

Introduction

Our project is an effort to create an interactive web application that allows users to search and navigate through video game data and information easily. The application is designed to both allow a seamless and accessible way of searching through a comprehensive database of video game information collected from MobyGames and recommend popular queries to users based on their interactions with the website. We have collected 1026 documents from the MobyGames site that contain information such as game descriptions, genres, release dates, and other metadata about the individual game. Through this website, we provide a better way for both consumers and publishers of the video game industry to understand the ever-growing scale of video games by organizing and presenting the data in a user-friendly way.

The application aims to provide a way for information seekers or video game enthusiasts to explore visualizations and rankings of our data so that they can quickly grasp trends or discover new and interesting games that they may not have played before. We provide an advanced search through our indexing of the collection, and also a popularity based recommendation system based on user interaction with our site to provide more insightful and exploratory results that users can use to expand their scope of search. The recommender system enhances the user experience by analyzing interactions such as search queries and the time of search to provide popular suggestions. By understanding user behavior and preferences, it delivers interesting recommendations that help users discover games they might not have encountered otherwise, creating a dynamic and engaging way to explore the database. Ultimately, our platform seeks to inspire discovery and foster a deeper appreciation for the diverse world of video games. By combining robust data with intuitive design, we aim to create an engaging and invaluable tool for gamers, enthusiasts, and industry professionals alike.

Design Details

Crawling and Pre-Processing

To crawl MobyGames, we used the command `wget -r https://mobygames.com`. We then selected 1026 documents, and pre-processed them using BeautifulSoup to extract the metadata from each .html file. Since our .html files have complex structures that Pyterrier struggles to deal with efficiently, we used BeautifulSoup to extract sections containing essential information and remove other unnecessary tags to simplify our files structures. Additionally, we modified the tags of these sections so that the content is encapsulated inside unique tags, making metadata extraction easier. We then stored the metadata in a dictionary for easier retrieval.

Indexing

Next, we indexed the files using python-terrier's FilesIndexer. We chose not to conduct stemming or stopword removal to preserve semantic meaning when indexing. This was mainly to account for slang or special terminology commonly used within gaming communities that could lose meaning through normalization.

```
files = pt.io.find_files('www.mobygames.com-copy/game')
```

```
# create a folder to store the index
indexer = pt.FilesIndexer('./fi_index', meta={'docno': 1000, 'title': 1024,
'review-text': 30000, 'description': 30000, 'image': 2000, 'mobyscore': 500,
'player-review': 1000, 'critic-review': 1000}, meta_tags={'title': 'h1',
'image': 'img', 'description': 'article', 'mobyscore': 'score',
'critic-review': 'critic', 'review-text': 'review', 'player-review':
'player'})

# do the indexing
indexref = indexer.index(files)

# provide me with a pointer to that index
index = pt.IndexFactory.of(indexref)
```

To determine the optimal retrieval model for indexing, we randomly chose 10 documents from our index and determined their relevance to 5 sample queries: “nintendo”, “racing”, “fight”, “action”, and “arcade”. We then tested TF-IDF, BM25 and LM’s rankings against our relevance judgement matrix.

	name	map	Rprec	recip_rank
0	TerrierRetr(TF_IDF)	0.175133	0.1	0.193723
1	TerrierRetr(BM25)	0.175133	0.1	0.193723
2	TerrierRetr(Hiemstra_LM)	0.208515	0.1	0.229792

Out of the three retrieval models, Hiemstra LM performed the best, at a mean average precision of 0.21, r-precision of 0.1, and reciprocal rank of 0.23. This is likely due to the language model’s ability to pick up on semantic similarity between terms– rather than searching for specific terms like game titles, users are more likely to use MobySearch for more abstract searches like ones based on genre, art style or gameplay. By using a model that focuses on semantics, we’re able to retrieve results that use similar language to the search term rather than focusing on the specific term.

```
lm = pt.BatchRetrieve(index, wmodel="Hiemstra_LM", metadata=["title",
"description", "mobyscore", "player-review", "critic-review",
"company-game", "company-trivia"])
results = lm.transform(queries)
```

Logging and Recommendations

For our recommendation feature, we recommend users popular queries based on the most frequent queries that week. We did this by logging each query's IP address, search terms, and date in a logs.txt file, calculating the count of each search term that week, and displaying the top 5 most searched terms.

```
function getTopQueries() {
    $log_file = 'logs.txt';
    $lines = file($log_file, FILE_IGNORE_NEW_LINES |
FILE_SKIP_EMPTY_LINES);
    $search_counts = [];
    $current_date = time();

    foreach ($lines as $line) {
        $log_parts = explode(',', $line);
        if (count($log_parts) === 3) {
            $log_date = strtotime(trim($log_parts[1]));
            if ($log_date) {
                $date_diff = $current_date - $log_date;
                $days_diff = $date_diff / (60 * 60 * 24);
                if ($days_diff <= 7) {
                    $search_term =
ucfirst(strtolower(trim($log_parts[2])));
                    if (isset($search_counts[$search_term])) {
                        $search_counts[$search_term]++;
                    } else {
                        $search_counts[$search_term] = 1;
                    }
                }
            }
        }
    }

    arsort($search_counts);
    return array_slice($search_counts, 0, 3);
}
```

Usage Scenario

MobySearch offers a straightforward platform for video game enthusiasts to search and explore game information. Users begin their journey on the homepage, featuring a prominent search bar at the top and a "Weekly Trending Searches & Queries" section to the right, which adapts based on their recent activity. The interface is clean and designed for simplicity, enabling users to quickly find and explore game-related content.

A typical interaction starts with some suggested queries placed appropriately under the search bar to help the users start their interaction. MobySearch processes the input and displays a ranked list of relevant games, including essential details like the game's title, description, and other details like Mobyscore, Critic & Player reviews if applicable. The platform also suggests similar queries on this page, encouraging further exploration. Personalized recommended queries dynamically update in a Recommendation section based on popular queries and users' browsing history.

Screenshots

1. If there are any results that match the query (Each result is a card):

MobySearch

Search for games related information

The indexing collection used for this search includes multiple documents from www.mobygames.com. This website provides information on video games, including video game companies, games reviewers and a variety of video games on different platforms. The website also has information about game ratings and reviews given by critics and other players.

Search

SUGGESTED QUERY

cardanimalNintendoracemagicMarioninja turtlesarcade

Search Results

[title: Star Wars: Bounty Hunter](#)

Description: Jango Fett is about to take on a project that could make or break his bounty hunting career, and his life. Chosen to be the blue-print for the Clone Army, Jango must now prove himself in order to gain the biggest job on the streets, wiping out the deadly cult, Bando Gora. Every other bounty hunter in the galaxy is willing and able to do the job, so you must work hard to gain the contract from Darth Tyranus. To win the contract, you must complete a range of missions and collect the bounties on worlds very familiar to Star Wars fans. Jango has his trusty dual pistols to help him out, but also has the ability to use his flame thrower, jet pack, poison darts and much more. Bounty Hunter also has the unique bounty hunting system, where each mission has a specific number of bounties hiding out in the level. Jango must identify them with his helmet gear, and catch them alive...or dead. Bounties allow the player to unlock bonus materials. The 3rd person action title also includes full motion cut-scenes designed by Industrial Light and Magic and Skywalker Sound, plus a few hidden extras and Easter eggs. Bounty Hunter also connects to the movies in that it explains how Jango became the model for the Clone Army of the Republic, and how he became acquainted with Zam Wesell, who makes an appearance in the movies.

Weekly Trending Searches & Queries

See what other people are searching!

Hunter (searched 3 times)

Ninja turtles (searched 3 times)

Race (searched 2 times)

Multiple results returned:

MobySearch

Search for games related information

The indexing collection used for this search includes multiple documents from www.mobygames.com. This website provides information on video game information about game ratings and reviews given by critics and other players.

SUGGESTED QUERY

cardanimalNintendoracemagicMarioninja turtlesarcade

Search Results

[title: Hot Wheels: Beat That!](#)

Description: In Hot Wheels: Beat That!, the player can take the controls of 30 authentic Hot Wheels cars and compete in 12 different racing environments in a variety of single-player games and multiplayer games. Players race through enlarged environments, similar to Micro Machines, that include bedrooms and backyards, avoiding obstacles, knocking out competitors, picking up power-ups and complete stunts through a variety of tracks such as the loop-de-loop. There are standard races along with elimination races where players fire rockets at each other in an attempt to knock each other out of the race.

Mobyscore: 6.4

Critic-review: Average score: 50% (based on 12 ratings)

Player-review: Average score: 3.9 out of 5 (based on 20 ratings with 0 reviews)

[title: Ridge Racer 6](#)

Description: Big drift racing is back in Ridge Racer 6. There are 130 cars, loads of prizes and other unlockables in World Xplorer mode. World Xplorer sees you going through a hexagonal based 'universe' from easy to difficult races. Nitro is gained by drifting and jumping through the races and allows you to build up three nitro boost cells to use one at a time or one after the other for a bigger boost. For the first time in the series, Ridge Racer goes online with Online Battle Mode allowing up to 14 players to drift against each other. Also, if you don't want to race you can watch other players racing on Spectator mode. Global Time Attack mode challenges you to get the best times and upload them to the scoreboards, to see if you are best in the world or not.

Mobyscore: 7.4

Critic-review: Average score: 75% (based on 30 ratings)

Player-review: Average score: 3.7 out of 5 (based on 12 ratings with 0 reviews)

2. If no result is returned for the given query, output is the string “No result found for this query”:

MobySearch

Search for games related information

The indexing collection used for this search includes multiple documents from www.mobygames.com. This website provides information on video games, including video game companies, games reviewers and a variety of video games on different platforms. The website also has information about game ratings and reviews given by critics and other players.

SUGGESTED QUERY

cardanimalNintendoracemagicMarioninja turtlesarcade

Search Results

No result found for this query

Weekly Trending Searches & Queries

See what other people are searching!

Ninja turtles (searched 3 times)

Hunter (searched 2 times)

Race (searched 2 times)

Issues

As we implemented the project, one big issue we encountered was the structure of HTML files. Given the deeply nested structure, it is hard to index information using Pyterrier since it only supports simple HTML files with now nested tags. Since our collection is crawled from different pages of the MobyGames website, the structure for each page is inconsistent, which adds another layer of complexity and makes it difficult to index without preprocessing the crawled files. Additionally, Pyterrier ignores anchor tags and hyperlinks so we need to do more preprocessing to retrieve the important information stored in the attributes of these tags such as the src attribute of elements or the href attribute of the <a> tags.

Another issue we had was the limitation of the course server and since a lot of people have projects and assignments running on the server, it might take a while for us to get the indexed information to display on the UI. Working with files on the course server also makes it hard to make changes or debugging the files because every time I make any changes, I would need to use Filezilla to transfer the file from my local computer to the server to test it out.

Future Work

While MobySearch already serves as a platform for exploring video game data, there are numerous opportunities for future enhancements that could expand its functionality and interactivity. Additional visual features, such as trend-maps or regional popularity trends, could provide deeper insights.

The recommendation system could be improved by integrating machine learning models for more precise personalization. Social features, such as recommending games positively reviewed by similar users, would also enhance user engagement.

Expanding the database to include user reviews, and other content like walkthroughs, and gameplay videos would make MobySearch a comprehensive resource. Crawling additional related sites like Steam, IGN, or Metacritic could enrich/expand the dataset. Developing session features, such as bookmarking searches or saving game lists, would improve user convenience and continuity. Optimizing the site for mobile devices would ensure accessibility for users on the go.

By pursuing these enhancements, MobySearch could evolve into a versatile and comprehensive tool, appealing to gamers, researchers, and industry professionals alike, while providing an indispensable resource for video game exploration and discovery.