Steam Games - Information Processing and Retrieval

Gustavo Costa
MEIC - FEUP
Porto, Portugal
up202004187@up.pt

João Oliveira
MEIC - FEUP
Porto, Portugal
up202004407@up.pt

Luís Miranda
MEIC - FEUP
Porto, Portugal
up201306340@up.pt

Ricardo Cavalheiro
MEIC - FEUP
Porto, Portugal
up202005103@up.pt

Figure 1: Steam Games - Information Processing and Retrieval

ABSTRACT

In today's day and age, games are becoming increasingly more common and mainstream. With that in mind, we decided that this would be a good topic when it comes to the process of retrieving and preparing information for use in search platforms like Solr[1]. We found a rather large dataset (roughly 81000 entries) in Kaggle[2] containing relevant information, like game names, descriptions and prices, obtained from Steam[3], one of the biggest and most popular gaming platforms. One of the downsides of having lots of raw information is that it's almost guaranteed that you'll also have a lot of junk or outright useless data so, using the raw datasets as a starting point, we built a Python[4] pipeline capable of cleaning and preprocessing the data, so that it was ready to be used for information searching tasks. The steps used to retrieve and prepare the data and the tools and techniques used to clean and preprocess the data will be further discussed in detail in this paper.

CCS CONCEPTS

• **Information systems** → **Information retrieval**.

KEYWORDS

Data retrieval, Data Processing, Data Transformation, Data Cleaning, Data Integration, Python Pipeline, Steam games

1 INTRODUCTION

Games have become a cultural pillar in today's world of digital and interactive experiences. Given how prevalent games have become in today's society, this paper focuses on them as a thematic research. Steam stands out as an optimal data reservoir owing to its extensive user base and varied game selection, offering a comprehensive dataset that encompasses a diverse spectrum of gaming behaviors. Its digital nature enables precise data collection, encompassing playing statistics, user reviews and detailed information on an immense quantity of games.

2 DATA SOURCE IDENTIFICATION

The dataset used in this study was obtained from Kaggle[5] and consists of two files: `steam_data.csv` (36MB) and `text_content.csv` (148MB), each with about 81,000 entries. While `steam_data` offers a more organized set of information, focusing on non-descriptive features like categories, release dates and developers, `text_content` is rich in textual properties, prominently presenting descriptions and other narrative components relative to each game. The Steam API[6] also acts as a data source and is essential for obtaining the initial datasets' missing data, resulting in richer and more complete sets of data that could be used for future analysis.

[1] https://solr.apache.org/
[2] https://www.kaggle.com/
[3] https://store.steampowered.com/
[4] https://www.python.org/
[5] https://www.kaggle.com/datasets/deepann/80000-steam-games-dataset
[6] https://steamcommunity.com/dev

3 INITIAL DATA CHARACTERIZATION

In order to be able to preprocess the data, we needed to know what the data looked like, so that we could employ the correct procedures. With this in mind, after merging both csvs obtained from Kaggle and removing the duplicated information, we ended up with the following characterization of our data.

Attribute	Type	Null Count	Null Percentage
publisher	Text	76190	94.19
pegi	Text	70373	87.00
pegi_url	Text	66098	81.71
desc	Text	31597	39.06
categories	Text	6822	8.43
price	Text	6323	7.82
requirements	Text	6222	7.69
popu_tags	Text	5764	7.13
date	Text	5636	6.97
developer	Text	5547	6.86
all_reviews	Text	5145	6.36
name	Text	5137	6.35
img_url	Text	5134	6.35
user_reviews	Text	5132	6.34
full_desc	Text	1812	2.24
url	Text	0	0.00

Table 1: Attribute types, Null Counts and Percentages for Attributes

Descriptions:

- **publisher**: This attribute represents the company that published the game.
- **pegi**: PEGI[7] is a video game content rating system.
- **pegi_url**: URL associated with the PEGI rating.
- **desc**: A brief description of the game.
- **categories**: Represents the genres or categories to which the video game belongs.
- **price**: This attribute contains the cost of the video game.
- **requirements**: This column consists of information about the system requirements or specifications needed to run the video game.
- **popu_tags**: Includes popular tags given for the game by the users and developers.
- **date**: Represents the release date of the game.
- **developer**: Company which developed the game.
- **all_reviews**: Contains game reviews, from users and non-users, statistics.
- **name**: This attribute is the name or title of the video game.
- **img_url**: URL associated with the preview image of the game.
- **user_reviews**: Contains game reviews, from users, statistics. Consists of the following information: overall impression of the review, number of such reviews and percentage of positive reviews in the last 30 days.
- **full_desc**: Elaborate description of the game. Also contains information like whether it is a game or extra content (i.e. DLC).
- **url**: URL associated with the game. It contains the ID of the game.

At this stage it is important to note a couple of points:

- Typical numerical fields are classified as text fields. In the case of the "price" field, this happens because the numerical price is prefixed with the dollar sign ($).
- The "Null Percentage" column indicates the percentage of nulls per attribute field. For example, 94.19% of the entries in the "publisher" field contain the value "NULL"
- The "url" field contains no "NULL" values because they were automatically removed when the merging of data into a single source occurred, being this field the common field between both files.

4 DATA COLLECTION & PREPARATION

Data preparation and collecting surfaced as a significant challenge when dealing with the initial dataset of our project. The data's initial unstructured state was a big barrier, necessitating careful work to transform disorder into coherence and ensure a solid basis for future analysis and use.

4.1 Pipeline Description

We were tasked with building a pipeline capable of handling and preparing the raw data so that it could be used later. To do so, we chose Python due to its scripting language nature. We also chose to host the data on an SQLite[8] database, instead of the original csv format. SQLite is a fast and simple relational database management system with good Python integration, having many of the needed tools for the job already built in and optimized, allowing for efficient operations.

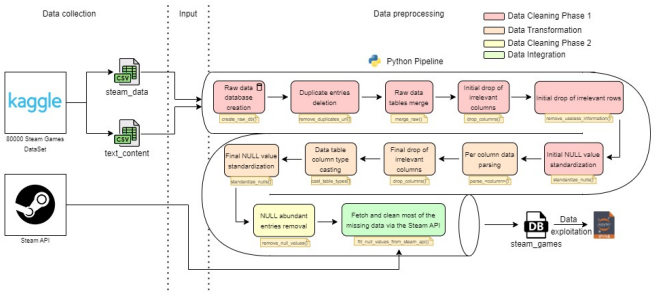

Figure 2: Pipeline Diagram

The previous figure displays the pipeline architecture diagram that our pipeline follows. As observable, the data goes through five distinct phases before it is ready to be used, explained in detail in the following topics.

[7] https://pegi.info/

[8] https://www.sqlite.org/index.html

4.2 Data Collection

Initially, we downloaded a dataset from Kaggle containing information on around 81,000 Steam games. However, the data proved chaotic, demanding extensive cleaning. Many fields ended up with null values, prompting us to consider filling them by making calls to the Steam API. Nevertheless, utilizing these API calls proved to be restrictive due to the rate limits imposed Steam. Despite allowing 100,000 calls per day, it has a soft limit of approximately 200 calls every 5 minutes, and given the abundance of null values, the process took a considerable amount of time to complete (11 hours).

4.3 Data Preprocessing

Data preprocessing is the phase of cleaning up chaotic, raw data. This involves deleting duplicates, filling in missing values and transforming the data into a format that can be understood. It also means figuring out which fields would play an important role in powering the search engine.

4.3.1 Data Cleaning Phase 1.

In this phase, the following tasks are performed:

- **Merging Dataset and Removing Duplicates:** Merging datasets and eliminating duplicate records to ensure data integrity and consistency.
- **Dropping Columns:** Removing irrelevant columns (pegi, pegi_url, img_url, popu_tags) to simplify the dataset.
- **Removing Specific Entries:** Removing entries related to DLCs, soundtracks, packs and new editions, as they have no relevance in this work's context. It also helped to reduce the size of the dataset.
- **Standardizing Nulls:** Standardizing the representation of null values for consistency.

4.3.2 Data Transformation.

Data transformation involves reformatting and parsing various data attributes to make them more suitable for analysis. Key tasks include:

- **Parsing Price Attribute:** Standardizing price formats for numerical analysis.
- **Parse Full Description:** Removed 'About this' prefix for all the entries for better readability.
- **Parsing Categories:** Parsing and standardizing game categories for better categorization.
- **Parsing Date:** Formatting date values consistently (e.g., DD MM, YYYY).
- **Parsing System Requirements:** Breaking down complex system requirement data into structured components.
- **Parsing Reviews:** Extracting and parsing the percentage of positive reviews from the textual review information.
- **Parsing Developer:** Standardizing developer names for easier analysis.
- **Dropping Columns (user_reviews, publisher):** Removing columns that are not relevant to the analysis. The publisher contained a high percentage of null values.

4.3.3 Data Cleaning Phase 2.

The need for a second phase of data cleaning arises from inconsistencies introduced during the parsing and transformation steps:

- **Standardizing Nulls:** Continuing the standardization of null values to ensure data consistency.
- **Removing Entries with more than 3 Null Values:** Removing records with a significant number of missing values while considering the impact on data integrity.

4.3.4 Data Integration.

Data integration is the final phase, focusing on filling in missing data using external sources, specifically the Steam API.

- **Filling Null Values with Data from the Steam API:** Retrieving missing data from the Steam API to enhance the dataset's completeness and accuracy.

4.4 Conceptual Data Modelling

After preprocessing a clear model for our data can be defined, as follows:

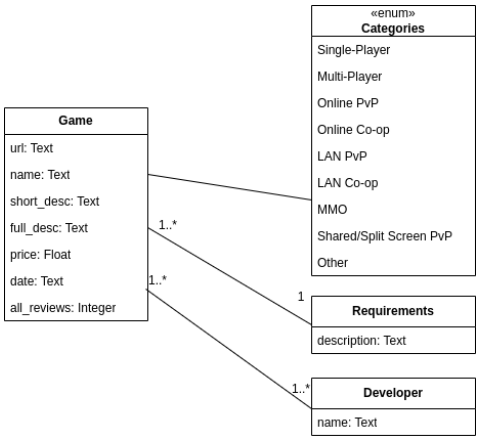

Figure 3: Conceptual Data Model

Only three entities are clearly identifiable, being the game, its requirements and its developer. Other than that, a game can have a list of categories that easily help with identifying if a certain game is desired when a certain information need is carried out.

5 DATA CHARACTERIZATION

After running the pipeline through the initial dataset, the processed data ended up with exactly 48219 entries and 10 features, with both numerical and textually rich attributes.

5.1 Collection Characterization

Attribute	Type	Null Values	Null Percentage (%)
all_reviews	Integer	6532	13.54
price	Float	677	1.40
developer	Text	82	0.17
desc	Text	65	0.13
requirements	Text	42	0.09
date	Text	41	0.09
full_desc	Text	29	0.06
url	Text	0	0.00
categories	Text	0	0.00
name	Text	0	0.00

Table 2: Final Dataset Attributes, Type, Null Count and Null Percentage

Compared to the original dataset, the refined version now boasts approximately 48,000 entries—roughly half of its initial size. This reduction is primarily attributed to the elimination of redundant entries, such as DLCs and soundtracks, streamlining the dataset for greater precision and relevance.

5.2 Free & Paid Game Releases

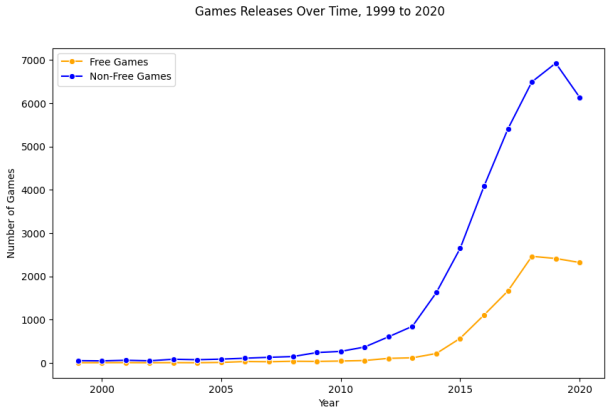

Figure 4: Games Released from 1999 to 2020

The previous plot illustrates an increase in game releases over time, with one standout trend: most of these games are not for free. This indicates a shift in developer preferences toward paid models, which may be caused by reasons like rising development costs or a desire for higher financial rewards. The amount of games available is growing and so are the monetization methods used.

5.3 Game Review Evolution

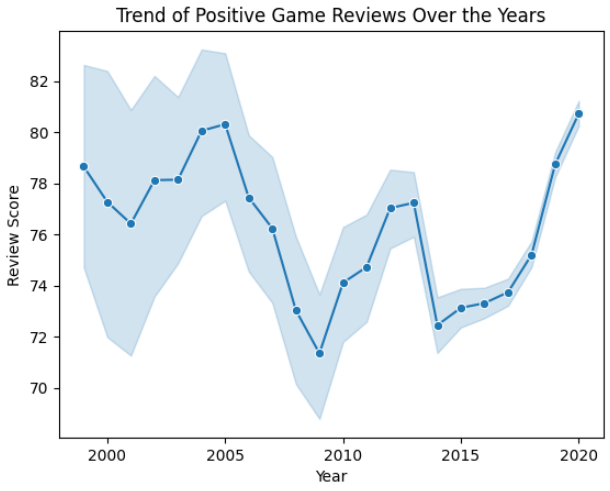

Figure 5: Average Positive Game Reviews from 1999 to 2000

Despite the oscillations, the overall trend seems to be that the users are actually enjoying the different games. It's interesting how the plot showcases a consistent range of 70-80% positive reviews per year.

5.4 Text Analysis

Figure 6: Word Cloud Image

We made a Wordcloud based on the attribute 'full_desc', which is the one to be most likely used in the search engine as it is the richest in textual data. We can observe that common words like 'Game', 'Play', 'Battle' and 'Fight' are predominant, which goes according to our expectations.

6 INFORMATION NEEDS

- Single-player games released in the last decade with an underwater setting
- Highest Rated Games with futuristic landscapes and advanced technology
- Highest Rated Free Games
- Least Expensive First-Person Shooter in a post-apocalyptic setting
- Developers with the most Single-Player games in the last decade
- 2D platform game where the main character is an animal
- Best Free Fishing Games

7 CONCLUSION

We were able to put together a strong and well-structured collection despite the unstructured data in the original dataset and also the difficulties encountered when dealing directly with the Steam API. The result of all of these efforts is a comprehensive dataset with both textual and numerical elements, ready to be used in the future.