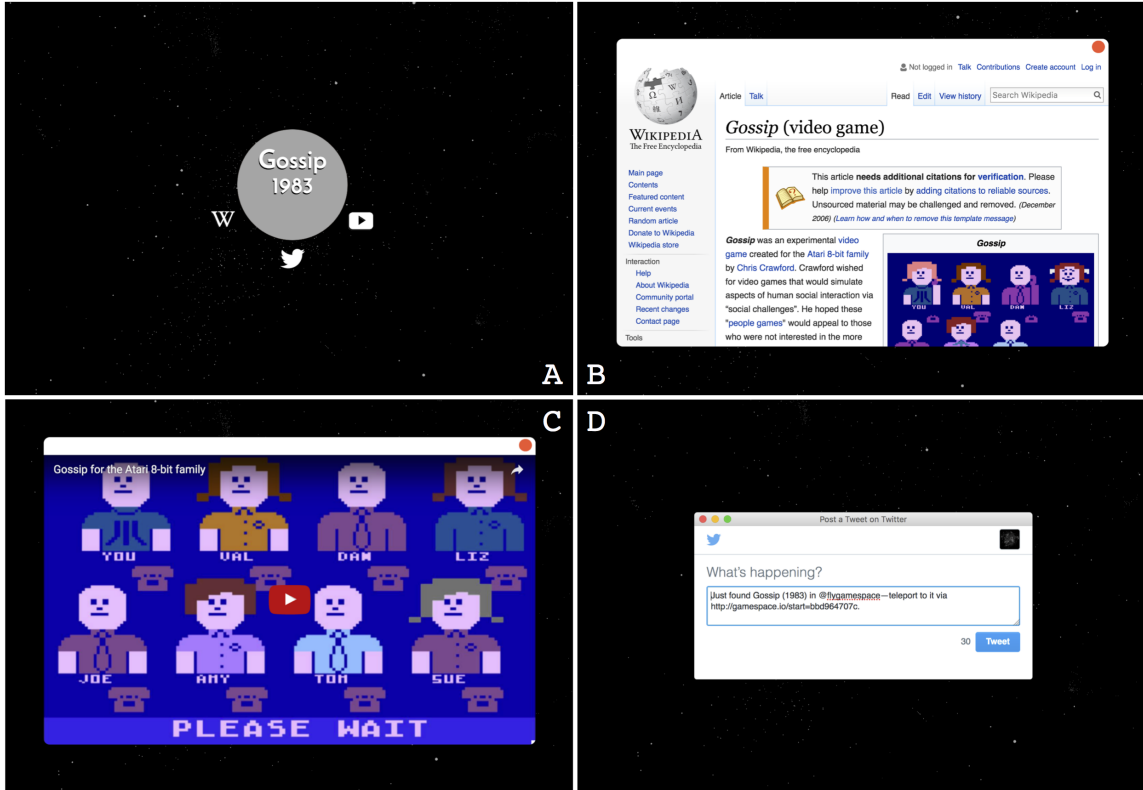
Fig. 4. A GameSpace user locates and selects the Chris Crawford game *Gossip* (A). This allows her to view its Wikipedia article (B) or watch a gameplay video (C), in place, without leaving the space. Additionally, she may tweet out a special link that provides a portal to the game's position in the space (D).

Specifically, we used a variant of multidimensional scaling called *t-distributed stochastic neighbor embedding* (t-SNE) [29], which has found widespread use since its introduction in 2008 [51]; the particular implementation that we used is included in the Python scikit-learn library [35]. We submitted a matrix of pairwise affinities (cosine similarities between games in our LSA space) to the technique using the following hyperparameter values: `perplexity=30.0`, `learning_rate=200.0`, `early_exaggeration=4.0`. Our model converged after 364 iterations, with an error of 4.55. Later, in Section 6.1, we discuss how we validated our eventual three-dimensional space by comparing it against ones derived through various techniques and hyperparameter configurations.

4.5 App Implementation

After deriving our 3D space, we built a web application that allows users to explore it—this is GameSpace. For rendering, we used `three.js`, a JavaScript library for 3D graphics [6]. The application itself is deployed using the Python web framework `Flask` [20]. Additionally, in pursuit of the third design goal in Section 2, we devoted hundreds of person hours to sound design and visual polish.

4.6   Recurring Updates

New games will be added into GameSpace automatically (and periodically) as new content appears on Wikipedia. We have set up a process that will scrape Wikipedia at recurring intervals in search of new videogame articles—*i.e.*, altogether new articles, or articles that have been expanded beyond 250 words— which will then be automatically processed and added into the 3D space. Specifically, newly extracted articles are added into the corpus, which triggers derivation of an updated LSA model, which is used to add new games into the application. As an ad hoc method, games will be placed in the 3D space next to their nearest neighbor in the high-dimensional LSA space. At less frequent intervals, we may carry out MDS from scratch, though this could become disruptive to users who become familiar with the space's topology.[4] In Section 7.4, we discuss how this recurring process makes GameSpace a *living visualization*, which we hope will encourage authorship of new videogame articles on Wikipedia.

5   GAMESPACE

GameSpace visualizes the videogame medium as a 3D space that users can explore using conventional game controls. It has a space theming: the medium is a galaxy of stars—shown in Figures 1 and 2—each of which represents an existing videogame. As described in Section 4.4, games are positioned in the space such that related games are near one another. A byproduct of this, when rendered as a galaxy, is that clusters of related games form constellation-like structures, which themselves may combine to compose larger nebulae corresponding to conventonal notions of game genre or subgenre. To illustrate this, Figure 3 shows the racing nebula, with its various component clusters annotated.

5.1   Affordances

A user can freely move about the space by flying, and at any point she is near enough to a game, she can click on it to *select* it. This notion of unconstrained movement in the space is meant to support the second design goal that we listed in Section 2, which is to encourage spontaneous and serendipitous discovery. Upon selecting a game, the camera locks into orbit around the star and displays the game's name and year of release (see Figure 4A). Additionally, the user may view the game's Wikipedia article (Figure 4B) or watch a video of its gameplay (Figure 4C)—critically, she can do this *in place*, without having to leave the space, which satisfies the fourth design goal that we listed in Section 2. Finally, an icon is provided that, when clicked, generates a draft of a Tweet containing a special link that may be used to transport directly to the selected game upon starting the application (Figure 4D). The latter affordance is meant to engender discussion around interesting games that users may find in the space—the fifth design goal that we listed in Section 2.

6   VALIDATION AND EVALUATION

In this section, we validate GameSpace in two ways. First, we describe the process by which we selected a particular variant of multidimensional scaling that produced a space with desirable characteristics. Second,

---

[4]Rederiving an MDS model with a different random seed, let alone a new objects and affinities, can produce an entirely different space.
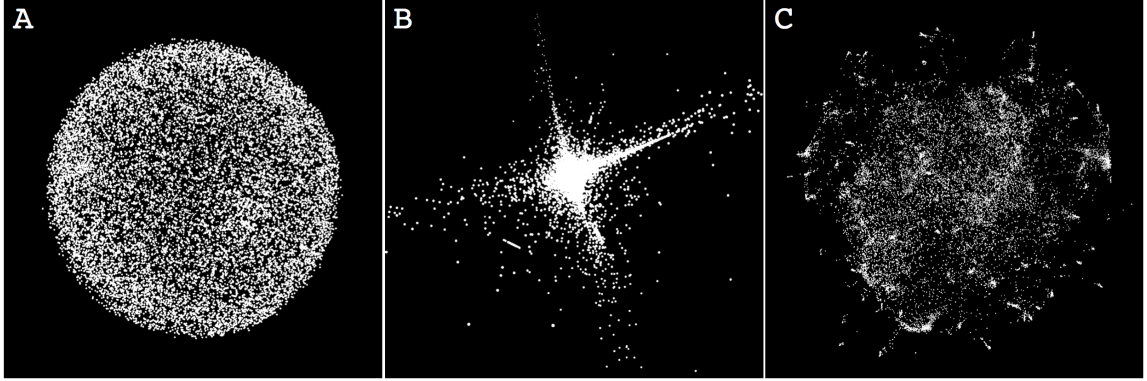
Fig. 5. Our eventual 3D space (C) along with some earlier experiments. Using metric MDS with a Euclidean distance metric produced a hollow sphere (A); this is an artifact of using Euclidean distance on certain datasets. Most MDS techniques are susceptible to a dense clustering of objects around the origin point, as seen in the space derived using locally linear embedding (B). Our final space was derived using t-stochastic neighbor embedding, which is known for avoiding this shortcoming.

we compare user metrics automatically collected by both GameSpace and our earlier tool, GameNet, to support the notion that the new tool is more engaging for users (which was our core design goal).

6.1 Selecting a Space

As explained in Section 4.4, GameSpace is critically enabled by the use of multidimensional scaling to reduce our 207-dimensional LSA model (described in Section 4.3) to a 3D space that may be visualized. In this section, we we will explain how we selected our eventual space.

6.1.1 Method. To explain how we settled on our derived space, we must first explain more about the purpose and practice of MDS. Whereas techniques like LSA are meant to uncover hidden patterns in high-dimensional data, MDS is all about harnessing the innate sensemaking capacity of the human visual processing system—a good MDS model is simply one that makes immediate sense to humans when visualized. As such, the process of selecting an MDS configuration (*i.e.*, the specific variant to use, and its hyperparameter values) is typically done through informal exploration [51]. Our particular method entailed setting up configurations, deriving spaces using them, rendering those spaces in our game engine, flying around them, and informally validating the placement of games using our knowledge of the medium (and researching unknown games, if necessary).

6.1.2 Results. During our exploration period, we experimented with 3D spaces that had undesirable characteristics that were eventually eliminated in our final space. For instance, using metric MDS [2] with a Euclidean distance metric produced a hollow sphere, shown in Figure 5A. One artifact of Euclidean distance is that it is sensitive to *vector norms*—for a given vector, this is characterized by its distance from the origin point (a vector with 0 for each dimension value). In our case, all the games in our model have LSA vectors that are roughly equidistant from the model's origin point, and this is what the resulting 3D space captured. But we do not care about distance from the origin so much as *where exactly* the game is

positioned in the high-dimensional space of our LSA model, and so cosine similarity (which captures the latter) proves to be the better distance metric for our purposes. After settling on cosine similarity as our distance metric, we tried a number of MDS variants. Many of the classic techniques in this area, however, are susceptible to a particular issue: objects tend to cluster densely around the origin point of the derived space, as in the space derived by *locally linear embedding* [39] shown in Figure 5B.[5] This is because most MDS variants attempt to simultaneously move objects toward similar objects and away from dissimilar ones—if this is not feasible for a given object (since there are only three axes for movement), all prospective movement of it in the space becomes equally (un)appealing, and the object stays relatively put near the origin point. A core appeal of t-SNE, the technique we ended up using, is that it overcomes this deficiency by ignoring dissimilarity to focus only on object similarity—*i.e.*, it favors the preservation of local structure over global structure [29]. Specifically, it seeks to maintain (in the low-dimensional space) neighborhoods that exist in the high-dimensional space, and then works to position those neighborhoods according to how related *they* are to one another. The result, as seen in Figure 5C, is that related objects cluster together, and related clusters may themselves cluster together (as in the racing nebula in in Figure 3), but with the unavoidable drawback that unrelated clusters may be positioned nearby one another. In the end, we care more about related games appearing near one another than unrelated games appearing far apart, and so t-SNE produced the most appealing space. As for the particular hyperparameter values that we chose, we again made decisions through informal exploration, following the suggestions of [51].

6.2 Assessing Engagement

As noted in Section 2, our primary goal in creating GameSpace was to maintain the benefits of visualizing models built by applying LSA to text about games—benefits that we have validated elsewhere in evaluations of GameNet and GameSage [40, 45]—while critically boosting the level of *user engagement*. Specifically, we sought to create a new tool that would be more engaging to users than our primary existing tool, GameNet [45].[6] In this section, we compare GameNet and GameSpace in terms of user engagement.

6.2.1 Method. In the research community that builds and develops web applications, there is a belief that evaluation instruments such as interviews or questionnaires are too subjective to provide a reliable measure of user engagement [25, 36]. The idea is that users have trouble self-reporting engagement, and so two alternative objective approaches are advocated [25]. The first approach utilizes biometric instruments such as eye tracking and heart-rate monitoring to discern signals that are known to be associated with engagement. While this approach can be effective, it is costly and only feasible for studies with small samples. The second approach, which is more frequently taken, is to remotely track user behavior in the wild and then analyze the telemetry data using validated metrics for measuring user engagement. Here, common metrics include session duration, user return rate, and number of clicks [25, 36]. Both GameNet and GameSpace track user behavior, but unfortunately only GameSpace logs this behavior using a notion of user, which makes a comparison of user return rate impossible. Moreover, the applications are different enough in kind that it does not make sense to compare click behavior. As such, we are left with one metric—fortunately an

---

[5]This space was used for an initial prototype of GameSpace that was reported in [42].
[6]Note that we view GameSage as an alternative interface to GameNet, meaning both systems integrate into a single tool called GameNet. Moreover, sessions with either tool are logged into the same database.

Application	Sessions	Session Duration
		mean (standard deviation)
GameNet	32,077	73.4s* (290.7s)
GameSpace	7,358	109.2s* (144.4s)

Table 1.   A comparison of user engagement between GameNet and GameSpace: GameSpace user sessions are significantly longer on average ($p < 0.0001$).

important one, session duration—though comparing this across the tools requires us to first operationally define a notion of session duration in each application. A primary issue here is that GameNet does not detect when users exit the tool. Given this, we defined a GameNet session as terminating on the last user event before five or more minutes of inactivity. GameSpace *does* detect users exiting it, but users may leave the application idle. To support a fair comparison, we defined GameSpace sessions as also terminating on the last user event before five or more minutes of inactivity (or upon exit of the application, naturally). Using these definitions of session length, we compared means across the applications using a standard left-tailed z-test, since sample sizes are large for both tools.

6.2.2   Results. Our results are shown in Table 1. We found that GameSpace user sessions are on average significant longer than GameNet user sessions ($M = 109.2$ vs. 73.4 seconds; $z = -1.64$, $p < 0.0001$). As session duration is both intuitively and conventionally [25, 36] a metric for evaluating web applications for their levels of user engagement, these results indicate that GameSpace improves upon GameNet in this area. This suggests that our core design goal, articulated in Section 2, has been satisfied.

7   DISCUSSION AND FUTURE WORK

In this section, we wrap up the paper with discussion of GameSpace and our larger project in the context of various pertinent or interesting concerns. Additionally, we forecast potential avenues for future work and collaboration.

7.1   Similarity versus Relatedness

Instead of using a notion of game likeness that is based purely on gameplay, we use one that takes into account any aspects of broader videogame ontology that may show up in a Wikipedia article: designer, developer, publisher, platform, critical reception, theme, and so forth. In [43], we articulate this model of game likeness, which we call *game relatedness*, as a bottom-up alternative to commercial game genre. As we explain more deeply in that paper, our idea is influenced by the distinction in computational linguistics between *semantic similarity* and *semantic relatedness* [5]. Concepts that are semantically similar are strongly alike in form or meaning—for instance, *mouse* and *rat*, *hot* and *warm*—whereas concepts that are semantically related may be so due to any type of association—*e.g.*, *mouse* and *cheese*, *hot* and *volcano*. As such, semantic similarity actually represents a special case of the more general notion of semantic relatedness, which is to say that all concepts that are semantically similar are also semantically related, but not vice versa. Following this distinction (and terminology), we consider games whose gameplay is alike to be *similar games* (and thus also
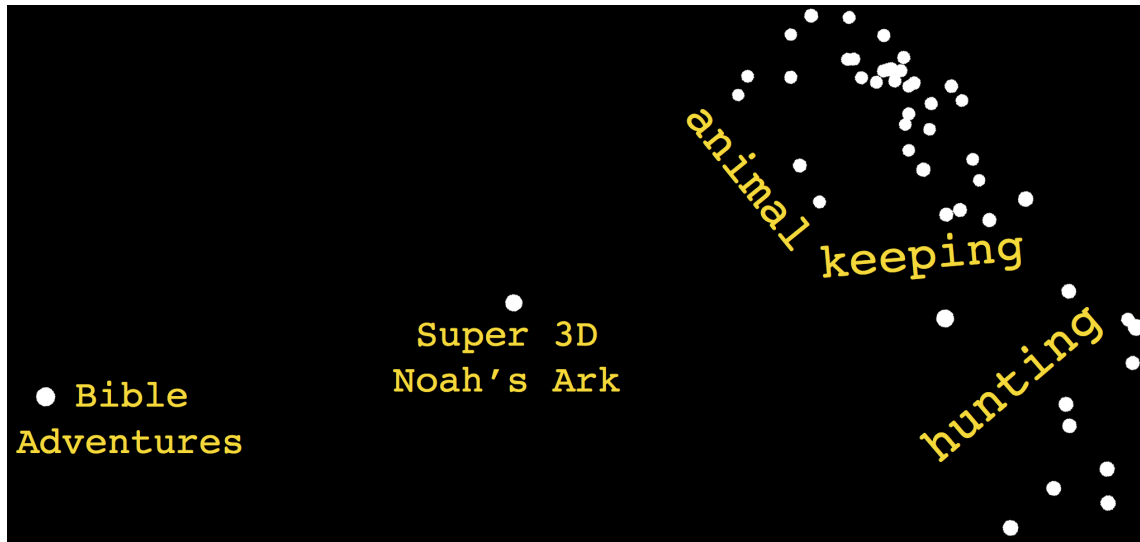
Technical Report UCSC-SOE-17-14

Fig. 6. Game genre as a continuous space. *Super 3D Noah's Ark* (1994) is placed between Christian games and animal-keeping games, for obvious reasons, and it is also near hunting games—as a *Wolfenstein 3D* clone, it is mechanically a first-person shooter, like most hunting examples.

related games), whereas, as we have explained, related games are not necessarily alike in terms of gameplay, but share other ontological features. We note that it is similar to Lindsay Grace's posited model containing both *game types* (similar gameplay) and game genre (similar content and theme) [18], except that it extends to include *all possible associative features* in a single heterogeneous, infinitely extensible set of dimensions. Semantic relatedness is sometimes talked about (and measured) in terms of the likelihood that one concept will call to mind the other [34], and so another way of understanding game relatedness is that it is a notion of how likely it is that one game will evoke the other in the mind of a player who is familiar with both. Thus, in GameSpace, games are positioned such that they may call their neighbors to mind.

7.2 Game Genre as a Continuous Space

Beyond associating games according to any and all features that may overlap, our approach is novel in that it represents game genre as a *continuous space*. In most typologies, genre is discrete, meaning games can only have binary associations with one another: they have a genre in common, or they do not. In our approach, games are alike to the degree that they are described similarly. Because GameSpace translates such affinities between games as distances in 3D space, it literally renders the notion of genre as a continuous space. A game's genre identity, then, can be thought of in various ways: its exact coordinates in the space, its placement in any apparent constellations, or the result of triangulating between all the games that are nearby it. Figure 6 nicely illustrates these concerns using the case of *Super 3D Noah's Ark* (1994), which is positioned in GameSpace between Christian games, animal-keeping games, and hunting games.[7] Further, when genre is a continuous space, physical isolation reifies conceptual obscurity, as illustrated in Figure

---

[7]If the proximity of hunting games seems unfitting, consider that *Super 3D Noah's Ark* is actually a *Wolfenstein 3D* clone, making it (mechanically) a first-person shooter (like most hunting games).

Fig. 7. One of the most isolated games in GameSpace. In our approach, physical isolation reifies conceptual obscurity, and this sports game about cultural heritage, with playable halftime shows and educational content, fits the bill. Its closest neighbors are other American football games, but it is off by itself.

7. Relatedly, empty spots in the space might be seen as constituting *unexplored design space*—one fun project that we have considered would enable the generation of synthetic Wikipedia articles from arbitrary coordinates in GameSpace.

7.3 Videogame Embeddings

A hot area of research in the digital humanities and in computational linguistics more broadly utilizes *word embeddings*. This approach belongs to the larger class of techniques called *distributional semantics*, of which LSA is a classical example. With word embeddings, the meanings of individual words (or phrases) are represented as vectors that are derived according to the words' distributions across a large corpus of text. Here, we represent games as vectors that are derived according to the how the games are described in a large corpus—this could be characterized as a *videogame embeddings* approach. In the area of word embeddings, the hottest technique is called *word2vec* [32]. In currently unpublished early exploration, we have encountered some impressive results from a word2vec model that we trained on text about videogames, which allows for arithmetic queries whose operands are videogame concepts. Our favorite example is a query $super\_mario\_bros - bowser + legend\_of\_zelda$, for which our model returns *gannon*. We hope to eventually release a tool that makes this model interactive, but in the meantime we encourage others to explore the application of word2vec to videogame text.

7.4 A Living Visualization

In Section 4.6, we outlined our recurring process that will automatically cause new games to be added into GameSpace as new descriptive content appears on Wikipedia—this will make GameSpace a *living*

visualization, specifically one that grows and evolves organically as the set of Wikipedia articles describing videogames grows and evolves. While we generally find this notion to be intriguing, we specifically hope that this will help to encourage authorship of descriptive content about videogames on Wikipedia. For instance, if a user discovers that a particular game is not in GameSpace, she can author a Wikipedia article for it, which will cause the game to automatically be added into the space. This feedback loop would mutually improve both our model and Wikipedia coverage of videogames, which corresponds to the last design goal that we listed in Section 2.

7.5   Making Data Playable

While GameSpace is clearly explorable, our goal for it is to be a *playable* visualization of the videogame medium. Toward this, we have bound user affordances to conventional game controls, adapted UI and UX patterns from game design (*e.g.*, we devoted energy to sound design), and generally tried to make the experience *feel* like playing a game. Still, however, GameSpace is not quite a game. Looking ahead to the future of the application, and to a potential future with related applications, we wonder what a truly playable visualization of the medium—*a game about games*—would look like. One idea for making GameSpace more gameful would be to introduce a progression system: achievements that unlock badges (*e.g.*, one for finding the oldest game in the model) or even new affordances (*e.g.*, finding the oldest game allows the user to filter the space by year of release). In Section 7.7, we outline some ideas for new affordances that we hope to eventually implement in GameSpace, and these could potentially be integrated into a progression system. More broadly, we are interested in the idea of *making data playable*. In Section 3, we outlined data games—games that are automatically generated from open data—and explorable explanations—rhetorical interactive visualizations that explain real-world phenomena. When we say 'making data playable', we refer to a general area that includes these approaches and any others that make the act of exploring and understanding data (or complex models, more broadly) more playful.[8] In particular, we would like to see more projects in game studies that produce explorable—or, better yet, *playable*—byproducts.

7.6   Visualizing Physical Collections

An interesting extension of this project could visualize physical videogame collections, like library holdings or extensive personal collections, as 3D explorable spaces. For example, a library patron could fly around her library's holdings to select the game that she will borrow next. In earlier work, we discussed the prospect of a version of GameNet whose model is restricted to include only games present in Stanford University's Cabrinety Collection [12].

7.7   New Affordances

We are currently considering whether to include a number of new affordances. An initial prototype of GameSpace included a search bar, which allowed users to search through the games in the model to teleport directly to a selected game's location. We found that this did not encourage serendipitous exploration—one of our design goals listed in Section 2—and so we removed it. Many users have requested it recently, though, so we may choose to reimplement it. Instead of it triggering teleportation, however, we may instead have

---

[8]We note that there is of course a subtle distinction between boosting playfulness and gamefulness, but, given our current stage of exploration, we do not believe it is not worth diving into here.

it trigger a flythrough of the space that ends at the search result. That way, the user still receives some context for the location of the result, helping support serendipitous exploration. As alluded to in Section 7.5, search could possibly be integrated into a progression system, meaning that the affordance would be unlocked through a gameplay achievement. Other possible affordances we have considered would utilize structured data that we have about the games in our model. For instance, a scrubber UI element could be used to filter the games in the space by their year of release, *e.g.*, allowing a user to explore only the space of games released through 1979. The same could be done for other features such as release platform, developer, or publisher. A more ambitious new affordance would allow users to collaboratively label constellations of games, resulting in something that looks like Figure 3. Here, we would need a mechanism for determining which labels appear for whom, and where, but something like a friending system or voting system could potentially work. Whether these new additions (or ones discussed in Section 7.8) are implemented will depend on whether we (or new collaborators) can find the time and energy to return to development on this project. We would also like to note that we are open to ideas for new affordances that readers may have.

7.8   New Visualization Features

We have also considered a number of possible new visualization features, which we may or may not implement in the future. While all games in the space are currently the same size, we could represent more important games using larger stars—here, the open question is how to automatically determine the importance of a game. One possible solution is operationalizing importance as the length of a game's Wikipedia article or as the number of articles that link back to its article; of course, this would be prone to issues related to Wikipedia authoring practices, but it could potentially work well enough for our purposes. We have also considered the prospect of using color and hue semantically (*e.g.*, to cue a game's release platforms), but this could damage the game's calm visual aesthetic. Beyond data about the games themselves, we could also visualize data about how users explore the space, since we are automatically collecting coordinates at regular intervals for all sessions with the application. For instance, we could capture the paths that (a subset of) users have taken through the space as vapor trails that are visualized in another user's session. Beyond visualization, we are currently analyzing this user data and are open to collaboration with anyone who would like to conduct their own analyses.

7.9   Visualizing Other Domains

Our method could easily be applied to other domains. We are currently conceiving of a series of related visualizations, and we are interested in brainstorming with prospective collaborators.

8   ACKNOWLEDGMENTS

REFERENCES

[1] Espen Aarseth, Solveig Marie Smedstad, and Lise Sunnanå. 2003. A multidimensional typology of games. In *Proc. DIGRA*.

[2] Hervé Abdi. 2007. Metric multidimensional scaling (MDS): Analyzing distance matrices. *Encyclopedia of Measurement and Statistics* (2007), 1–13.

[3] Dominic Arsenault. 2009. Video game genre, evolution and innovation. *Eludamos* 3, 2 (2009), 149–176.

[4] Gabriella AB Barros, Antonios Liapis, and Julian Togelius. 2016. Playing with data: Procedural generation of adventures from open data. In *Proceedings of the International Joint Conference of DiGRA and FDG*.

[5] Alexander Budanitsky and Graeme Hirst. 2006. Evaluating wordnet-based measures of lexical semantic relatedness. *Computational Linguistics* 32, 1 (2006).

[6] Ricardo Cabello. 2010. Three.js. (2010). https://github.com/mrdoob/three.js.

[7] Andrew Borg Cardona, Aske Walter Hansen, Julian Togelius, and Marie Gustafsson Friberger. 2014. Open trumps, a data game. In *Proc. Foundations of Digital Games*.

[8] Samuel Coavoux, Manuel Boutet, and Vinciane Zabban. 2016. What We Know About Games: A Scientometric Approach to Game Studies in the 2000s. *Games and Culture* (2016).

[9] Trevor F Cox and Michael AA Cox. 2000. *Multidimensional scaling*. CRC Press.

[10] Scott C. Deerwester, Susan T Dumais, Thomas K. Landauer, George W. Furnas, and Richard A. Harshman. 1990. Indexing by latent semantic analysis. *JASIS* 41, 6 (1990).

[11] Anders Drachen, Alessandro Canossa, and Georgios N Yannakakis. 2009. Player modeling using self-organization in Tomb Raider: Underworld. In *Proc. Computational Intelligence and Games*. IEEE, 1–8.

[12] Glynn Edwards, Eric Kaltman, James Owen Ryan, Timothy Hong, and Noah Wardrip-Fruin. 2015. Augmented Exploration of Library Videogame Holdings by Techniques from Computational Linguistics. In *Society of American Archivists Science, Technology, and Healthcare Roundtable*.

[13] M Seif El-Nasr, Anders Drachen, and Alessandro Canossa. 2013. Game analytics. *New York, Sprint* (2013).

[14] Marie Gustafsson Friberger and Julian Togelius. 2012. Generating game content from open data. In *Proc. Foundations of Digital Games*.

[15] M Gustafsson Friberger, Julian Togelius, A Borg Cardona, Michele Ermacora, Anders Mousten, M Møller Jensen, V Tanase, and Ulrik Brøndsted. 2013. Data games. In *Proc. Procedural Content Generation*.

[16] Jacob Garbe, Aaron A Reed, Melanie Dickinson, Noah Wardrip-Fruin, and Michael Mateas. 2014. Author assistance visualizations for Ice-Bound, a combinatorial narrative.. In *Proc. Foundations of Digital Games*.

[17] Gene H Golub and Christian Reinsch. 1970. Singular value decomposition and least squares solutions. *Numer. Math.* 14, 5 (1970), 403–420.

[18] Lindsay Grace. 2005. Game type and game genre. (2005).

[19] Lindsay D Grace. 2014. A Linguistic Analysis of Mobile Games: Verbs and Nouns for Content Estimation. In *Proc. Foundations of Digital Games*.

[20] Miguel Grinberg. 2014. *Flask web development: developing web applications with python*. O'Reilly Media, Inc.

[21] Britton Horn, Steve Dahlskog, Noor Shaker, Gillian Smith, and Julian Togelius. 2014. A comparative evaluation of procedural level generators in the Mario AI framework. In *Proc. Foundations of Digital Games*.

[22] Magnus Johansson, Harko Verhagen, and Yubo Kou. 2015. I Am Being Watched By The Tribunal: Trust and Control in Multiplayer Online Battle Arena Games. (2015).

[23] Thomas K Landauer and Susan T Dumais. 1997. A solution to Plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychological Review* 104, 2 (1997).

[24] Jin Ha Lee, Sungsoo Ray Hong, Hyerim Cho, and Yea-Seul Kim. 2015. VIZMO Game Browser: Accessing Video Games by Visual Style and Mood. In *Proc. CHI*.

[25] Janette Lehmann, Mounia Lalmas, Elad Yom-Tov, and Georges Dupret. 2012. Models of user engagement. In *Proc. User Modeling, Adaptation, and Personalization*. Springer, 164–175.

[26] Jonathan Lessard. 2014. Game Genres and High-Level Design Pattern Formations. In *Proc. Foundations of Digital Games*.

[27] Jonathan Lessard and Pippin Barr. 2016. *Game Studies*. http://www.pippinbarr.com/games/press/sheet.php?p=Game_Studies

[28] Yun-En Liu, Erik Andersen, Richard Snider, Seth Cooper, and Zoran Popović. 2011. Feature-based projections for effective playtrace analysis. In *Proc. Foundations of Digital Games*.

[29] Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *Journal of Machine Learning Research* 9, Nov (2008), 2579–2605.

[30] Michael Meidl, Steven Lytinen, and Kevin Raison. 2014. Using game reviews to recommend games. *Proc. Games and NLP* (2014).

[31] Edward Melcer, T Nguyen, Zhengxing Chen, Alessandro Canossa, Magy Seif El-Nasr, and Katherine Isbister. 2015. Games research today: Analyzing the academic landscape 2000-2014. *Proc. Foundations of Digital Games* (2015).

[32] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Proc. Neural Information Processing Systems*. 3111–3119.

[33] Truong-Huy D Nguyen, Magy Seif El-Nasr, and Alessandro Canossa. 2015. Glyph: Visualization Tool for Understanding Problem Solving Strategies in Puzzle Games.. In *Proc. Foundations of Digital Games*.

[34] Serguei VS Pakhomov, Lynn Eberly, and David Knopman. 2016. Characterizing cognitive performance in a large longitudinal study of aging with computerized semantic indices of verbal fluency. *Neuropsychologia* 89 (2016), 42–56.

[35] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* 12 (2011), 2825–2830.

[36] Eric T Peterson and Joseph Carrabis. 2008. Measuring the immeasurable: Visitor engagement. *Web Analytics Demystified* 14 (2008), 16.

[37] Kevin Raison, Noriko Tomuro, Steve Lytinen, and José P Zagal. 2012. Extraction of User Opinions by Adjective-Context Co-clustering for Game Review Texts. In *Proc. Advances in NLP*.

[38] Radim Řehůřek, Petr Sojka, and others. 2010. Software framework for topic modelling with large corpora. (2010).

[39] Sam T Roweis and Lawrence K Saul. 2000. Nonlinear dimensionality reduction by locally linear embedding. *Science* 290, 5500 (2000), 2323–2326.

[40] James Ryan, Eric Kaltman, Timothy Hong, Katherine Isbister, Michael Mateas, and Noah Wardrip-Fruin. 2016. GameNet and GameSage: Videogame Discovery as Design Insight. In *Proc. DiGRA–FDG*.

[41] James Ryan and Sergiy Ravnyago. 2016. A Typology of Verbs Culled From 23,000 Videogame Walkthroughs. In *Proc. DiGRA–FDG*.

[42] James Owen Ryan, Eric Kaltman, Andrew Max Fisher, Timothy Hong, Taylor Owen-Milner, Michael Mateas, and Noah Wardrip-Fruin. 2015. Large-Scale Interactive Visualizations of Nearly 12,000 Digital Games. In *Proc. Foundations of Digital Games*.

[43] James Owen Ryan, Eric Kaltman, Timothy Hong, Michael Mateas, and Noah Wardrip-Fruin. 2015. People Tend to Like Related Games. In *Proc. Foundations of Digital Games*.

[44] James Owen Ryan, Eric Kaltman, Michael Mateas, and Noah Wardrip-Fruin. 2015. Tools for videogame discovery built using latent semantic analysis. In *Proc. Foundations of Digital Games*.

[45] James Owen Ryan, Eric Kaltman, Michael Mateas, and Noah Wardrip-Fruin. 2015. What We Talk About When We Talk About Games: Bottom-up Game Studies Using Natural Language Processing. In *Proc. Foundations of Digital Games*.

[46] Edward Segel and Jeffrey Heer. 2010. Narrative visualization: Telling stories with data. *IEEE Transactions on Visualization and Computer Graphics* 16, 6 (2010), 1139–1148.

[47] Emily Short. 2016. Visualizing Procgen Text. https://emshort.blog/2016/06/13/visualizing-procgen-text/. (13 June 2016).

[48] Adam M Smith and Michael Mateas. 2011. Answer set programming for procedural content generation: A design space approach. *IEEE Transactions on Computational Intelligence and AI in Games* 3, 3 (2011), 187–200.

[49] Björn Strååt and Harko Verhagen. 2015. Top Versus Bottom: Game Evaluation from an Expert or Player Perspective.. In *Proc. Foundations of Digital Games*.

[50] Bret Victor. 2011. Explorable explanations. https://emshort.blog/2016/06/13/visualizing-procgen-text/. (10 March 2011).

[51] Martin Wattenberg, Fernanda Viégas, and Ian Johnson. 2016. How to Use t-SNE Effectively. *Distill* (2016). http://distill.pub/2016/misread-tsne.

[52] José P Zagal and Noriko Tomuro. 2010. The aesthetics of gameplay: A lexical approach. In *Proc. Academic MindTrek Conference*.

[53] José Pablo Zagal and Noriko Tomuro. 2013. Cultural differences in game appreciation: A study of player game reviews.. In *Proc. Foundations of Digital Games*.

[54] José P Zagal, Noriko Tomuro, and Andriy Shepitsen. 2011. Natural language processing in game studies research: An overview. *Simulation & Gaming* (2011).

[55] Miaoqi Zhu, Fan Zhao, Xiaowen Fang, and Christopher Moser. 2016. Developing Playability Heuristics Based on Nouns and Adjectives from Online Game Reviews. *International Journal of Human–Computer Interaction* (2016), 1–13.