People Tend to Like Related Games

James Owen Ryan[1,2], Eric Kaltman[1], Timothy Hong[3], Michael Mateas[1], and Noah Wardrip-Fruin[1]
[1] Expressive Intelligence Studio
[2] Natural Language and Dialogue Studies Lab
[3] Department of Computer Science
University of California, Santa Cruz
{jor, ekaltman, nwf, michaelm}@soe.ucsc.edu, thon6@ucsc.edu

ABSTRACT

The idea that people tend to like games that are alike is intuitive, even obvious. But is it true? Like many intuitive ideas, it may be wrong, and it could be challenging to test. While it is relatively straightforward to test how well a particular notion of game likeness predicts which games an individual will like, the difficulty lies in developing such a conceptualization that is robust enough to handle all types of likeness. In this paper, we propose *game relatedness*, which we argue is more robust than the dominant top-down notion, commercial game genre. Borrowing from the concept in computational linguistics of *semantic relatedness*, games are *related* to the degree that one calls to mind the other. Having this notion, we operationalize it by a latent semantic analysis model, which we then use to build a game recommender system that recommends the games that are most related to the ones that a person already likes. Using a conventional recommender evaluation scheme, we find that our system recommends games at an accuracy well above chance, indicating that people tend to like related games.

Categories and Subject Descriptors

I.5.3 [**Clustering**]: Similarity Measures; K.8.0 [**General**]: Games

General Terms

Experimentation, Human Factors, Measurement

Keywords

game studies, game genre, recommender systems

1. INTRODUCTION

There is a tradition in scientific inquiry of challenging intuitive notions, and indeed many of the great human findings were in their time counterintuitive. It is not obvious that the Earth is round; it is not intuitive that it revolves around the sun. As a less radical and more recent example, we find in the Monty Hall problem a true notion that even practiced mathematicians have struggled to accept [32]. So while it is intuitive that individuals like games that are alike,[1] it may not be true. It seems this notion is particularly natural to those who play games—one knows what games she likes, and it appears to her that these are alike. But this is really just a form of self-reporting, which is a notoriously bad way at getting at the facts [24]. While the truth of the matter here is not as fundamentally significant as even the Monty Hall problem, it has certain practical implications, especially for those who study and sell games. In scholarship, intuitive ideas tend to get used as unexamined presuppositions; when these are in fact false the larger arguments they support break down. And in the games industry, new releases are often marketed as being like earlier, successful titles—but what if people do not like games that are alike?

In this paper, we set out to empirically test this idea. While it is actually relatively straightforward to determine whether a particular notion of game likeness is a good predictor of which games an individual will like, there is a significant difficulty in developing a notion of game likeness that is robust to all types of likeness. Herein, we introduce *game relatedness*, a bottom-up notion of game likeness that is more robust than the dominant top-down typology—game genre. Borrowing from the concept in computational linguistics of *semantic relatedness*, games are *related* to the extent that one calls to mind the other. Having this notion, we operationalize it by a latent semantic analysis model, which we then use to build a game recommender system that simply recommends the games that are most related to the games that a person already likes. Finally, using a conventional evaluation scheme, we determine the accuracy of our recommender and utilize this to answer the motivating question.

2. GAME RELATEDNESS

In this section, we introduce *game relatedness*, a bottom-up notion of game likeness that is influenced by concepts from computational linguistics. Before we describe this notion, we will motivate its existence by highlighting cases where games that are clearly alike are not considered to be so by the dominant concept of game likeness, genre.

If we based our notion of game likeness on conventional game genres—that is, game-industry marketing categories like *racing*, *fighting*, *shooter* [11]—we would fail to capture many cases in which games are obviously alike. For instance,

[1]To clarify, we mean the notion that a person is likely to enjoy games that are like the games *she* already likes.

Proceedings of the 10th International Conference on the Foundations of Digital Games (FDG 2015), June 22-25, 2015, Pacific Grove, CA, USA.
ISBN 978-0-9913982-4-9.

Figure 1: *River City Ransom* and *Super Dodge Ball* belong to quite different genres but are still alike.

the Nintendo Entertainment System titles *River City Ransom* [29] and *Super Dodge Ball* [30] are in terms of game genre quite distant—the first is a side-scrolling beat 'em up, while the latter is a sports game. But anyone familiar with these titles would consider them highly related—they share a developer (Technōs Japan), are set in the same fictional universe, and evoke the same signature visual style (as Figure 1 illustrates), among other commonalities. Consider also *Super Mario 64* [22], a 3D platformer, and *Mario Kart 64* [21], a racing game. Though they come from different genres, these two games are alike in many ways, some of which are even obvious to the naked eye (see Figure 2). Similarly, *Chocobo Racing* [28], though not a role-playing game, is part of the greater *Final Fantasy* series and as such is related to its other games. *Super Smash Bros.* [15] is a fighting game that features many famous Nintendo characters and is therefore clearly associated with the various games (representing several genres) from which its playable characters originate. As a final example, we give *Utopia* [7] and *Intellivison World Series Major League Baseball* [8]. These games do not take place in the same fictional universe, but are still associated (though less saliently) for both being distinctive Intellivision titles that were designed by Don Daglow[2] and were each innovative in their use of simulation. Certainly, there is no shortage of examples in which games from different genres are in fact quite alike.

As a more robust notion of game likeness that can account for these types of cases, we offer *game relatedness*. While most conventional game genres are understood as groupings at the level of gameplay interactions [11], games that are dissimilar at this level can still be alike in other ways, as we have shown. For example, they may be set in the same fictional universe or feature the same characters, as in most of the examples above, or they may be alike in ways that have nothing to do with gameplay *or* fiction. Indeed, as in our final example, games can be alike for purely ontological reasons such as sharing the same designer, platform, or any number of other features. To account for this, by our concept of game relatedness, games may be *related* according to *any* type of similarity or association.[3]

In this sense, our notion borrows from the distinction in computational linguistics between *semantic similarity* and *semantic relatedness* [4, 20]. 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 *related games*), whereas, as we have explained, related games are not necessarily alike in terms of gameplay, but share other ontological features. Semantic relatedness is sometimes talked about (and measured) in terms of the likelihood that one concept will call to mind the other [23], 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.

[2] *Intellivison World Series Major League Baseball* was co-designed by Eddie Dombrower.

[3] This aspect may seem indiscriminate; we address this concern in Section 6.

Figure 2: *Super Mario 64* and *Mario Kart 64* are even visibly alike, despite their genre difference.

3. MODEL

We operationalize game relatedness using a bottom-up technique from natural language processing (NLP) called *latent semantic analysis* (LSA) [17]. More specifically, we train an LSA model on Wikipedia articles written about individual games, and this model affords direct calculation of relatedness between nearly 12,000 games. In [26], we thoroughly explain LSA and the derivation of our model, and so we only briefly recount these aspects here.

LSA is built on the assumption that words with similar meanings will occur in similar contexts, and that related concepts will be described similarly. From a large corpus of text, a *co-occurrence matrix* of its terms (the words and other tokens appearing anywhere in it) and its documents (the individual texts it comprises) is constructed. In this matrix, each row represents an individual term and each column an individual document. The cells of this matrix are populated with frequency counts, such that each cell will have a count of the number of times the term of the corresponding row occurred in the document of the corresponding column. Typically, these cell counts are transformed using *term frequency−inverse document frequency* (tf−idf) weighting, which penalizes terms for appearing in many documents and rewards them for appearing in few. At this point, the matrix can be thought of as specifying a *vector space*, in which the documents are represented as tf−idf vectors (their rows in the matrix). These will be very high-dimensional vectors, because they will have an entry representing a weighted frequency (in that document) for every term that appears anywhere in the entire corpus.

The hallmark of LSA is that it learns global associations from these local co-occurrences by reducing the dimensionality of the full weighted matrix. This is done by a technique called *singular-value decomposition* (SVD) [12], which is invoked with a hyperparameter k that specifies the desired number of dimensions. LSA's use of SVD causes the high-dimensional document vectors to become k-dimensional vectors in the space derived by the SVD. Remarkably, this allows the model to infer semantic associations that are not encoded in the full tf−idf matrix [17]. Like in other vector space models, semantic relatedness is calculated by cosine similarity. That is, the semantic relatedness of two documents is measured by taking the cosine between their k-dimensional LSA feature vectors, which are said to reside in a *semantic space*. In corpora in which each document pertains to a distinct concept, such as a corpus comprising encyclopedia entries, these relatedness scores can be used as a measure of the relatedness of the concepts themselves.

Utilizing this notion, we trained an LSA model on a corpus comprising 11,829 Wikipedia articles that each pertain to an individual game. By measuring cosine similarity between document vectors in this semantic space, we can in fact measure game relatedness between the individual titles that those documents represent. Given the nature of encyclopedic description as a text domain, this allows for games to be related according to any notable, shared aspect of their ontologies. Anything that is worth describing about a game may appear in a description of it, and if that same thing appears in another game's description, the two are related in that way. As such, using cosine similarity between game representations in this semantic space suitably operationalizes our notion of game relatedness. Indeed, we note that our model finds all the games given within the same examples in Section 2 to be very related.

4. RECOMMENDER SYSTEM

We use this operationalization of game relatedness to build a recommender system that recommends the games most related to the ones a person already likes. In this section, we give a brief overview of this area, including existing game recommenders, before detailing how our system works.

A *recommender system* (or just *recommender*) is software that predicts what else a user may like given what they are already known to like [25]. In certain applications, the system may actually attempt to predict the exact rating a user would give to a particular item by considering the ratings she has given to other items. Given the flurry of academic activity surrounding them, there has surprisingly been only three academic projects that have presented videogame recommender systems, all of which were first reported (independently of one another) in 2014.

In [27], its authors employ multiple machine-learning techniques, namely *archetypal analysis*, to build several game recommender systems. In archetypal analysis, instances in a data set get represented as mixtures of *pure types*, or *archetypes*, which themselves are represented as mixtures of the instances [6]. The authors use this technique to derive archetypes from 500,000 Steam users according to playtime data, and then represent each user as a mixture of these archetypes. From here, their specific recommendation task is to predict how long a user will play a particular game. To evaluate their system, they use a typical offline method in which they have their systems make predictions about outcomes that are already known (but withheld from the systems). From this, they report 0.86 recall.

The authors of [5] present a domain-specific system that uses case-based reasoning to recommend rehabilitation games to patients. Specifically, their system reasons about the user's personality (using a questionnaire and social-networking data) and specific medical condition to pick out a game that rehabilitates for that condition and whose genre and difficulty best match the user's personality. Their system's precision increases as the case database expands, but it appears to perform poorly when using training-set sizes that are typically used to evaluate recommenders.

The authors of [19] process a collection of nearly 400,000 user-submitted game reviews to build a system that recommends games that have been evaluated similarly to the games a user is already known to like. Starting from the unique adjectives that modified the word 'gameplay' in some review, the authors proceed to *co-cluster* these adjectives and the context words (nouns, adjectives, or verbs) that occurred nearby them. The resulting co-clusters include, for example, one that has {'great', 'amazing', 'excellent', ...} as its adjectival cluster and {'graphics', 'look', 'sound', ...} for its contextual cluster. They then represent games using *feature vectors* that specify how often particular adjectives were used to evaluate particular aspects of gameplay in reviews for that game. As such, these full feature vectors give a fairly rich specification of how each game was evaluated (according to the various gameplay aspects that are represented among the co-clusters) across all its reviews. Using an offline evaluation scheme, they report 0.86 precision—that is, 86% of the games their system recommended were indeed liked by the players being recommended to.

While the above game recommender systems were built, naturally, to recommend games at a high accuracy, we built ours to test whether people like games that are related. To this end, our system simply recommends the games that are most related (in the sense that our LSA model operationalizes this notion) to the games a person already likes. Specifically, recommendations are generated in the following way. Given a set of games that a person is known to like, our system iterates over each of the other games that are included in our LSA model to calculate average relatedness (cosine similarity) with regard to all the liked games. The system then simply recommends the n most related games, where n is specified up front.

5. EXPERIMENT

To test the notion that people like related games, we evaluate our recommender system using the method and test data described in [19]. This data specifies, for the ten most prolific GameSpot user reviewers[4] through April 2009, which games each user reviewed and which of those she liked. Because the reviews include no explicit indication of whether the reviewer liked the game, but rather a numerical rating, a reviewer is considered to have liked a game if her rating for it exceeded her median rating given across all her reviews.

To evaluate system performance, we use the offline k-fold cross-validation procedure described in [19], which works as follows. For each user, the set of games that a person likes is divided into k folds of three games each. Then, each of the k folds is iterated over, with the three games in a fold being treated as a set of seed games that are used to recommend n other games (*i.e.*, to predict n other games the user likes). To allow measurement of system accuracy, only

[4]Mean number of reviews was 166.2 across these users.

games that a user reviewed may be recommended to her. Because we already know which of these reviewed games the user actually likes, we can objectively determine the correctness of a recommendation by checking whether she indeed likes that game.[5] System accuracy is measured in terms of precision—the percentage of recommendations that were correct. Following [19], precision for a fold is averaged across iterations using all values of n between one and ten. Finally, we measure total system performance as the average precision across all users, these values themselves representing average precision across the k folds tested for that user.

As we mentioned in Section 3, the dimensionality of an LSA model is selected according to a hyperparameter k whose value is specified prior to derivation. While in [26] we enacted a conventional parameterization scheme that led to a dimensionality of 207, here we test recommender precision as the dimensionality of the underlying model varies.

6. RESULTS AND DISCUSSION

Our system recommended games with 0.7 precision. While the precision of 0.86 reported in [19] far exceeds this, our precision itself is much greater than chance, which in this task is 0.57.[6] Figure 3 plots system precision as a function of the underlying LSA model's dimensionality, which demonstrates a learning curve that fairly quickly tops out at k=63 (the dimensionality at which maximum precision was observed).

That our system simply recommends the games most related to the games a person already likes, and does so at an accuracy well above chance, indicates that people tend to like related games. Of course, there is a significant limitation to this finding. While our system generated several thousand recommendations for each user, it only did this for ten users total. As recommendation precision varied across users, we would ideally have used review data in this experiment from several more users.

As for the recommender in [19] performing much better than ours, this is not particularly surprising. Their system recommends games that were appraised by GameSpot users similarly to how the games a person already likes were appraised. If we assume most people like games that are generally well-regarded, their system might often simply recommend other games that are also thought highly of.[7] Our system, however, relies on a different underlying assumption (that people like related games) and is meant to test a particular idea (the underlying assumption) rather than to achieve better accuracy than earlier recommenders.

Lastly, we will attempt to address potential criticisms the reader may still have at this point. It could be said that our insistence that games may be related according to *any* commonality makes the notion of game relatedness so broad as to lose all analytic potential. To this we respond that the notion cannot be used analytically without first getting operationalized—we do this here by processing Wikipedia content—and when this is done, games still may be related according to any commonality, but they only *will* be related according to *attested commonalities*. As such, we believe that this inclusivity does not make the notion indiscriminate, but instead robust. The reader may also find it problematic that we use Wikipedia articles to actually represent the games that they describe. Consider that we can rarely compare things in the world except by using proxies. A Wikipedia article for a game is by no means a perfect representation of it, but we do maintain that it may serve as a good enough approximation so as to afford meaningful comparison. In other work, we have in fact worked to validate the relatedness judgements yielded by this approach [26].

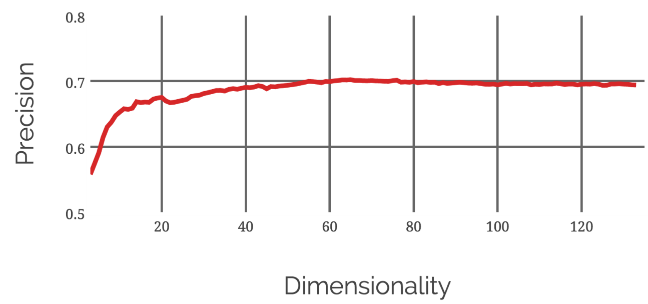

Figure 3: System precision as a function of the dimensionality of the underlying LSA model. (Chance in this task is 0.57.)

7. CONCLUSION AND FUTURE WORK

In this paper, we set out to test the idea that people tend to like games that are alike, which is intuitive but could be wrong. In order to do this, we first developed *game relatedness*, a bottom-up conceptualization that we demonstrated is stronger than the dominant top-down one, game genre, in that it can account for associations between games that come from different genres. To answer the motivating question, we operationalized game relatedness by a latent semantic analysis model, and then built and evaluated a recommender system that is driven by this model. From this evaluation, we find that the system, which simply recommends the games that are most related to the games that a person already likes, does so at an accuracy well above chance, which indicates that people tend to like related games.

While we compared game relatedness to commercial genre because the latter is the dominant top-down notion of game likeness, this, we admit, underplays academic work in this area; indeed, very many typologies have been developed in research contexts [2, 1, 3, 10, 34, 9, 16, 14, 31, 33, 18, 13]. Unfortunately, due to time and space considerations, comparing our bottom-up notion with these more nuanced top-down analogues is beyond the scope of this paper. That being said, we are currently surveying these and plan to explore this issue more deeply in subsequent work. Generally, we encourage such comparisons and hope that this line of work does well to further the bottom-up approach to game studies that we have advocated elsewhere [26].

8. ACKNOWLEDGMENTS

We sincerely thank Michael Meidl, Steven Lytinen, and Kevin Raison for graciously providing us with their test data. Our gratitude also goes to Joseph C. Osborne, whose insight influenced our model parameterization. This project was made possible in part by Institute of Museum and Library Services grant LG-06-13-0205-13.

[5]This is a standard method for evaluating recommender systems.

[6]The ten users liked 57% of the games they reviewed.

[7]Well-regardedness is itself an aspect of our larger notion of game relatedness, but in [19] it is the core concern.

9. REFERENCES

[1] E. Aarseth, S. M. Smedstad, and L. Sunnanå. A multi-dimensional typology of games. In *Proc. DiGRA*, 2003.

[2] E. J. Aarseth. *Cybertext: perspectives on ergodic literature*. JHU Press, 1997.

[3] T. H. Apperley. Genre and game studies: Toward a critical approach to video game genres. *Simulation & Gaming*, 37(1), 2006.

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

[5] L. Catalá, V. Julián, and J.-A. Gil-Gómez. A cbr-based game recommender for rehabilitation videogames in social networks. In *Proc. IDEAL*, 2014.

[6] A. Cutler and L. Breiman. Archetypal analysis. *Technometrics*, 36(4), 1994.

[7] D. Daglow. *Utopia*. Mattel, 1981.

[8] D. Daglow and E. Dombrower. *Intellivision World Series Major League Baseball*. Mattel, 1983.

[9] S. Dahlskog, A. Kamstrup, and E. Aarseth. Mapping the game landscape: Locating genres using functional classification. In *Proc. DiGRA*, 2009.

[10] C. Elverdam and E. Aarseth. Game classification and game design construction through critical analysis. *Games and Culture*, 2(1), 2007.

[11] Entertainment Software Association. Essential facts about the computer and video game industry. 2014.

[12] G. H. Golub and C. Reinsch. Singular value decomposition and least squares solutions. *Numerische Mathematik*, 14(5), 1970.

[13] C. S. G. González and V. N. Adelantado. A structural theoretical framework based on motor play to categorize and analyze active video games. *Games and Culture*, 2015.

[14] M. Greco, N. Baldissin, and F. Nonino. An exploratory taxonomy of business games. *Simulation & Gaming*, 2013.

[15] HAL Laboratory. *Super Smash Bros.* Nintendo, 1999.

[16] K. P. Jantke and S. Gaudl. Taxonomic contributions to digital games science. In *Proc. Games Innovations Conference*, 2010.

[17] T. K. Landauer, P. W. Foltz, and D. Laham. An introduction to latent semantic analysis. *Discourse Processes*, 25(2-3), 1998.

[18] J. H. Lee, N. Karlova, R. I. Clarke, K. Thornton, and A. Perti. Facet analysis of video game genres. 2014.

[19] M. Meidl, S. Lytinen, and K. Raison. Using game reviews to recommend games. In *Proc. AIIDE*, 2014.

[20] J. Morris and G. Hirst. Non-classical lexical semantic relations. In *Proc. HLT-NAACL Workshop on Computational Lexical Semantics*, 2004.

[21] Nintendo EAD. *Mario Kart 64*. Nintendo, 1996.

[22] Nintendo EAD. *Super Mario 64*. Nintendo, 1996.

[23] S. Pakhomov, B. McInnes, T. Adam, Y. Liu, T. Pedersen, and G. B. Melton. Semantic similarity and relatedness between clinical terms: An experimental study. In *Proc. AMIA*, 2010.

[24] P. M. Podsakoff and D. W. Organ. Self-reports in organizational research: Problems and prospects. *Journal of management*, 12(4), 1986.

[25] P. Resnick and H. R. Varian. Recommender systems. *Communications of the ACM*, 40(3), 1997.

[26] J. O. Ryan, E. Kaltman, M. Mateas, and N. Wardrip-Fruin. What we talk about when we talk about games: Bottom-up game studies using natural language processing. In *Proc. FDG*, 2015.

[27] R. Sifa, C. Bauckhage, and A. Drachen. Archetypal game recommender systems. *Proc. KDML-LWA*, 2014.

[28] Square. *Chocobo Racing*. Square Electronic Arts, 1999.

[29] Technōs Japan. *River City Ransom*. Technōs, 1989.

[30] Technōs Japan. *Super Dodge Ball*. [Nintendo Entertainment System]. CSG Imagesoft, 1989.

[31] M. J. Thomét. Using components to describe videogames and their players. In *In Terms of Play: Essays on Words That Matter in Videogame Theory*. 2013.

[32] J. Tierney. Behind Monty Hall's doors: Puzzle, debate and answer. *New York Times*, 1991.

[33] Z. Welhouse, J. H. Lee, and J. Bancroft. âĂIJwhat am i fighting for?âĂİ: Creating a controlled vocabulary for video game plot metadata. *Cataloging & Classification Quarterly*, 2014.

[34] J. P. Zagal, M. Mateas, C. Fernández-Vara, B. Hochhalter, and N. Lichti. Towards an ontological language for game analysis. *Worlds in Play: International Perspectives on Digital Games Research*, 21, 2007.