The Undergraduate Games Corpus: A Dataset for Machine Perception of Interactive Media

Barrett R. Anderson, Adam M. Smith
Design Reasoning Lab, University of California Santa Cruz
barander@ucsc.edu, amsmith@ucsc.edu

Abstract

Machine perception research primarily focuses on processing static inputs (e.g. images and texts). We are interested in machine perception of interactive media (such as games, apps, and complex web applications) where interactive audience choices have long-term implications for the audience experience. While there is ample research on AI methods for the task of playing games (often just one game at a time), this work is difficult to apply to new and in-development games or to use for non-playing tasks such as similarity-based retrieval or authoring assistance. In response, we contribute a corpus of 755 games and structured metadata, spread across several platforms (Twine, Bitsy, Construct, and Godot), with full source and assets available and appropriately licensed for use and redistribution in research. Because these games were sourced from student projects in an undergraduate game development program, they reference timely themes in their content and represent a variety of levels of design polish rather than only representing past commercial successes. This corpus could accelerate research in understanding interactive media while anchoring that work in freshly-developed games intended as legitimate human experiences (rather than lab-created AI testbeds). We validate the utility of this corpus by setting up the novel task of predicting tags relevant to the player experience from the game source code, showing that representations that better exploit the structure of the media outperform a text-only baseline.

Introduction

There is an extensive history of research into machine perception for many kinds of non-interactive media (e.g. text, images, video), but interactivity remains comparatively understudied. Interactivity is an important component of many modern media, including apps, complex web applications, and games. Games in particular are almost pathologically interactive. Many games are not just incidentally but primarily interactive. Further, the meaning of text, images, and other observations made while interacting with a game can be conditioned on the activity leading up to those observations. A screen that reads "game over, try again?" comments on the players's recent choices and suggests the possibility of alternate outcomes with different choices. We conjecture that a large collection of games could accelerate the development of machine perception techniques that directly address the unique features of interactive media.

This paper contributes a new, annotated corpus of 755 games. We provide full game source code and assets, in many cases licensed for reuse, that were collected with informed consent for their use in AI research. In comparison to lab-created game clones intended as AI testbeds, these games are complete works intended as human experiences, and they represent a range of design polish rather than being filtered to only commercially successful games. By offering games for a variety of platforms (from micro-games in Bitsy and text-oriented stories in Twine to the heavyweight games built in the Godot Engine), we set up future research with a wide range of technical challenges.

In the following sections, we place machine perception of interactive media into the context of emerging work on information retrieval for interactive media, humanist theories of the meaning of interactive media, and automated game-playing. We then compare our corpus with other sources of games and game-like artifacts, and quantitatively characterize the scale and variety of the corpus. To demonstrate the usefulness of just one slice of this data (interactive narratives), we setup a metadata tag prediction task and compare the performance of a system that reduces an interactive story to plain text with one that exploits knowledge of the connectivity between scenes by player selected choices. The complete data and metadata for the new corpus are available at https://github.com/barrettrees/undergraduate_games_corpus.

Background

Towards a future of search engines that could search within as well as across games, Zhang et al. recently introduced the challenge of crawling, indexing, and retrieving *moments* from the vast space of interactivity contained within each game (Zhang et al. 2018). This requires representing and reasoning about what makes one moment different from another or more or less relevant to a user-provided query. Preliminary work in this area made progress by representing moments as vectors in a space where proximity of points in space approximated their semantic similarity. By training a neural network to predict the state of a videogame's working memory from the pixels in the screenshot of a moment, the Pix2Mem technique induces an intermediate vector representation that implicitly represented spatial and temporal

aspects of moments (Zhan and Smith 2018). As with word vectors in natural language processing (Mikolov, Yih, and Zweig 2013), this representation allowed reasoning by analogy using simple operations on vectors (Zhang et al. 2018).

Variations on this vector representation strategy have been explored. For example, search with open-vocabulary, natural language queries for game moments containing recognizable objects or having topics mentioned in character dialog was demonstrated by fusing information from visual and auditory observations (Zhang and Smith 2019). This work (and past work like Pix2Mem) reduced perception of interactive media to perception of linear streams of images, audio, and text (potentially augmented with software execution state annotations). To date, little work outside of automated gameplaying (addressed shortly) has built machine perception systems that model the semantics of observations as being conditioned on past player choices.

Rooted in the humanistic field of game studies, choice poetics offers a theory of how authors of interactive media build meaning through the structure of the choices they offer to the audience (Mawhorter et al. 2018). Patterns in the framing of options and how they are linked to outcomes can construct simple "dead-end" situations that encourage the player to backtrack in search of a more desirable outcome or more subtle "unchoice" situations where the fact that all choices eventually lead to the same outcome can convey inevitability or even convey reluctance of the story's protagonist to do what the player wants.

The related theory of procedural rhetoric (Bogost 2007) attempts to capture how games convey meaning and build arguments even through potentially very abstract systems. The rules governing the production and consumption of resources under the influence of player actions, for example, can be used to convey subtle and culture-dependent messages even what the player directly observes is just colored rectangles moving in straight lines. Proceduralist readings (Treanor et al. 2011) offer one way of analyzing meaning in games employing graphical logics, and they have been applied to the interpretation of newsgames (which function like political cartoons) (Treanor and Mateas 2009). Efforts to operationalize proceduralist readings in AI systems (Martens et al. 2016) have thus far worked by taking a static game description (analogous to a videogame's source code) as input.

Many automated gameplaying systems, such as AlphaZero (Silver et al. 2017) and AlphaStar (Vinyals et al. 2019), learn to represent and reason about moments in games through direct action and observation within a game. In gameplaying systems based on reinforcement learning, it has been common to summarize a state (moment) by applying an LSTM to the stream of past actions and observations (Bakker 2002). To date these learned representations have been specific to one game at a time (i.e. not meaningfully comparable across games via vector operations) and heavily optimized for the task of choosing the next action in a high-scoring play policy rather than trying to capture human significance. Related automated game exploration algorithms (Ecoffet et al. 2019; Zhan, Aytemiz, and Smith 2019) eschew strong play in favor of covering a diverse sample of a game's interactive space, but do not interpret what they find.

Related Corpora

Partlan et al. (2019) recently proposed a formal representation for interactive narratives that distinguished scenes within a larger story. Significantly, their representation was automatically constructable from a game's definition (by running something like exhaustive symbolic execution of the game's scripting logic) *and* validated by an expert interview study designed to highlight structural features of narratives that had human-level significance. This work was in turn based on a collection of 20 student-authored interactive narratives (Partlan et al. 2018) built on the StudyCrafter (Harteveld et al. 2016) platform.

Our contribution of the Undergraduate Games Corpus aims to accelerate research on the development of perceptual representations of interactive media (comparable within and across individual works) in a way that is anchored in human meaning-making processes and the work of a diverse collection of authors.

Several corpora of games (like the 20 StudyCrafter stories mentioned previously) exist and in some cases have already been used in AI research. A corpus of all known historical games (130 total) for the ZX Spectrum version of the Graphic Adventure Creator, a 1986 game development tool advertised to nonprofessionals, was published (Aycock and Biittner 2020) while this article was under review. In the history of AI, the effort to build one system (or at least devise one technique) that could competently interact with multiple games—otherwise known as general game playing (GGP)—prompted Pell's Metagamer (1996), a system that could play any symmetric, chess-like games that could be modeled in the SCL-Metagame language. Pell's dissertation (1993) included a collection of such games in the appendix. This thread was followed by the commercial release of the Zillions of Games "universal gaming engine" (Lefler and Mallett 1998-2020) which allowed competitive human play (including manipulation of artist created pieces) against an AI opponent able to play hundreds of bundled board games with the software. As of the time of publication, there are 2995 free add-on games available for the ZOG engine.[1]

Academically, GGP research led to the development of the GDL (Love et al. 2006) and GDL-II (Thielscher 2010) game description languages. Collections of games in these formats are made to evaluate agents submitted to annual GGP competitions (Genesereth and Björnsson 2013). These collections narrowly focus on turn-based games in the styled after classic board games (modestly extending Pell's "symmetric, chess-like" scope of application). As such, the associated representation languages are ill-suited to representing certain common elements of videogames like free-form text, graphics, or synchronous real-time interaction. The more recent VGDL (Ebner et al. 2013) attempts to fill this gap, and it is provided with a collection of game formalizations inspired by popular classic videogames (Perez-Liebana et al. 2016). Whether for GDL or VGDL, however, the game collections almost always represent the work of AI researchers to model elements of classic games that are most relevant to the operation of their AI systems rather than, say, the perceptual

[1] http://www.zillions-of-games.com/games/index.html

elements that made those games popular notable originally.

Where should we look for large and diverse collections of games made by human game designers and designed for human audiences? In 2021, mobile app stores are a natural first place to look. Indeed, machine perception of the experience of arbitrary items in an app store might have large commercial impacts by improving app search. Limited automated interaction with app store items is already improving malware detection in app stores (Odusami et al. 2018). From a research perspective, however, this collection is too varied to work with in its entirety. Further, even if a sufficiently narrow space of games can be pulled from an app store, these games are unlikely to be licensed in a way that is compatible with this research (which might depend on forms of reverse engineering proscribed by license agreements). Source code is also not generally available for such games. These conditions prohibit direct use of similar game collections, including those that even more narrowly focus on videogames, such as Itch.io,[2] or text-based interactive fictions, such as those games indexed by IFDB.[3]

Recently (Morrow and Casucci 2019), the Marriott Library at the University of Utah announced a plan to collect, index, and archive videogames created as thesis projects by students in the Entertainment Arts & Engineering program. This effort is motivated by digital scholarship: allowing future researchers to examine and interpret the bit-precise details of past games. These games, when eventually made available, might solidly anchor machine perception work in authentic interactive media (rather than game clones created by AI researchers to demonstrate their AI systems). However, many of the concerns that apply to mobile app stores apply here as well. Further, the tendency for ambitious thesis games to try out the latest interaction design trends (e.g. augmented reality) make thesis game collections much more varied than the traditions of clean and consistent representation formalisms of GDL/VGDL would allow.

There is a clear need for a large and diverse archive of authentic games. At the same time, we seek some level of technical consistency so that new AI techniques can be applied at scale without needing to be generalized to work on arbitrary software first. Once these needs are met, research could be accelerated by also providing the kind of consistently structured metadata like tags and description texts that might give AI systems clues as to what is most notable from a human perspective. The ethical considerations of building such a dataset should not be overlooked either (Leidner and Plachouras 2017). The archive should neither misleadingly erase the labor of people who contributed to it—participation-washing (Sloane et al. 2020)—nor problematically represent people or work that was never intended to be used in this kind of research (Prabhu and Birhane 2020). Our construction of the Undergraduate Games Corpus offers one way of navigating these constraints.

[2]https://itch.io/

[3]https://ifdb.tads.org/

Corpus Construction

To construct the Undergraduate Games Corpus we solicited contributions from undergraduate students in large introductory game design courses at UC Santa Cruz. These students, who are not required to have any previous programming or game design experience, create 2–3 complete games by the end of the course. Students worked on these game projects over several weeks (both individually and in small groups), including at least one round of both peer and expert feedback on their drafts, and these projects accounted for the majority of their course grade.

Soliciting the games that students were creating for this course for the corpus proceeded in two phases that were incorporated into the course. In the first phase, the students were given a lecture regarding the emerging landscape of technical games research (Nelson 2020). This lecture incorporated examples of AI projects enabled by data, such as the GameSage game recommendation engine (Ryan et al. 2016) and the GameSpace exploration system (Ryan et al. 2017), and invited them to contribute their own games towards similar research projects in the future. Students specifically saw examples of how individual games could be plucked out of a large space for inspection using simple text queries. In the second phase, the students completed a form indicating their willingness to have their own games included in the corpus. If they agreed, they were also asked to provide a game description, provide appropriate keywords/tags, and if they opted for a Creative Commons license make some decisions about how they would like their game to be distributed. Responding to this form, even only to indicate that they did not wish for their games to be added, was part of the students' participation grade for the course.

Corpus Characterization

The Undergraduate Games Corpus consists of 755 individual games. These are complete, playable games, created by student authors in introductory game design courses at UC Santa Cruz between July 2019 and July 2020. In many cases, these games reflect student's lives (e.g. their interests and personal struggles) and the trends of the cultures around them (e.g. which pre-existing games they may choose to clone). The overlap of this specific student population and these specific dates strongly shaped the themes of many of the games (e.g. relating the experience of local wildfires, power outages, labor strikes, and the global emergence of the COVID-19 pandemic). Students conveyed these themes in methods specific to interactive media, potentially employing techniques like the procedural rhetoric of failure (a topic covered in course lectures) (Treanor and Mateas 2009) to convey inevitability by having multiple choice lead to the same undesirable outcome. The subtle topic of anxiety (unsurprisingly common in student games) could be conveyed by embedding many dead-end choices in a story, allowing the player to interactively worry along with the protagonist as they see many different outcomes play out in doom.

All games in the corpus were initially evaluated as assigned projects in a game development course. This means they have a narrower scope than typical commercial games.

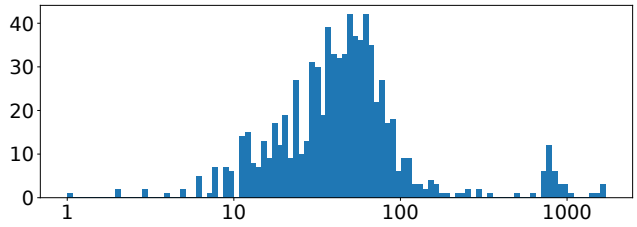

Figure 1: Game description word counts.

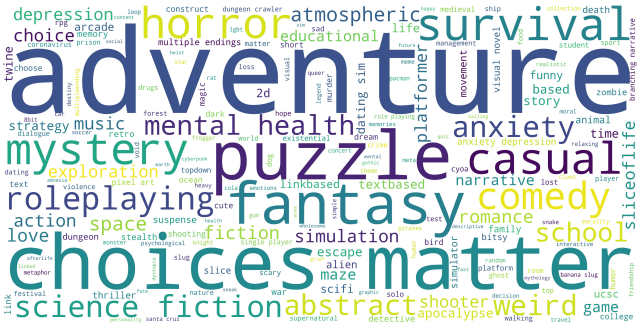

Figure 2: Game tags word cloud; size reflects frequency.

However, all resulted in passing grades and we conservatively estimate that >90% are complete human experiences. We intentionally do not provide an API for playing these games automatically, evaluating progress within games (e.g. via a score), or even identifying the kinds of input a game is expecting (e.g. mouse or keyboard). Part of the challenge of working with interestingly messy real-world playable artifacts is even figuring out how to play them in the first place. Compared to the challenge of trying to play unorganized games from a site like Itch.io (where even figuring out how to download and install each game is an idiosyncratic process), the games in our corpus fall into manageable number of interaction paradigms.

We envision future systems that are able to infer aspects of the player experience from direct inspection of game project materials. Towards this goal, we characterize the descriptive metadata provided with each game. Most (about 85%) of games come with student-author provided descriptions. Similarly, most (about 70%) of games come with descriptive tags. The histogram in Fig. 1 visualizes the distribution of word counts in descriptions while the word cloud in Fig. 2 characterizes the distribution of descriptive tags.

Some of our games are closer to being represented in well understood formats than others. The majority of games in the corpus (about 61%) are primarily textual, created in the Twine[4] game engine. Twine games (more often called stories) are represented by collections of passages that usually represent individual scenes in the story. While student games made extensive use of (hypertext) links to present the player (interactive reader) with choices, some students experimented with using the engine's scripting language to generate player choices procedurally (modeling combinatorial spaces that would be unreasonable to express with a manually constructed network of hyperlinks). Almost all Twine games make some use of dynamic scripting logic, even if it is only to slightly alter the text for passages once they have been seen by the player once (e.g. to omit the verbose description of an object once it has been introduced).

[4]http://twinery.org/

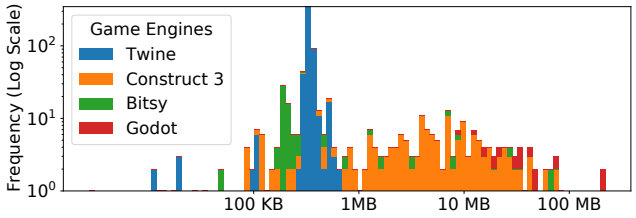

Figure 3: Distribution of game project sizes (including source code and assets), also representing the variety of game types by authoring tool.

Our corpus also includes graphical games. About 11% of the corpus was created using Bitsy,[5] an authoring tool for "little" games that exposes a very tightly scoped scripting language (whereas Twine authors can draw on all of JavaScript when needed). About 25% of the corpus was created with Construct 3,[6] a relatively flexible tool for making games with two-dimensional graphics that run inside web browsers. The remaining 3% build on the Godot[7] game engine, an open source tool comparable to Unity[8] in terms of flexibility and support for advanced, three-dimensional graphics rendering. From smaller Twine stories to larger Godot games, the artifacts in our corpus have a wide distribution in weight and complexity. Fig. 3 characterizes this diversity in terms of total project file size, which might be read as a crude proxy for the technical complexity of machine perception for those games.

Focusing specifically on the largest segment of our corpus, Twine games, Fig. 4 characterizes the distribution of sizes of Twine games in terms of number of passages per story and number of words per passage within each story. By defining words simply as the number of whitespace-separated tokens, we intentionally include words that are not directly seen by the player such as those contributing to scripting logic or to the visual formatting of the text (or even references to image files). All of these words represent effort by the author to shape the audience's perceptual experience.

Every game in our corpus is attributed to one or more specific student authors under their chosen names, and all are available for public consumption (while respecting the author's copyright). Further, many (about 54%) of our games are provided under a Creative Commons license that grants further rights,[9] such as to remix and transform the work, under the condition that appropriate credit is given to the original authors. Table 1 summarizes the distribution of chosen

[5]https://ledoux.itch.io/bitsy
[6]https://www.construct.net/en
[7]https://godotengine.org/
[8]https://unity.com/
[9]https://creativecommons.org/licenses/

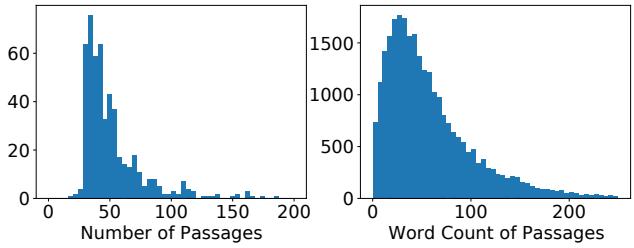

Figure 4: Scale of game for just the Twine stories, representing the number of scenes (passages) and the detail within each scene (words and source code tokens per passage).

License type	Number of games
Copyright only	*342*
CC-BY	*219*
CC-BY-SA	*15*
CC-BY-NC	*73*
CC-BY-NC-SA	*4*
CC-BY-NC-ND	*2*

Table 1: Use of Creative Commons licenses for game project files.

licenses.

A recent paper on videogame text corpora (which would not capture how that text was directly linked to player choices) offered desiderata for corpus quality (van Stegeren and Theune 2020). Among their criteria was a concern for *representativeness*, that the dataset represent the work of professional videogame writers and be sourced from well-known games that have a substantial user base. Our corpus instead strives for a sense of *authenticity* in the sense of having the dataset represent work intended as human experiences even if not made by professionals. Their concern for *diversity* suggested that the dataset reflect the variety of types of text occurring in videogames (e.g. dialog, tutorial, character names, etc.). By contrast, our concern for *diversity* considers the variety of subject matter, and a population of student authors that is more diverse than the population of professional game authors.

Example Application: Tag Prediction

To validate the utility of our corpus in accelerating machine perception research, this section sets up a small machine learning experiment. Consider the task of predicting notable features of the player experience for a game, given access to the project materials. Without compiling and executing the game (and extensively interacting with it), we may still be able to glean clues about the intended player experience from key words and phrases used in the source code. These clues might be in the raw text seen by players (perhaps a character's dialog include the line "I'm feeling depressed lately.") or in structural identifiers used in the game's scripting logic (such as a variable called "`$romanceLevel`" or a scene internally titled "Boss Battle"). Here, we train several simple neural networks on the task of predicting the

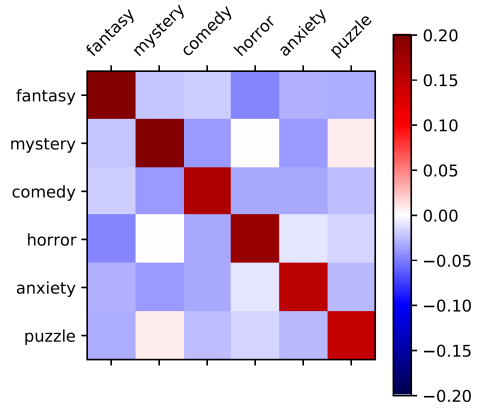

Figure 5: Covariance of ground truth labels used in tag prediction (a multi-task classification task).

student-author provided tags from the provided source code for Twine stories. Our general approach is depicted in Fig. 6.

Selecting among the most commonly used tags for Twine games (skipping "adventure" and "choices matter" because these two were given as examples in the form we used to prompt students for tags), we chose six labels to be used as targets in a multi-task classification problem: fantasy, mystery, comedy, horror, anxiety, and puzzle. Filtering the corpus to games associated with at least one of these tags and being authored in the (common) Harlowe format of Twine version 2 yielded 181 games. On this subset, many games had more than one tag within our selected set (1.3 on average). Figure 5 characterizes the covariance of the binary labels in this classification task.

These 207 games comprise a total of 11,822 passages. We preprocess the source code of each passage by extracting an approximation of all player-readable text. We do this by iteratively rewriting scripting macro calls. For example, the source text "He said '(if: $romanceLevel $>$ 5)[yes](else:)[no].'" is rewritten as "He said ' yes no .'" At the same time, we try to identify references between passages. This most often happens when one passage has a simple hyperlink to the other. To account for references that depend on scripting logic (such as when a link is only visible to players who have achieved a certain status tracked by game state variables), we look for mentions of passage names in the string literals of the scripting logic. This approximation recovers about 1.5 references per passage on average. Some passages (such as ending scenes) have no outgoing references, most commonly (as part of linear story segments) they have just one, and there is a long tail of higher reference counts owing to scenes where the player makes importance choices or even interacts with procedurally generated menus (as in the case of some game inventory systems). After sorting the list of games by title, we split the set in half: the first 100 games are used to train models and the remaining 107 are used for evaluation. This results in about 22 positive (whole-game) examples of each tag in each split.

Because the goal of this exercise is to demonstrate *use of the corpus* rather than contribute any specific new tech-

niques for machine perception of interactive media, we consider a restricted family of simple neural architectures in the models used in our experiment. In particular, all models digest the variable-sized text of passages by first applying the same pre-trained Universal Sentence Encoder (USE) (Cer et al. 2018). That is, we use a specific instance[10] of USE to preprocess passage texts into 512-dimensional vectors. USE is based on deep averaging networks (DANs), which can be seen as representing the unordered bags of words within each passage.

To reason about graph-structured data (where the meaning of a passage might be influenced by the meaning of passages that link to or are linked from it), we apply graph convolutional networks (GCNs) (Kipf and Welling 2017). Our implementation is based on the Spektral library (Grattarola and Alippi 2020). In our most sophisticated (but still very simple) model, the input passage vectors are passed through a Dense layer that reduces the dimensionality to 128 (with tanh activation). This reduced-dimensionality representation is then used for several rounds of message passing with a single recurrently-applied GraphConvSkip layer (also using tanh activation). This refined representation is concatenated with the original passage vectors, and the result is globally average pooled to produce a single 640-dimensional vector for the whole game. A distribution for each of the six separate labels is predicted with a single linear transformation (sigmoid activation), effectively implementing logistic regression on the learned game representations.

To understand the usefulness of reasoning about Twine games as a graph of linked passages, we consider sweeping the number of message passing rounds down to 0. This does not alter the number of free parameters in the model, only the number of recurrent iterations. We separately consider replacing the GCNConv layer with an equivalently shaped Dense layer to implement a traditional multi-layer perceptron (MLP) architecture at each node in the graph. We call this the MLP rather than Graph model. Finally, we also consider a Default model that predicts a label distribution without looking at any features of the game.

Following an overfit-then-regularize methodology, we ensured each (non-Default) model could achieve negligible error on the training set (ensuring each had enough capacity to distinguish all training games if needed and express the full label set). Then, we applied regularization to improve generalization to the test set. In particular, we applied label smoothing (0.3), and dropout (0.25) on the node vector representations before the global graph pooling operation.

Each model was trained for 500 epochs with the Adam optimizer using a constant learning rate of 0.01. Table 2 records the test performance of the various models at the end of training.

Observed improvements over the Default model indicates that there is some useful signal in the USE passage vectors (i.e. content words are somewhat predictive of tags). Improvement over the MLP model indicates that there is a benefit to working with passage linkage information, and scaling up the range of communication over those links results

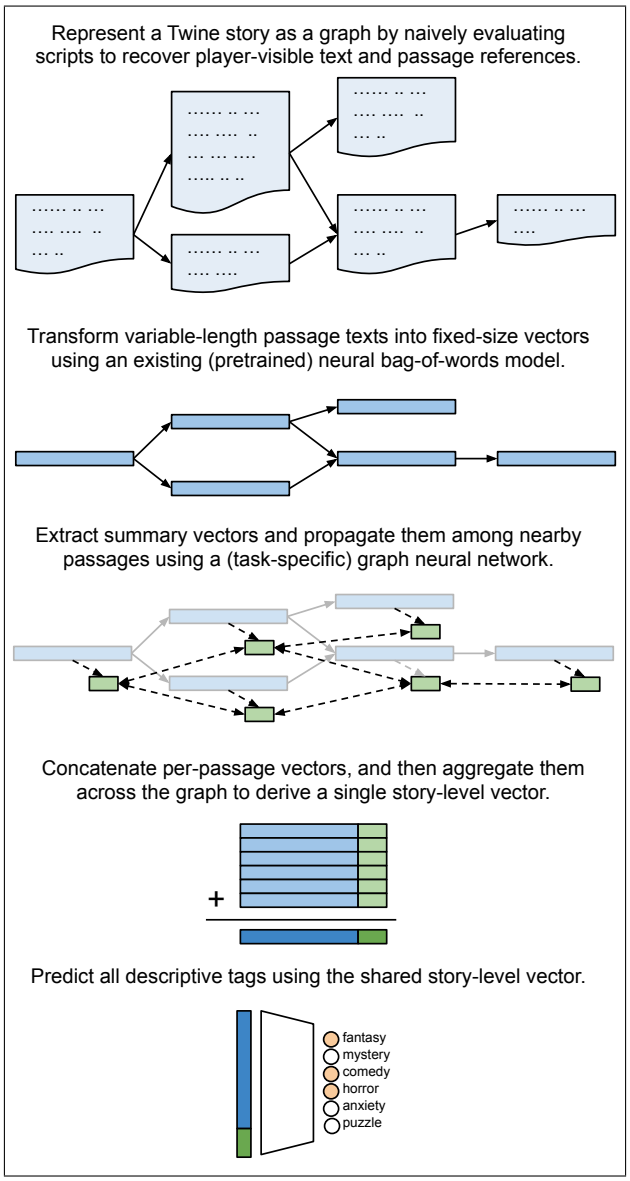

Figure 6: General architecture of our tag prediction models. The causal impact of interactive choices is roughly represented by the network of references between passages. We probe the utility of this information by comparing against models that scale down and even eliminate inter-passage communication during inference.

[10]https://tfhub.dev/google/universal-sentence-encoder/4

Model architecture	AUC-ROC
Default (no input features)	0.469
MLP	0.645
Graph0	0.647
Graph1	0.648
Graph2	0.669
Graph4	0.678
Graph8	0.681
Graph16	**0.688**

Table 2: Area under the receiver operating characteristic curve (a classifier performance metric) of trained tag prediction models evaluated on the test split (higher is better). All GraphK models have the same number of parameters, but they differ in the number of message passing rounds used to propagate information between linked passages (K). For example, in Graph4, a passage can influenced by another passage through a chain of up to 4 references.

in improved performance. If we think of the MLP model as reasoning about patterns of word co-occurence within each passage (averaged over the whole story), we can think of the Graph models as capturing patterns of word co-occrence between nearby linked passages. Though hardly interpretable as an operationalization of a theory like choice poetics, the Graph model can at least crudely represent the way player observations in one moment of gameplay contextualize the observations made in another that is reachable via certain choices from the first.

Future Research Directions

The availability of this new dataset prompts several new research directions. One very concrete possibility is to simply scale up the tag-prediction task examined above. Although we built our graph from the network of passages seen in the game's source code, the same graph might be interpreted as the state space graph realized by a semi-exhaustive game exploration algorithm. In this interpretation, our use of the GraphConvSkip operation is related to the use of LSTMs in other gameplaying systems: a way of summarizing what has been experienced so far in the current interaction episode. Future work might build more representative graphs during preprocessing. Alternatively, continuing to treat the graph as a representation of source code, further graph-inference techniques already used for code search (Xu et al. 2017) or program synthesis (Shin et al. 2019) might yield representations more predictive of player experience.

Rather than improving the predictive model, another line of future work might aim to improve the quality of the data. Such a project might, via crowdsourcing or other means, generate new textual labels (e.g. tags and descriptions) for the our given set of games with additional validation on the new labels. This pattern of upgrading the labels on an existing dataset was used in computer vision with ImageNet to reveal how past systems were overfitting to the data collection conditions of the original dataset (Beyer et al. 2020).

Because future research will want to draw observations from these games, it would be desirable to have GGP systems that could competently play (at least in terms of exploration) any of the games in the corpus. Because these games were not designed to be AI testbeds from the start (lending them authenticity), more work is needed to find ways of integrating them with existing GGP systems and techniques. Within our corpus, researchers may choose to work on just one type of game at a time, perhaps building a system to handle all of the Bitsy games before attempting any of the Construct or Godot games.

A recent organized collection of novice-created games (Aycock and Biittner 2020) was published with the explicit intention of supporting historical analysis. Both this collection and our own Undergraduate Games Corpus may also usefully support empirical methods in software engineering research. For example, consider Techapalokul and Tilevich's (2017) analysis of hundreds of student-created Scratch programs to understand software quality problems.

Finally, there is need for further research that will align new perceptual models with human requirements for practical AI systems. Anderson et al did a requirements analysis for search finding, amongst other things, a desire to search for moments using game-specific vocabulary (e.g. "Mario on Yoshi in an underwater level with 8 lives remaining") and for such queries to be able to match against videos of gameplay in the wild (where access to source code is not available) (Anderson and Smith 2019). Machine perception of interactive media should move beyond perceiving the optimal next move to predicting aspects of games and the moments in them that are relevant to applications like app store search, authoring assistance, or even lend tools usable in the digital humanities to support quantitative, distant readings of games (Iantorno 2020).

Conclusion

This paper contributes the Undergraduate Games Corpus, a new dataset intended to accelerate research on machine perception of interactive media. This game collection represents authentic design work exhibiting several dimensions of diversity: from varied student author identities and levels of design polish to widely varied genres, topics, and game mechanics. By providing groups of games organized by authoring tool, we offer a wide spectrum of technical complexity for future machine perception projects. Compared with many other datasets for computer vision and natural language processing which might repurpose use bulk data found online, every item in our corpus was explicitly contributed for use in technical games research.

Ethics Statement

As articulated above, our contribution of the Undergraduate Games Corpus has a number of potential societal benefits. It can accelerate the progress of AI research with potential for commercial, artistic, and scholarly applications. It can broaden the representation of games and stories considered by AI research from (researcher-created clones of) a narrow selection of games that have already achieved widespread commercial success to those representing the interests and

lives of diverse student authors. Further, it can anchor research on the types and scales of games being created by students in a way that makes it easier to translate techniques developed in research into educational applications (such as offering a way for a student to visualize their current game draft in relation to their peers' games with respect to structural and content features).

The provision of an organized dataset of student work is not without risks. Some student contributors may eventually come to regret having their name attached to their contribution. Public availability of this data means the games are subject to crawling even by current-generation search engines that would severely limit their ability to revoke this attribution later. This concern applies to any student work that is made available online, even outside of organized datasets.

References

Anderson, B. R.; and Smith, A. M. 2019. Understanding User Needs in Videogame Moment Retrieval. In *Proceedings of the 14th International Conference on the Foundations of Digital Games*, FDG '19.

Aycock, J.; and Biittner, K. 2020. LeGACy Code: Studying How (Amateur) Game Developers Used Graphic Adventure Creator. In *Proceedings of the 15th International Conference on the Foundations of Digital Games*.

Bakker, B. 2002. Reinforcement Learning with Long Short-Term Memory. In Dietterich, T. G.; Becker, S.; and Ghahramani, Z., eds., *Advances in Neural Information Processing Systems 14*, 1475–1482. MIT Press.

Beyer, L.; Hénaff, O. J.; Kolesnikov, A.; Zhai, X.; and van den Oord, A. 2020. Are we done with ImageNet? *arXiv preprint arXiv:2006.07159* .

Bogost, I. 2007. *Persuasive Games: the Expressive Power of Videogames*. MIT Press. ISBN 9780262026147.

Cer, D.; Yang, Y.; Kong, S.-y.; Hua, N.; Limtiaco, N.; John, R. S.; Constant, N.; Guajardo-Cespedes, M.; Yuan, S.; Tar, C.; et al. 2018. Universal sentence encoder. *arXiv preprint arXiv:1803.11175* .

Ebner, M.; Levine, J.; Lucas, S. M.; Schaul, T.; Thompson, T.; and Togelius, J. 2013. Towards a Video Game Description Language. In Lucas, S. M.; Mateas, M.; Preuss, M.; Spronck, P.; and Togelius, J., eds., *Artificial and Computational Intelligence in Games*, volume 6 of *Dagstuhl Follow-Ups*, 85–100. Dagstuhl, Germany: Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. ISBN 978-3-939897-62-0. doi:10.4230/DFU.Vol6.12191.85.

Ecoffet, A.; Huizinga, J.; Lehman, J.; Stanley, K. O.; and Clune, J. 2019. Go-explore: a new approach for hard-exploration problems. *arXiv preprint arXiv:1901.10995* .

Genesereth, M.; and Björnsson, Y. 2013. The international general game playing competition. *AI Magazine* 34(2): 107–107.

Grattarola, D.; and Alippi, C. 2020. Graph Neural Networks in TensorFlow and Keras with Spektral. *arXiv preprint arXiv:2006.12138* .

Harteveld, C.; Stahl, A.; Smith, G.; Talgar, C.; and Sutherland, S. C. 2016. Standing on the shoulders of citizens: Exploring gameful collaboration for creating social experiments. In *2016 49th Hawaii International Conference on System Sciences (HICSS)*, 74–83. IEEE.

Iantorno, M. 2020. GameSound, Quantitative Games Analysis, and the Digital Humanities. *Digital Studies/Le champ numérique* 10(1).

Kipf, T. N.; and Welling, M. 2017. Semi-Supervised Classification with Graph Convolutional Networks. In *International Conference on Learning Representations (ICLR)*.

Lefler, M.; and Mallett, J. 1998-2020. Zillions of Games – Unlimited Board Games & Puzzles. URL http://www.zillions-of-games.com/index.html. (accessed Sep. 09, 2020).

Leidner, J. L.; and Plachouras, V. 2017. Ethical by design: Ethics best practices for natural language processing. In *Proceedings of the First ACL Workshop on Ethics in Natural Language Processing*, 30–40.

Love, N.; Hinrichs, T.; Haley, D.; Schkufza, E.; and Genesereth, M. 2006. General Game Playing: Game Description Language Specification. Technical report, Stanford University. LG-2006-01 games.stanford.edu.

Martens, C.; Summerville, A.; Mateas, M.; Osborn, J.; Harmon, S.; Wardrip-Fruin, N.; and Jhala, A. 2016. Proceduralist readings, procedurally. In *Twelfth Artificial Intelligence and Interactive Digital Entertainment Conference, AIIDE'16*.

Mawhorter, P.; Zegura, C.; Gray, A.; Jhala, A.; Mateas, M.; and Wardrip-Fruin, N. 2018. Choice poetics by example. In *Arts*, volume 7(3), 47. Multidisciplinary Digital Publishing Institute.

Mikolov, T.; Yih, W.-t.; and Zweig, G. 2013. Linguistic regularities in continuous space word representations. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 746–751.

Morrow, A.; and Casucci, T. 2019. Preserving and Disseminating Emerging Forms of Digital Scholarship in Academic and Research Libraries: The EDS Report. Technical report, University of Utah.

Nelson, M. J. 2020. Institutions Active in Technical Games Research. URL https://www.kmjn.org/game-rankings/. (accessed Sep. 07, 2020).

Odusami, M.; Abayomi-Alli, O.; Misra, S.; Shobayo, O.; Damasevicius, R.; and Maskeliunas, R. 2018. Android malware detection: A survey. In *International Conference on Applied Informatics*, 255–266. Springer.

Partlan, N.; Carstensdottir, E.; Kleinman, E.; Snodgrass, S.; Harteveld, C.; Smith, G.; Matuk, C.; Sutherland, S. C.; and El-Nasr, M. S. 2019. Evaluation of an automatically-constructed graph-based representation for interactive narrative. In *Proceedings of the 14th International Conference on the Foundations of Digital Games*, FDG'19.

Partlan, N.; Carstensdottir, E.; Snodgrass, S.; Kleinman, E.; Smith, G.; Harteveld, C.; and El-Nasr, M. S. 2018. Exploratory Automated Analysis of Structural Features of Interactive Narrative. In *Fourteenth Artificial Intelligence and Interactive Digital Entertainment Conference (AIIDE'18)*, 88–94.

Pell, B. 1993. *Strategy generation and evaluation for meta-game playing*. Ph.D. thesis, University of Cambridge.

Pell, B. 1996. A strategic metagame player for general chess-like games. *Computational intelligence* 12(1): 177–198.

Perez-Liebana, D.; Samothrakis, S.; Togelius, J.; Lucas, S. M.; and Schaul, T. 2016. General Video Game AI: Competition, Challenges, and Opportunities. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, AAAI'16, 4335–4337. AAAI Press.

Prabhu, V. U.; and Birhane, A. 2020. Large image datasets: A pyrrhic win for computer vision? *arXiv preprint arXiv:2006.16923* .

Ryan, J.; Kaltman, E.; Fisher, A.; Owen-Milner, T.; Mateas, M.; and Wardrip, N. 2017. GameSpace: An Explorable Visualization of the Videogame Medium. Technical report, University of California, Santa Cruz, School of Engineering. doi:10.13140/RG.2.2.33425.94565.

Ryan, J.; Kaltman, E.; Hong, T.; Isbister, K.; Mateas, M.; and Wardrip-Fruin, N. 2016. GameNet and GameSage: Videogame Discovery as Design Insight. In *Proceedings of the First International Joint Conference of DiGRA and FDG*, DiGRA/FDG '16. Dundee, Scotland: Digital Games Research Association and Society for the Advancement of the Science of Digital Games. ISBN ISSN 2342-9666.

Shin, E. C.; Allamanis, M.; Brockschmidt, M.; and Polozov, A. 2019. Program synthesis and semantic parsing with learned code idioms. In *Advances in Neural Information Processing Systems*, 10825–10835.

Silver, D.; Schrittwieser, J.; Simonyan, K.; Antonoglou, I.; Huang, A.; Guez, A.; Hubert, T.; Baker, L.; Lai, M.; Bolton, A.; et al. 2017. Mastering the game of go without human knowledge. *Nature* 550(7676): 354–359.

Sloane, M.; Moss, E.; Awomolo, O.; and Forlano, L. 2020. Participation is not a Design Fix for Machine Learning. *arXiv preprint arXiv:2007.02423* .

van Stegeren, J.; and Theune, M. 2020. Fantastic Strings and Where to Find Them:The Quest for High-Quality Video Game Text Corpora. In *2020 Workshop in Intelligent Narrative Technologies (INT'20)*.

Techapalokul, P.; and Tilevich, E. 2017. Understanding recurring software quality problems of novice programmers. In *Proceedings of the 2017 IEEE Symposium on Visual Languages and Human-Centric Computing*, 43–51.

Thielscher, M. 2010. A general game description language for incomplete information games. In *AAAI*, volume 10, 994–999.

Treanor, M.; and Mateas, M. 2009. Newsgames-Procedural Rhetoric Meets Political Cartoons. In *In Proceedings of 2009 Conference of the Digital Games Research Association (DiGRA'09)*.

Treanor, M.; Schweizer, B.; Bogost, I.; and Mateas, M. 2011. Proceduralist Readings: How to Find Meaning in Games with Graphical Logics. In *Proceedings of the 6th International Conference on Foundations of Digital Games*, FDG '11, 115–122. New York, NY, USA: ACM. ISBN 978-1-4503-0804-5. doi:10.1145/2159365.2159381. URL http://doi.acm.org/10.1145/2159365.2159381.

Vinyals, O.; Babuschkin, I.; Chung, J.; Mathieu, M.; Jaderberg, M.; Czarnecki, W. M.; Dudzik, A.; Huang, A.; Georgiev, P.; Powell, R.; et al. 2019. Alphastar: Mastering the real-time strategy game starcraft ii. *DeepMind blog* 2.

Xu, X.; Liu, C.; Feng, Q.; Yin, H.; Song, L.; and Song, D. 2017. Neural network-based graph embedding for cross-platform binary code similarity detection. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, 363–376.

Zhan, Z.; Aytemiz, B.; and Smith, A. M. 2019. Taking the Scenic Route: Automatic Exploration for Videogames. In *Proceedings of the Second AAAI Knowledge Extraction from Games Workshop (KEG-19)*.

Zhan, Z.; and Smith, A. M. 2018. Retrieving Game States with Moment Vectors. In *Proceedings of the First AAAI Knowledge Extraction from Games Workshop (KEG-18)*.

Zhang, X.; and Smith, A. M. 2019. Retrieving videogame moments with natural language queries. In *Proceedings of the 14th International Conference on the Foundations of Digital Games*, FDG '19.

Zhang, X.; Zhan, Z.; Holtz, M.; and Smith, A. M. 2018. Crawling, Indexing, and Retrieving Moments in Videogames. In *Proceedings of the 13th International Conference on the Foundations of Digital Games*, FDG '18. New York, NY, USA: ACM. ISBN 978-1-4503-6571-0. doi: 10.1145/3235765.3235786.