Retrieving Videogame Moments with Natural Language Queries

Xiaoxuan Zhang
Department of Computational Media
University of California Santa Cruz
xzhan209@ucsc.edu

Adam M. Smith
Department of Computational Media
University of California Santa Cruz
amsmith@ucsc.edu

ABSTRACT

Search engines for books can usually tell us which specific pages in a book mention the concepts we seek. A similar ability to search within the contents of games, locating specific moments in their spaces of interactivity, is not yet available. This limits players' ability to find deeply relevant games and game scholars' ability to find moments that advance their arguments. Drawing on computer vision and natural language processing, our work introduces the ability to search within a space of game moments using natural language queries. We describe and evaluate a prototype system which is capable of retrieving moments from two contemporary, narrative-driven games by semantic matching on both the auditory and visual content of scenes.

CCS CONCEPTS

• **Information systems** → *Information retrieval*; • **Computing methodologies** → *Artificial intelligence*; • **Applied computing** → Computer games.

KEYWORDS

videogames, content-based retrieval, natural language processing, image recognition

ACM Reference Format:
Xiaoxuan Zhang and Adam M. Smith. 2019. Retrieving Videogame Moments with Natural Language Queries. In *The Fourteenth International Conference on the Foundations of Digital Games (FDG '19), August 26–30, 2019, San Luis Obispo, CA, USA.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3337722.3341867

1 INTRODUCTION

The *content* of a videogame is conveyed through the visual, auditory, and other perceptual details attached to moments reachable through gameplay. Unfortunately, this is not the information considered by current search engines. Game search engines, such as those that power online game markets (e.g. Google Play[1]) or review sites (e.g. Metacritic[2]), emphasize developer-provided *paratext* [7] such as title and description or audience-provided *metadata* such as ratings, comments, and play-counts. Our work aims to allow directly searching the interactive content of games.

Consumers, scholars, and other stakeholders in gameplay experiences often resort to generic web search tools to find fragments of gameplay in videos, images, or blogs that offer indirect representations of the game content. This works to find information about the most-discussed moments of already-popular games, but it only offers coverage for the tiny fraction of available experiences that have managed to bleed into the textual media understood by web search engines. Recently, we introduced the problem of crawling, indexing, and retrieving moments in videogames [21] towards addressing the problem of discoverability for game contents.

Unlike non-interactive media, a videogame can provide a potentially infinite space of content that can only be unpacked through gameplay. This large space increases the difficulty of information retrieval tasks (compared to, say, text, image, or video retrieval), inasmuch as a system competent for this task should have exposure to all of the relevant content in the first place. The problem of automatically exploring a videogame with the intent to discover the most significant moments (the kind the user might seek in a search engine) has only recently been investigated [19].

Even with perfect access to the content experienceable in every possible playthrough of a game, a major question remains: *how should users identify the content they are seeking?* Unlike books, most videogames lack an obvious notion of keywords which might be used as the basis for matching moments with queries. Initial results in content-based retrieval for videogames (mentioned above) offered visual search capabilities: ranking the known moments of a game by their semantic similarity to one or more user-supplied *screenshots*. Our work suggests the use of *natural language* descriptions of game content, allowing users to search for moments for which they do not already have sample imagery.

We show how to combine ideas from computer vision and natural language processing to represent game moments in a way that makes them discoverable by descriptions of their visual and spoken content. We evaluate results from our prototype system using metrics from the information retrieval literature. Additionally, we provide a 3D information visualization where the user can view how moments are related according to their textual and visual features, and find interesting moments through immediate interactions. Using this system, we can ask questions like where in the game the character talked about a particular topic, what kind of environment the game is set in, or where a particular scene is present in this game. We further show that the system makes use of relatively deep semantic knowledge to judge similarity when the exact vocabulary for describing a moment is not available.

[1] https://play.google.com/store/apps/category/GAME
[2] https://www.metacritic.com/game

FDG '19, August 26–30, 2019, San Luis Obispo, CA, USA
© 2019 Association for Computing Machinery.
ACM ISBN 978-1-4503-7217-6/19/08…$15.00
https://doi.org/10.1145/3337722.3341867

2 BACKGROUND

Despite the wealth of literature on information retrieval [12], the building blocks that make videogame moment search using natural language queries possible have only recently emerged.

2.1 Content-based Videogame Retrieval

The *GameSpace* project [15] offers an iconic example of a common, content-agnostic, strategy for organizing games. *GameSpace* takes the form of an interactive 3D visualization of 15,000 games in which games are spatially organized according to an analysis of the text of Wikipedia articles written *about* them; games that are described in similar terms are placed closer together. While this project is one of our direct inspirations, it fundamentally does not look into the interior of games where a similarly elaborate constellations of distinct moments might be found.

Our introduction of the challenges and opportunities for videogame moment search [21] highlighted a different kind of game space: the space of moments contained within individual games (analogous to the many pages or paragraphs in a book). Rather than relying on paratext or metadata, this project organized moments by perceptual similarity using the Pix2Mem [20] embedding strategy (which applies deep learning to collections of screenshot and memory-state pairings extracted from sample gameplay traces). We described a prototype search engine which accepts one or more sample screenshots as queries and visualizes results (moments also represented by a screenshot) in the ranked-list format of traditional search engines. This prototype search engine demonstrated content-based retrieval, but it required users to already have access to some of the content they were seeking.

Following up on this work, Anderson and Smith [1] conducted a semi-structured interview study with various stakeholders in videogame moment retrieval to understand user needs: what kind of moments do various users seek; how do they want to identify those moments in a search; and what will they do with retrieved moments? Among other outcomes (such as the need to cover contemporary games), this study highlighted a broad desire for future systems to support text search, either using general-purpose natural language or with highly domain-specific vocabularies. One subject offered the example query (in reference to the game *Super Mario World*): "Mario on Yoshi in an underwater level with 8 lives remaining." For a system to judge the relevance of a moment to this query, the system needs to be able to visually identify specific characters and their relationships (e.g. Mario on Yoshi) as well as understand general descriptions of situations for which on-screen text never offers specific clues (e.g. in an underwater level).

2.2 Cross-Modality Retrieval

In traditional information retrieval domains, textual queries are used to retrieve textual documents [12]. In image search, image queries are used to retrieve documents that are themselves images [3]. In the emerging field of cross-modality retrieval, one kind of query is used to retrieve documents of a very different type [18]. A very common use of cross-modality retrieval is to allow non-text items to be retrieved using textual queries.

Whether it is using text to retrieve images [9], music [8], or even code fragments [5], a common strategy is to map both the space of documents of interest and the space of possible text queries into a common embedding space where distances are meaningful; a document is embedded close to the embedding location of queries that are intended to retrieve that document. Locations in space are typically represented as vectors, and the function that embeds queries and documents into that vector space can take the form of a deep neural network (as in Pix2Mem mentioned above).

In the context of natural language processing, vector representations of words, sentences, and paragraphs find many uses beyond retrieval. A common approach for representing text in a vector space is word2vec [13], where words that appear in similar contexts are embedded to similar locations in, for example, a 300-dimensional space. The fastText library [2] offers a selection of freely downloadable embedding models that are trained on large number of Wikipedia articles in many languages. Different from other techniques for learning word representations, it treats each word as a bag of characters based on an n-gram model instead of learning each word as a distinct unit. This allows it to process not only common words but also subwords and misspelled words, making it possible to draw useful inferences for rare or even out-of-vocabulary words (those not included in training data). It could serve as a baseline for many potential downstream tasks such as text classification and sentiment analysis. The resulting vectors can be used to generate good sentence or phrase representations, making it a good fit for representing the descriptive words and phrases we expect to see in content-based videogame retrieval.

Towards visual search, deep convolutional neural networks have been widely used to solve computer vision problems such as image classification and recognition. For example, the Google Inception-v3 model [16] is among the most high-performance network architectures for image recognition. While not specifically aiming at perception of game content (instead trained on a collection of real-world photographic imagery [4]), this kind of technique can be useful for extracting visual features of contemporary games which often make use of photo-realistic graphics that intend to represent real-world objects. Such a system can express whether a given scene appears depict an object from a large collection of categories. When the vision system cannot recognize the precisely needed category of objects, it may still recognize objects that are linguistically relatable to the category of interest. Luo et al. [10] used a similar strategy of repurposing computer vision models originally designed for analyzing photographic data to analyze photorealistic videogame screenshots.

Later, we describe how we combine vector representations extracted by fastText and Inception-v3 into a common space so that cross-modal search for videogame moments can be achieved.

3 SYSTEM OVERVIEW

Figure 1 briefly illustrates the flow of data in our prototype system. In reference to *GameSpace*, our interactive system is called *Videogame Moment Space*.

From the *observation* of a gameplay trace, we select two features, audio and visual, to represent the player experience. Noting how a player *acts* would be also useful, for example in distinguishing moments of core gameplay from non-interactive cut-scenes, but we do not consider it in our initial steps in this research direction.

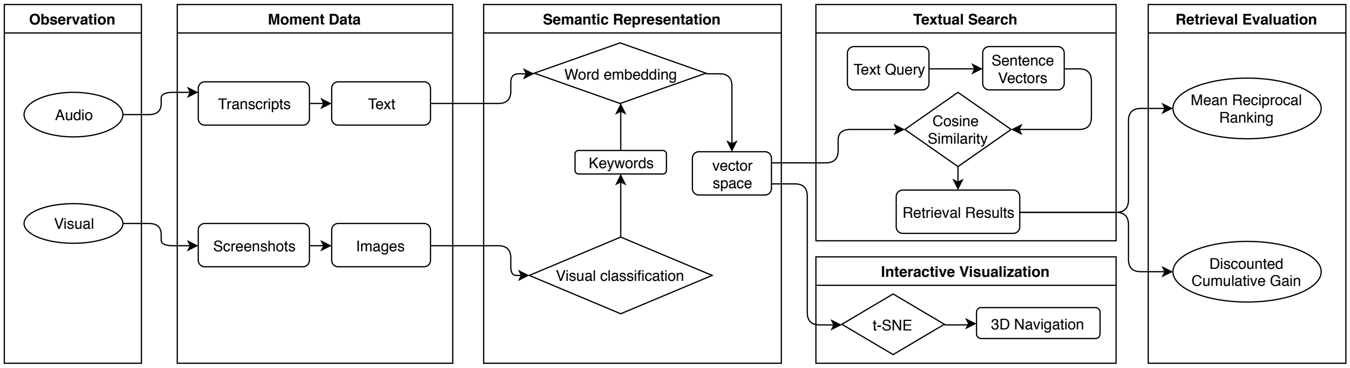


Figure 1: Data flow and transformation for *Videogame Moment Space*

In our work, *moment data* is represented at the level of individual frames of animation, based on screenshot images and a textual transcript (which functions as a summary of the spoken words). While this strategy misses out on interesting non-linguistic features of game audio (such as the mood of background music or the objects referred to by the sound of gunshots or dogs barking), it is enough to get started.

Semantic representation maps visual and textual materials into a common representation in a continuous vector space. These are *moment vectors* [20]. Our system deals with contemporary games (often with photorealistic graphics), in which keywords describing objects can be extracted from screenshots using the Inception computer vision model [16] and then merged with transcript text. The combined text is turned into the final vector representation by using a pre-trained fastText natural language processing model [2]. This vector space is our basis for cross-modal retrieval. While summarizing the visual content of a scene by just a few textual keywords loses a great deal of information, we experimentally show later that enough is preserved to find relevant moments. Many games visually portray additional text such as on signs in the fictional world or as instructions for the player (e.g. "press [x] to open door").

Textual search responds to some of the problems highlighted by Anderson and Smith [1], applying already existing models designed for natural language processing and computer vision tasks to the problem of content-based videogame retrieval. For textual retrieval, a user's textual query is first embedded into a vector space using the same fastText model, and its nearest neighbors in a corpus of game moments are retrieved as those with the highest cosine similarity.

Interactive visualization presents a 3D space, reminiscent of *GameSpace*, where a user can explore interesting videogame moments via various interactions like search, navigation, bookmarking and timeline browsing. This is made possible by mapping the high-dimensional moment vectors into three-dimensional space using the *t-SNE* algorithm [11].

Retrieval evaluation considers the quality of retrieval results according to two metrics from the information retrieval literature: mean reciprocal ranking (MRR) [17] and normalized discounted cumulative gain (nDCG) [12], which are commonly used to evaluate web search engines.

4 TECHNICAL DESIGN

We selected two games for investigation in our prototype: *Life Is Strange* [6] and *The Last Of Us* [14]. Both of which are acclaimed especially for their rich narratives, conveyed through audio and visual channels.

4.1 Data Collection

Our moment data came from gameplay videos[3] on YouTube. For the sake of simplicity, we used the videos with automatic captioning enabled such that the transcripts could be downloaded using a third-party tool.

Screenshots were captured from gameplay videos with a variable interval (approximately one screenshot per second) so as to keep the dataset within a reasonable size without missing out on valuable information. The interval was adjusted automatically according to the intensity of the dialog during gameplay to keep as much information as possible from the narratives. In order to keep only the high-quality screenshots, images with easily-detectable problems like showing a completely blank screen or excessive motion-blur were discarded using simple computer vision techniques.

4.2 Vector Representation

Following our previous work [21], our system operates in the vector space retrieval model [12, Chap. 6]. To obtain moment vectors, we used a model pre-trained on Wikipedia[4] by fastText, for its efficiency and accuracy, as well as its capability of dealing with out-of-vocabulary words [2]. First, the caption text in each moment was cleaned up, filtering out the punctuation, stop words, numbers and so forth. The text is then tokenized into individual words that are later fed into the fastText model to generate word vectors. If a word did not exist in the model's vocabulary, a similar word inferred by the model using subword information was used instead. All of the words associated with a given moment (including those keywords suggested by the computer vision system) were combined by simple averaging. While our pre-processing and embedding steps are very English-specific, alternative vectorization strategies

[3] https://www.youtube.com/watch?v=AP5UBhyjMKA&list=PLuqolaDjqVv2VHWUrVuTLgT97oQnNy7we
https://www.youtube.com/watch?v=nGQM0yzg2Jk
[4] https://fasttext.cc/docs/en/pretrained-vectors.html

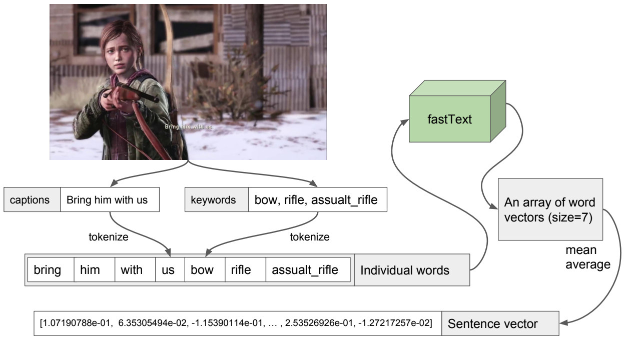


Figure 2: Moment vector representation process.

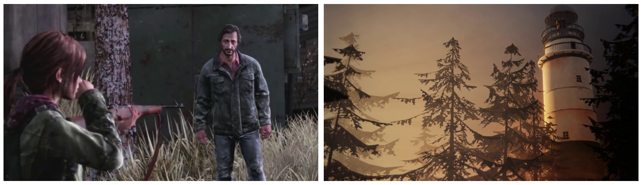


Figure 3: Extracted visual keywords for sample moments with no dialog. Left: `bow`, `rifle` and `assault rifle` (from *The Last Of Us*). Right: `beacon` (from *Life Is Strange*)

adapted to different natural languages or embedding methods could be applied here without adjusting the overall system architecture.

Because the meaning of any specific moment is influenced by the meaning of moments nearby in gameplay, we aggregate (or smooth) vector representations in time using a moving average filter. While this method is much less sophisticated than alternatives that consider the precise order and timing of audio and visual details, it works well enough to experimentally show a benefit.

Clearly, captions are not sufficient to represent all significant information about a moment, such as information about those moments that lack captions (see Figure 3). To augment captions, we apply the Google Inception-v3 [16] network to produce a list of possible object classifications for each screenshot. The names of these objects can be combined with the caption text as additional (visual) keywords. From the categories recognized by the Inception model, we considered a category detected in a screenshot when it was one of the top-three categories and the model reported a confidence value above 0.01.

After combining audio and visual details into text (via the process summarized in Figure 2), our moment vector space had a dimensionality of 300.

4.3 Textual Retrieval

With all the moment data embedded into a common vector space, retrieving moments within a corpus now becomes a problem of finding the nearest neighbours to a query document, in other words, calculating a distance between each candidate moment and the query. A query, which is a short sentence or some short phrases describing expected moments, is first mapped into a vector following the method previously described. Then, all vectors in the target corpus are ranked according to their cosine similarity to the given query through a linear scan. Once all the moment vectors are ranked, the top N neighbors (default $N = 20$) are the system's estimation of the most relevant answers to the given query. The corresponding moment data (screenshots, transcripts, etc.) are returned and shown via the visualization interface.

Figure 4 shows search results for the query "horses" in *The Last of Us*. The system successfully returns moments with horses even though the caption text of these moments do not mention horse-related terms and the vision system does not have an explicit "horse" category. The system has visually identified the related keywords `horse cart`, `Arabian camel`, and `ox`. A similar pattern is seen in Figure 5 where results for "sunset at the beach" in *Life Is Strange* show the system has reasoned from the query term "beach" to the visual keywords of `seashore`, `sandbar`, and `lakeside`.

As expected, the system can also match specific dialog text. In Figure 6, the query "selfie" yields results from a relevant scene in *Life Is Strange* where selfies are discussed in the dialog text despite the presence of distracting visual keywords like `refrigerator`, `sliding door`, or `projector` for the same moments.

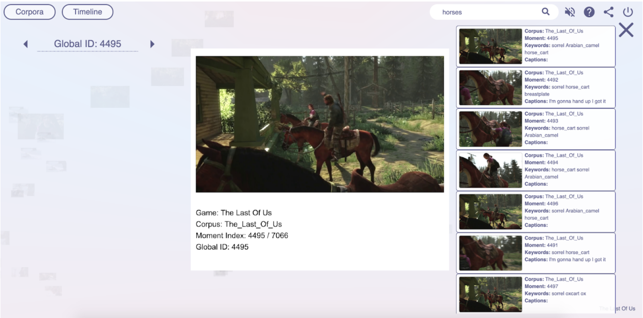
Although the system can reason through visual and linguistic similarity (synonymy and polysemy) for well-known concepts that were represented in the training data for the two vector embedding models, the system lacks knowledge of game-specific terminology. Figure 7 shows results for "zombie" in *The Last of Us*. In the fictional universe of this game, "clicker" is one of the terms for the zombie-like entities which drive action in the story. Because zombies/clickers are effectively unknown to the computer vision system and natural language processing subsystems, most of the highly-ranked results are not strongly related to the query. The fact that the top-most result is a both a perfect match for the query seems to be a fluke.

5 EVALUATION

While the previously referenced figures qualitatively illustrate the behavior of our system, we are interested in quantitatively evaluating its performance. This section considers a variety of different moment vector representations and their usefulness for matching the human judgment behind a collection of example queries with a small manually-curated expected results set. Experimental results are summarized in Figure 8.

5.1 Experiment Configurations

To understand the impact of various design choices in our moment vector representation, we experimentally disabled certain components of the system. First, considering that inaccurate predictions of the computer vision components of our system may introduce noise into to the vector representations, we try leaving out image data. Second, to examine how much caption text alone contributes to the retrieval performance, other experimental configurations disable the use of caption text. Third, because moments are captured frame-by-frame in time order from gameplay, a single moment only covers a small portion of a larger topic. We imagine that including the longer-term trends of a game's narratives would be



Game: The Last Of Us
Corpus: The_Last_Of_Us
Moment Index: 4495 / 7066
Global ID: 4495

Corpus: The_Last_Of_Us
Moment: 4495
Keywords: sorrel Arabian_camel horse_cart
Captions:

Corpus: The_Last_Of_Us
Moment: 4492
Keywords: sorrel horse_cart breastplate
Captions: I'm gonna hand up I got it

Corpus: The_Last_Of_Us
Moment: 4493
Keywords: horse_cart sorrel Arabian_camel
Captions:

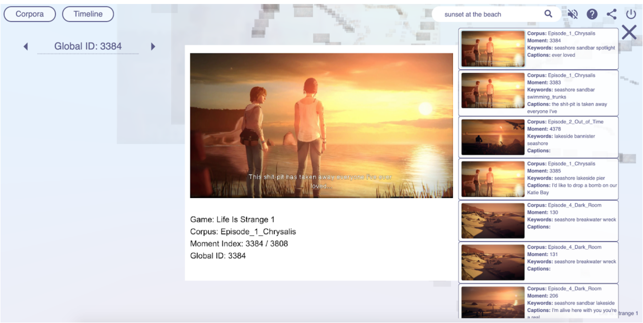
Corpus: The_Last_Of_Us
Moment: 4494
Keywords: horse_cart sorrel Arabian_camel
Captions:

Corpus: The_Last_Of_Us
Moment: 4496
Keywords: sorrel Arabian_camel horse_cart
Captions:

Corpus: The_Last_Of_Us
Moment: 4491
Keywords: sorrel horse_cart
Captions: I'm gonna hand up I got it

Corpus: The_Last_Of_Us
Moment: 4497
Keywords: sorrel oxcart ox
Captions:

Figure 4: Query "horses" in *The Last Of Us*.



This shit-pit has taken away everyone I've ever loved...

Game: Life Is Strange 1
Corpus: Episode_1_Chrysalis
Moment Index: 3384 / 3808
Global ID: 3384

Corpus: Episode_1_Chrysalis
Moment: 3384
Keywords: seashore sandbar spotlight
Captions: ever loved

Corpus: Episode_1_Chrysalis
Moment: 3383
Keywords: seashore sandbar swimming_trunks
Captions: the shit-pit is taken away everyone I've

Corpus: Episode_2_Out_of_Time
Moment: 4378
Keywords: lakeside bannister seashore
Captions:

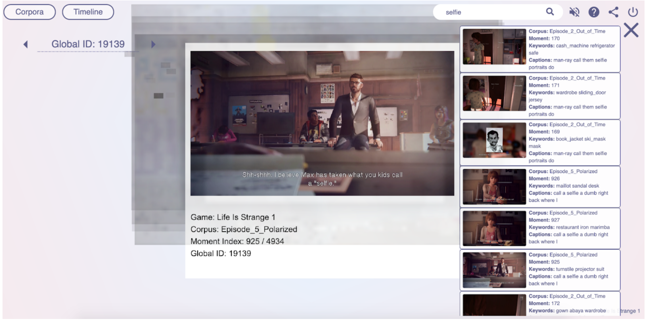
Corpus: Episode_1_Chrysalis
Moment: 3385
Keywords: seashore lakeside pier
Captions: I'd like to drop a bomb on our Katie Bay

Corpus: Episode_4_Dark_Room
Moment: 130
Keywords: seashore breakwater wreck
Captions:

Corpus: Episode_4_Dark_Room
Moment: 131
Keywords: seashore breakwater wreck
Captions:

Corpus: Episode_4_Dark_Room
Moment: 206
Keywords: seashore sandbar lakeside
Captions: I'm alive here with you you're a real

Figure 5: Query "sunset at the beach" in *Life Is Strange*.



Shh-shhh. I believe Max has taken what you kids call a "selfie."

Game: Life Is Strange 1
Corpus: Episode_5_Polarized
Moment Index: 925 / 4934
Global ID: 19139

Corpus: Episode_2_Out_of_Time
Moment: 170
Keywords: cash_machine refrigerator safe
Captions: man-ray call them selfie portraits do

Corpus: Episode_2_Out_of_Time
Moment: 171
Keywords: wardrobe sliding_door jersey
Captions: man-ray call them selfie portraits do

Corpus: Episode_2_Out_of_Time
Moment: 169
Keywords: book_jacket ski_mask mask
Captions: man-ray call them selfie portraits do

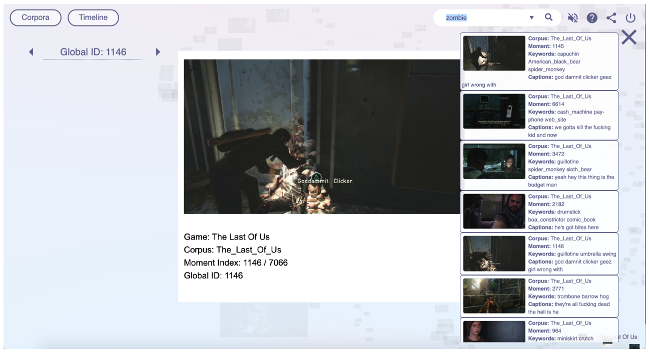
Corpus: Episode_5_Polarized
Moment: 926
Keywords: maillot sandal desk
Captions: call a selfie a dumb right back where I

Corpus: Episode_5_Polarized
Moment: 927
Keywords: restaurant iron marimba
Captions: call a selfie a dumb right back where I

Corpus: Episode_5_Polarized
Moment: 925
Keywords: turnstile projector suit
Captions: call a selfie a dumb right back where I

Corpus: Episode_2_Out_of_Time
Moment: 172
Keywords: gown abaya wardrobe

Figure 6: Query "selfie" in *Life Is Strange*.



Goddamnit. Clicker.

Game: The Last Of Us
Corpus: The_Last_Of_Us
Moment Index: 1146 / 7066
Global ID: 1146

Corpus: The_Last_Of_Us
Moment: 1145
Keywords: capuchin American_black_bear spider_monkey
Captions: god damnit clicker geez girl wrong with

Corpus: The_Last_Of_Us
Moment: 6614
Keywords: cash_machine pay-phone web_site
Captions: we gotta kill the fucking kid and now

Corpus: The_Last_Of_Us
Moment: 3472
Keywords: guillotine spider_monkey sloth_bear
Captions: yeah hey this thing is the budget man

Corpus: The_Last_Of_Us
Moment: 2182
Keywords: drumstick boa_constrictor comic_book
Captions: he's got bites here

Corpus: The_Last_Of_Us
Moment: 1146
Keywords: guillotine umbrella swing
Captions: god damnit clicker geez girl wrong with

Corpus: The_Last_Of_Us
Moment: 2771
Keywords: trombone barrow hog
Captions: they're all fucking dead the hell is he

Corpus: The_Last_Of_Us
Moment: 964
Keywords: miniskirt crutch

Figure 7: Query "zombie" in *The Last Of Us*.

more beneficial for finding the relevance among moments than the fragmented meaning of individual moments. To test this, we used different moving average sizes as well as disabling the smoothing altogether. Finally, we use results given by random sorting as a baseline to represent the default level of difficulty presented by our evaluation data set.

5.2 Ground Truth

These experiments require a set of queries and answers as the ground truth on which to operate. Whether an answer is relevant to its query is usually subject to human judgment. One might find multiple moments relevant to one given query, or multiple queries referencing one moment. Therefore, the query–answer data used in this evaluation were provided by several players who were familiar with two games we consider. They first came up with a topic described in a few words (based on their memory of significant moments of the game), and then they listed one or more contiguous snippets from the gameplay video as the ranges of correct answers for the query. The final query-answer dataset includes 27 entries for *Life Is Strange* and 14 entries for *The Last Of Us*. This data records only whether the human expert judges a moment as relevant or not, not indicating a single best answer or a graded notion of relevance.

5.3 Metrics

Information retrieval systems are typically evaluated by examining how the results they produce on a reference set of queries compare with a set of expected answers. A retrieval quality score of 1.0, on one of the metrics below, indicates perfect agreement with the ground truth data. A system that offers random results will usually get a score above 0.0, however, as relevant results can be highly ranked by chance if many results are marked as relevant in the ground truth data.

Mean Reciprocal Rank (MRR) [17] is a metric for statistically evaluating the quality of possible responses (ordered by cosine distance in our case) to a sample query. The reciprocal rank of a query response is the reciprocal of the rank of the first correct answer. For a set of queries, the mean value of each reciprocal rank is calculated. The best scenario for our retrieval task is when an answer to a query is returned as rank 1, which will have a score of 1 according this metric. So a higher score means better results. However, since only the first correct answer is concerned in this metric, it does not apply to cases where multiple relevant answers are considered as acceptable. For instance, MRR score would not be different between these two scenarios: a query has the correct answer returned as the first rank while the rest of the answers are all irrelevant, and a query has the correct answer returned as the first rank while the rest of the answers are also relevant. Users of a search engine would likely have a preference for the second case.

Normalized Discounted Cumulative Gain (nDCG) [12, Chap. 8] is a metric for ranking quality often used in web search engines. It assumes that a relevant result is more useful than less relevant or irrelevant ones, and a relevant result is more useful when it appears earlier in the result list. A gain value is cumulatively calculated from the top result to the bottom given an expected rank position. The lower ranks would be discounted from the gain as a kind of penalty. The normalized DCG is calculated by DCG/IDCG, where

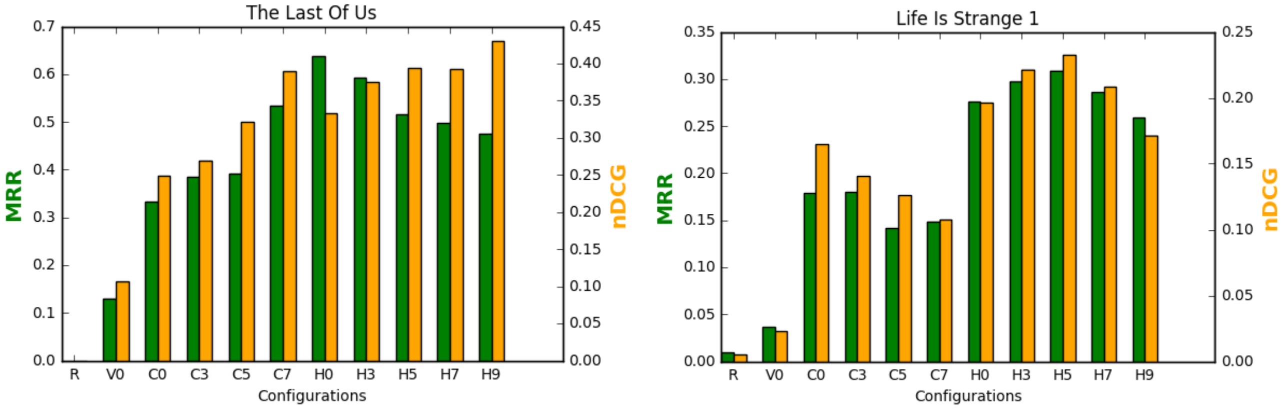


Figure 8: Evaluation of retrieval quality, using mean reciprocal rank (MRR) and normalized discounted cumulative gain (nDCG) metrics, on a player-created set of query-and-result-set pairs for two narrative-driven games. Different moment vector representations are compared: a random baseline (R), embeddings of only visual keywords (V0), embeddings of only caption keywords (C0, C3, C5, C7), and hybrid representations combining visual keywords and caption data (H0, H3, H5, H7, H9). Numerical tags indicates the width of the smoothing filter, e.g. vectors in H5 combine data from 5 additional adjacent moments.

IDCG is the ideal case of DCG (all relevant results are returned and ranked correctly). The overall retrieval quality of multiple queries can then be evaluated using the average score of normalized DCG. Typically, nDCG is evaluated only for the first few (k) results a user might see on a result page, nDCG@k. We evaluate nDCG@20.

5.4 Results

The evaluation scores in Figure 8 show several results. First, the low score for random guessing attests to the high difficulty of the retrieval task implied by our evaluation dataset. Next, retrieval using the semantic vector representations is always more effective than random sorting. Retrieval using the caption-based models is always more effective than the vision-only model, but we think this mostly illustrates the shallowness of our visual keyword extraction system. Combining visual and caption data is always better than using each data source in isolation. These results also show that even our simple moving average aggregation strategy improves retrieval performance, but the ideal amount of aggregation depends on the target game.

Modest disagreement between results for the MRR and nDCG metric cautions us not to over-interpret the relative performance of different vector representations in this experiment. Overall, the fact that scores never approach 1.0 on either metric for either game suggest there is ample room for improvement: both the natural language queries and specific game moments they reference could be much better understood. The scores for all techniques on one game should not be directly compared to the others because a different (game-specific) dataset was used in the evaluation for each game. Different senses of relevance and levels of precision were used by the different players consulted in the construction of each dataset.

6 VISUALIZATION

We implemented *Videogame Moment Space*, a web application with a 3D interactive visualization reminiscent of *GameSpace*. This visualization illustrates the relationships of our moment vectors and provides a convenient interface for searching for moments using natural language queries. The high dimensional moment vectors are mapped into two or three dimensional space using *t-SNE* [11]. Thumbnails of the screenshots are shown at the positions in the 3D space using the coordinates generated by t-SNE.

The user can use the text input box in the upper right corner of the interface to search for moments by short descriptions. The retrieved results will show up in a collapsible window with detailed information like screenshots, image keywords, and captions (as shown in Figures 4, 5, and 6). By clicking on a specific result, the user can view it in relation to other moments nearby.

7 CONCLUSION

This work shows how a new kind of cross-modality search engine can help users directly discover the content contained within videogames. It addresses a limitation of previous work (which only supported image-based queries) that was also highlighted for investigation by a recent user needs analysis [1]; many users would like to use textual descriptions of content to find moments of interest. We showed how off-the-shelf perceptual systems from computer vision and natural language processing can be combined with ideas from information retrieval to yield a system that makes useful judgments about the relevance of game moments to textual queries even when those judgments involve combining both visual and linguistic senses of similarity.

Content-based retrieval systems can play an important role in making a wider variety of games discoverable, particularly those that are not (or are not yet) popular enough to be documented in the media formats understood by traditional search engines.

REFERENCES

[1] Barrett Anderson and Adam M Smith. 2019. Understanding User Needs In Videogame Moment Retrieval. In *Proceedings of the 14th International Conference on the Foundations of Digital Games (FDG '19)*.

[2] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching Word Vectors with Subword Information. *Transactions of the Association for Computational Linguistics* 5 (2017), 135–146.

[3] Alberto Del Bimbo. 1999. *Visual information retrieval.* Morgan and Kaufmann, San Francisco, CA. http://cds.cern.ch/record/402484

[4] Jia Deng, Wei Dong, Richard Socher, Lia-Jia. Li, Kai Li, and Li Fei-Fei. 2009. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*.

[5] Github Engineering. 2018. Towards Natural Language Semantic Code Search - The Github Blog. https://github.blog/2018-09-18-towards-natural-language-semantic-code-search/

[6] Dontnod Entertainment. 2015. Life Is Strange. [Windows PC].

[7] Gérard Genette. 1997. *Paratexts: Thresholds of interpretation.* Vol. 20. Cambridge University Press.

[8] Peter Knees, Tim Pohle, Markus Schedl, and Gerhard Widmer. 2007. A music search engine built upon audio-based and web-based similarity measures. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval.* ACM, 447–454.

[9] Marco La Cascia, Saratendu Sethi, and Stan Sclaroff. 1998. Combining textual and visual cues for content-based image retrieval on the world wide web. In *Proceedings. IEEE Workshop on Content-Based Access of Image and Video Libraries (Cat. No. 98EX173)*. IEEE, 24–28.

[10] Zijin Luo, Matthew Guzdial, Nicholas Liao, and Mark Riedl. 2018. Player Experience Extraction from Gameplay Video. In *Fourteenth Artificial Intelligence and Interactive Digital Entertainment Conference.*

[11] Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *Journal of machine learning research* 9, Nov (2008), 2579–2605.

[12] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. 2008. *Introduction to Information Retrieval.* Cambridge University Press, New York, NY, USA.

[13] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781* (2013).

[14] NaughtyDog. 2013. The Last Of Us. [PlayStation 3].

[15] 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. *Proc. Foundations of Digital Games* (2015).

[16] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. 2016. Rethinking the Inception Architecture for Computer Vision. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR).*

[17] Ellen M Voorhees et al. 1999. The TREC-8 question answering track report. In *Trec*, Vol. 99. Citeseer, 77–82.

[18] Kaiye Wang, Qiyue Yin, Wei Wang, Shu Wu, and Liang Wang. 2016. A Comprehensive Survey on Cross-modal Retrieval. *CoRR* abs/1607.06215 (2016). arXiv:1607.06215 http://arxiv.org/abs/1607.06215

[19] Zeping Zhan, Batu Aytemiz, and Adam M Smith. 2018. Taking the scenic route: Automatic exploration for videogames. *Proceedings of the 2nd Workshop on Knowledge Extraction from Games co-located with 33rd AAAI Conference on Artificial Intelligence (AAAI 2019)* (2018).

[20] Zeping Zhan and Adam M Smith. 2018. Retrieving game states with moment vectors. In *Proceedings of the Workshop on Knowledge Extraction from Games co-located with 32rd AAAI Conference on Artificial Intelligence (AAAI 2018).*

[21] Xiaoxuan Zhang, Zeping Zhan, Misha Holtz, and Adam M. Smith. 2018. Crawling, Indexing, and Retrieving Moments in Videogames. In *Proceedings of the 13th International Conference on the Foundations of Digital Games (FDG '18).* ACM, New York, NY, USA, Article 16, 10 pages. https://doi.org/10.1145/3235765.3235786