Steam Scraper

Matthew Dixson,
Tom Hills, Rob Bray, Tan Nguyen

Objectives

- Scrape data from Steam's online storefront
- Data pertaining to the following topics:
  - Genre
  - Tags
  - Price
  - DLC Price and Discounts
  - User Reviews
  - System Requirements

Data Collection

- Originally planned on using Steam's web API to collect data
  - Severely limited in the type of data it provides
  - Used this to obtain a list of over 100,000 ID-title mappings
- We used this list to scrape directly from Steam's storefront
- https://store.steampowered.com/app/**489830**
  - Number at the end is the app's ID
- All members used Beautifulsoup4, but each scraped their own datasets of roughly the same size for their chosen topic
  - Some datasets were combined for data integration

Datasets

Matthew

DLC- 4,142 entries

Reviews- 80,567 entries

Tom

Dataset- 111,112 entries.

Rob

Descriptions- 113,920 entries

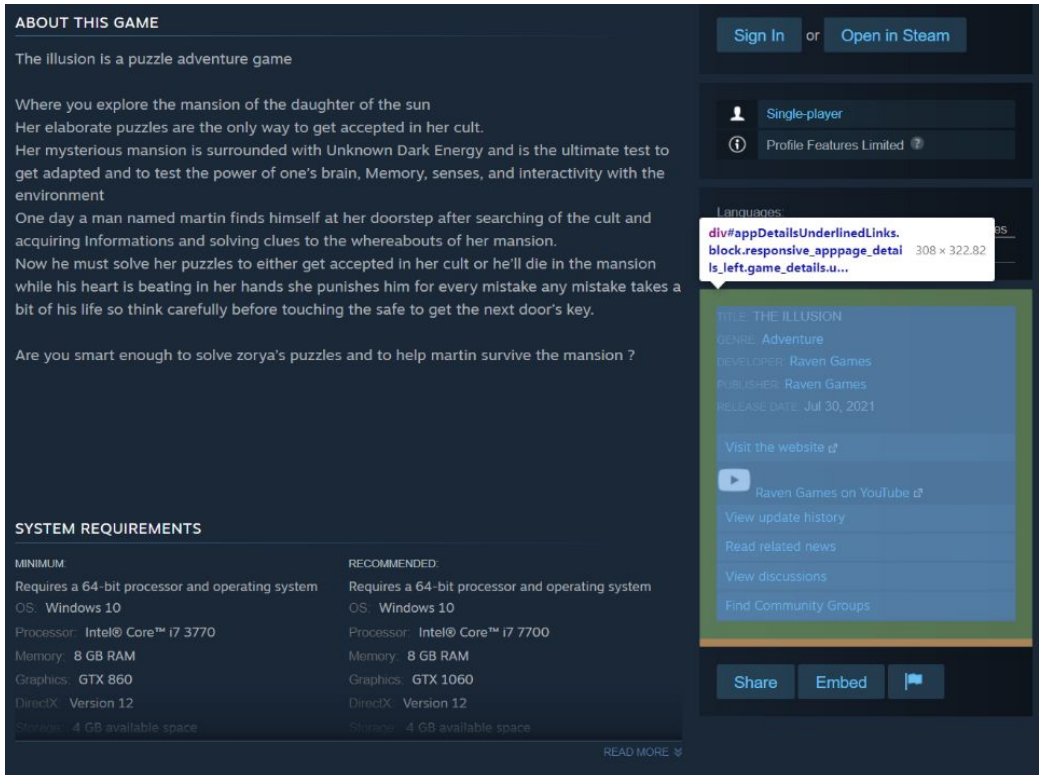
Tan

Recommended and minimum system - 77,400 entries each

Datasets were formatted as CSV files for easy parsing with Python and Excel

Data Collection

- Filtered Steam items to remove NSFW material
- Filtered results were used to scrape tags, genre, price, DLC information, user reviews, and system requirements from steam webpages using appID and Beautifulsoup4



Data Collection Problems

- Unstandardized HTML layouts
  - Varying elements for same property
  - Many pages without the desired property defined

```
# PRICE ----------------------------------------------------------------
# <div class="game_purchase_price price" data-price-final="1499">$14.99</div>
# <div class="discount_original_price">$0.99</div>
price_noDiscount = soup.find(class_='game_purchase_price price');
price_discount = soup.find(class_='discount_original_price');
if price_noDiscount is not None:
    price = price_noDiscount.text.strip();
    #print(f"    {game_name} ==> price_noDiscount: ${price}");
elif price_discount is not None:
    price = price_discount.text.strip();
    #print(f"    {game_name} ==> price_discount: ${price}");
else: price = "Bug: enter price manually"
```

Data Collection Problems

- Timeouts during URL requests

```
URL = 'https://store.steampowered.com/app/' + game_id

try:
    page = requests.get(URL)
except:
    print(f"REQUEST ERROR. Continuing....");
    continue;

soup = BeautifulSoup(page.content, 'html.parser')
```

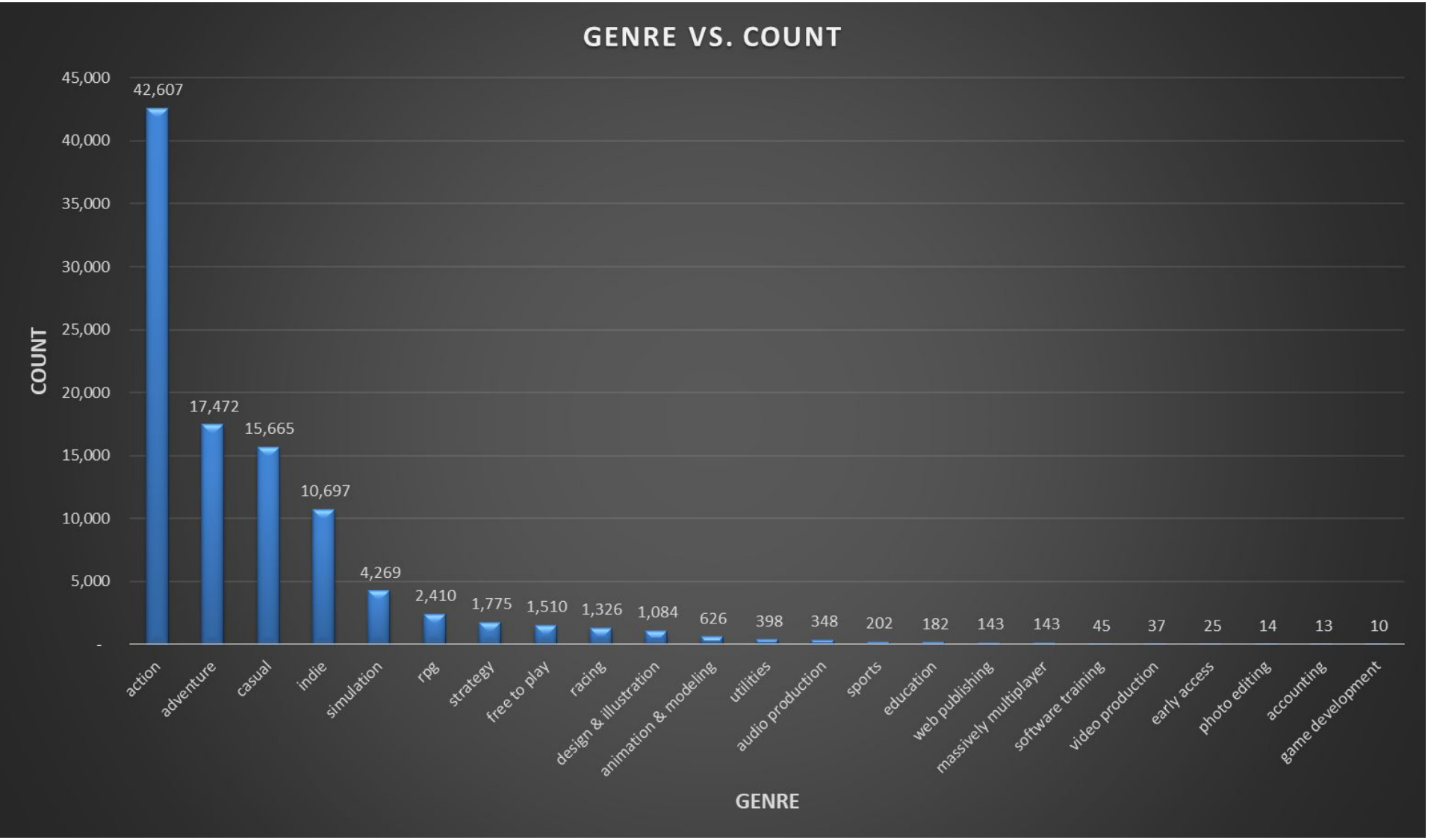
Data Collection Problems

- Long runtimes. Solutions?
  - Multiprocessing
  - Uninterrupted sequential processing
  - Stop/Resume sequential processing

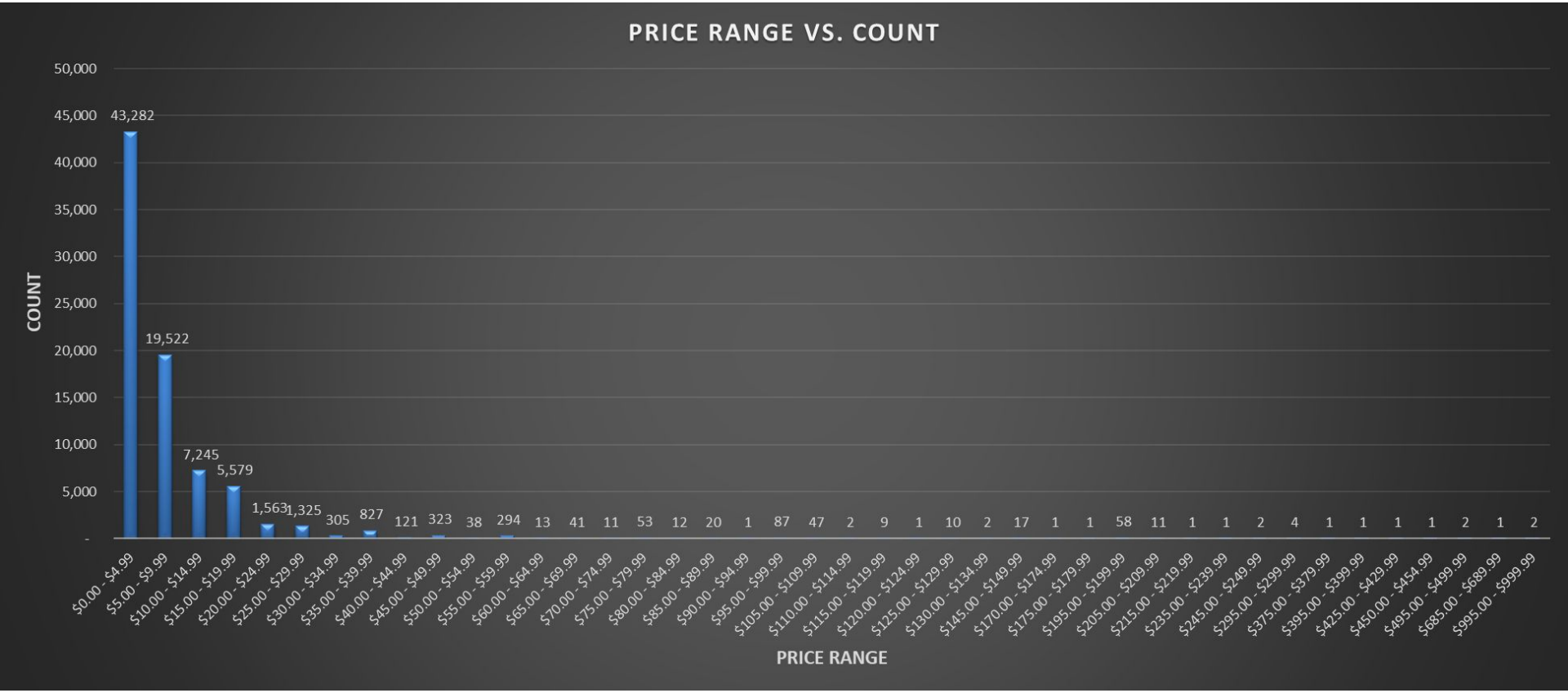
```
game_name = line.split(',')[0].strip();
game_id = line.split(',')[1].strip();

# resume code:
if game_id != lastGameID and not continueScrapping:
    #print(f"line {lineNum}: {game_id} != {lastGamel
    continue;
elif game_id == lastGameID:
    #print(f"line {lineNum}: {game_id} == {lastGamel
    continueScrapping = True; continue;
```

Data Processing: App Count vs. Genre



Data Processing: App Count vs. Price Range



Min	Max	Avg	Median
$0.49	$999.00	$8.85	$4.99

Data Integration

- We combined our datasets in order to make observations about our data
- Looked at:
  - DLC Prices and Discounts vs. Genre
  - User Reviews vs. Genre
  - Description vs Genre

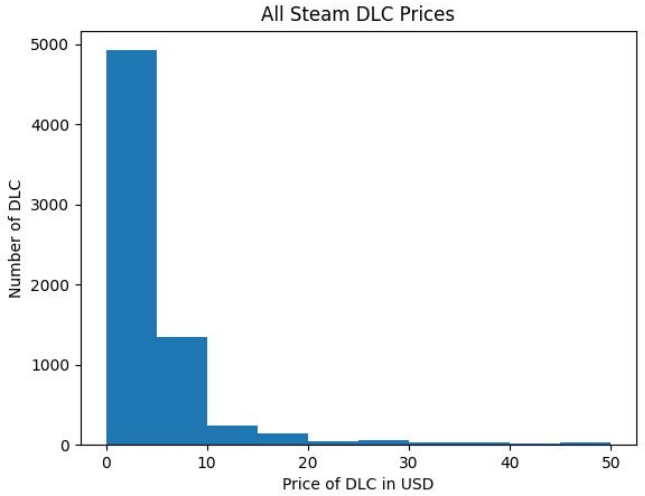
DLC Prices

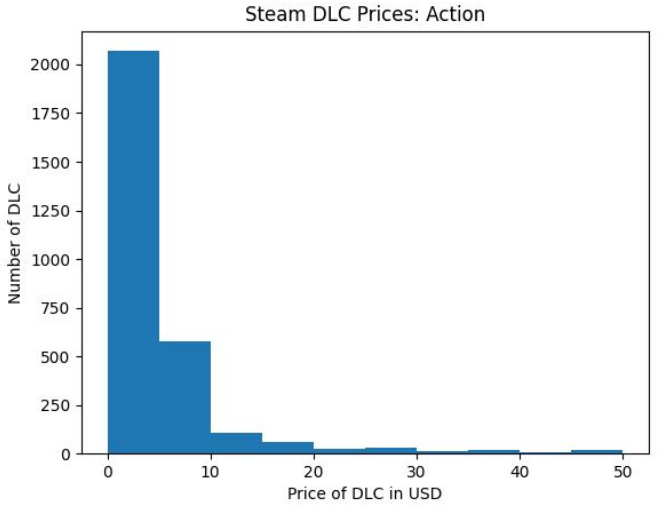
All Genres:

Mean: 6.09
Median: 3.99
Mode: 4.99 | Count: 1052
Standard deviation: 12.995518228520467
Min: 0.0
Max: 450.0
Total considered: 6918
DLCs ignored due to bad formatting: 3

Action:

Mean: 6.41
Median: 3.99
Mode: 4.99 | Count: 454
Standard deviation: 12.56819370257585
Min: 0.0
Max: 199.99
Total considered: 2951
DLCs ignored due to bad formatting: 0

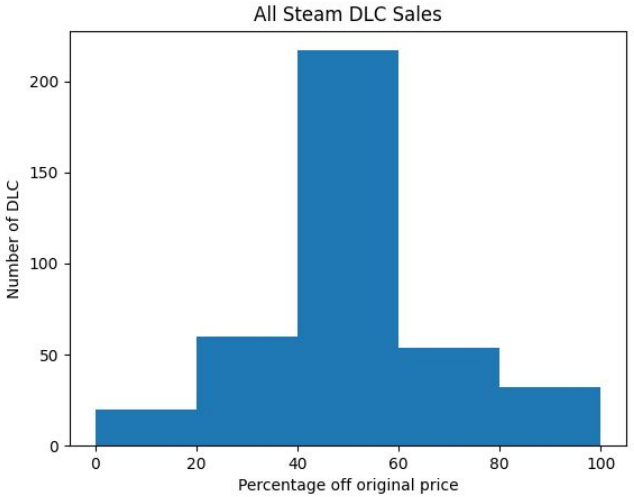




DLC Discounts

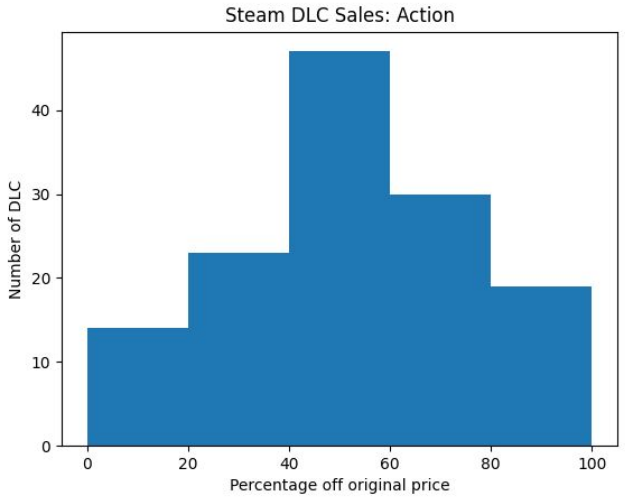
All Genres:

Mean: 49.23
Median: 50.0
Mode: 51.0 | Count: 104
Standard deviation: 18.333562987007383
Min: 5.0
Max: 100.0
Total considered: 383



Action:

Mean: 50.73
Median: 51.0
Mode: 51.0 | Count: 18
Standard deviation: 23.315788261366055
Min: 10.0
Max: 100.0
Total considered: 133



Data Processing: Review Conversion

- Text descriptive rating to Numerical rating
  - If app has too few ratings, it will not get a text descriptor, just the number of ratings

Overwhelmingly Positive -> 6

Very Positive -> 5

Positive / Mostly Positive -> 4

Mixed -> 3

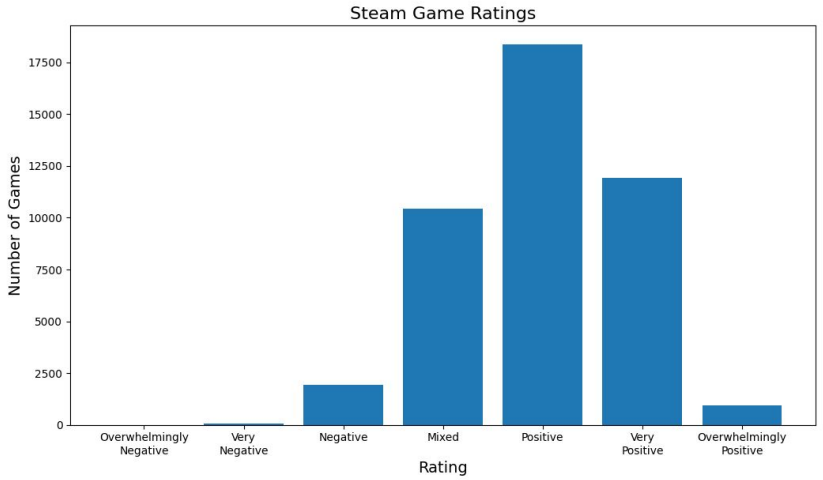
Negative / Mostly Negative -> 2

Very Negative -> 1

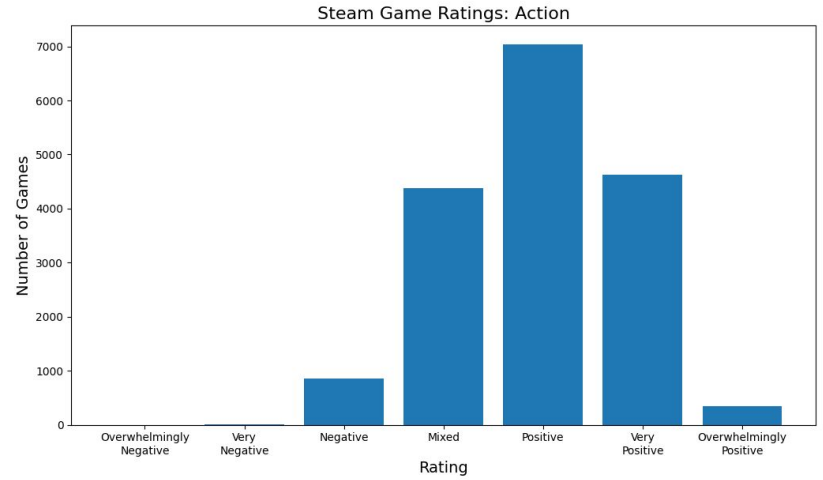
Overwhelmingly Negative -> 0

User Reviews

All Genres:
Mean: 3.99
- Median: 4.0
- Mode: 4 | Count: 18373
- Standard deviation: 0.8877693802708788
- Min: 0
- Max: 6
- Total considered: 43652
- # of entries removed: 36896



Action:
Mean: 3.95
- Median: 4.0
- Mode: 4 | Count: 7039
- Standard deviation: 0.8999604277629792
- Min: 0
- Max: 6
- Total considered: 17283
- # of entries removed: 13457



Natural Language Processing

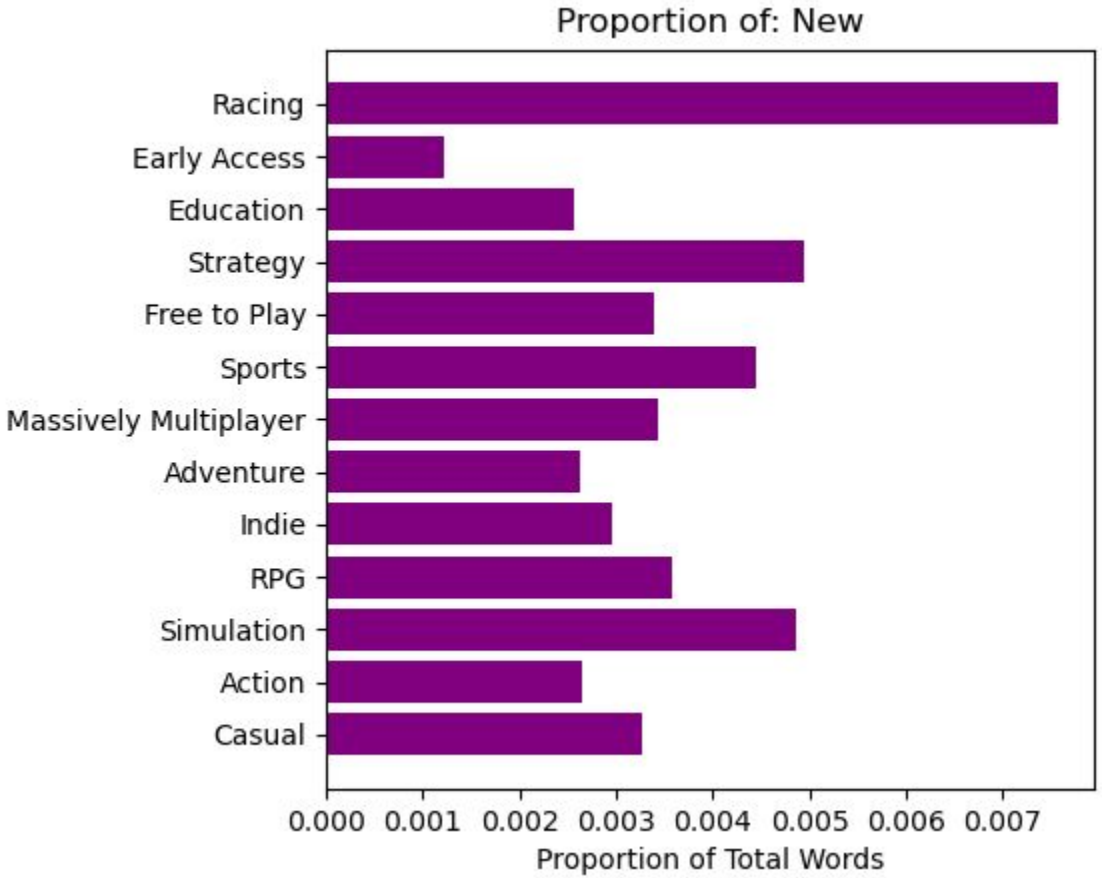
- Natural Language Toolkit (NLTK) used to filter and tag words in descriptions
- Part of Speech (POS) tagging is an averaged perceptron

```
# for most-used types of words
tokenized = pos_tag(genre_words[key])
tags_counted = list(Counter([j for i,j in pos_tag(genre_words[key])]).items())
print(key, tags_counted)
```

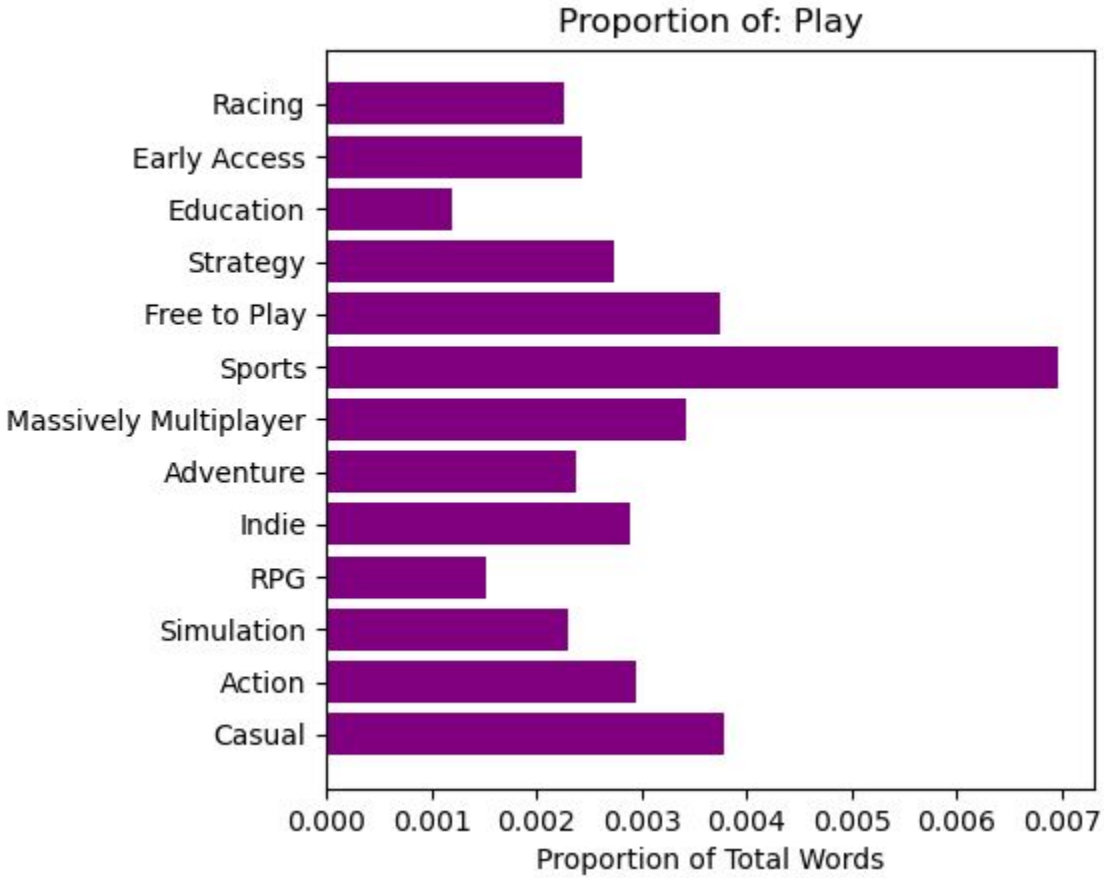
```
Casual [('NN', 78072), ('VBZ', 10602), ('DT', 36426), ('JJ', 36203),
VBD', 2524), ('WRB', 1656), ('PDT', 480), ('NNP', 1255), ('JJR', 932)
Casual [('puzzle', 2135), ('play', 1287), ('world', 1166), ('new', 11
```

```
# these are the words we want to ignore
filter_words = ['game', ''] + stopwords.words('english')
```

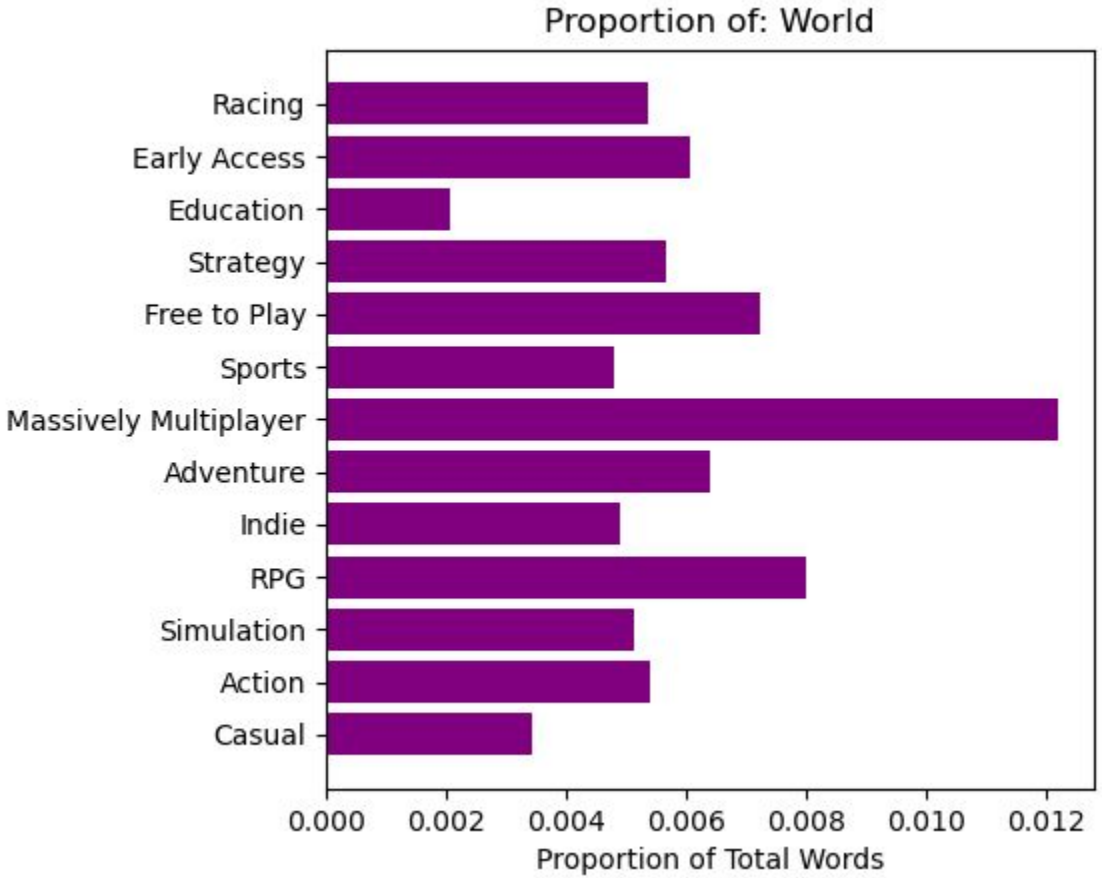
Genre & Description Words



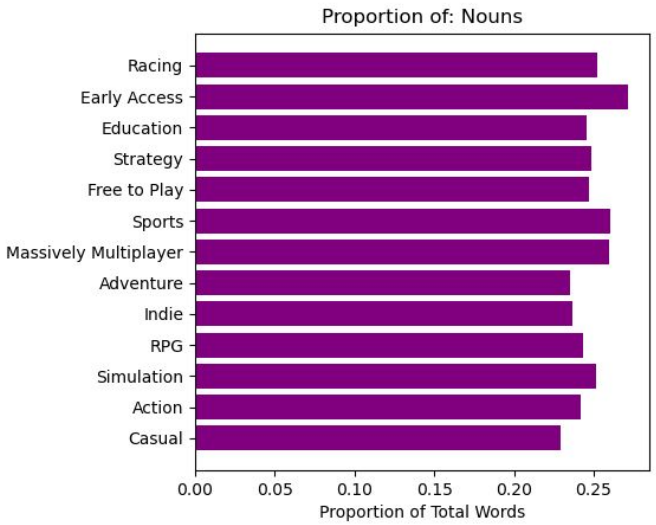
Genre & Description Words

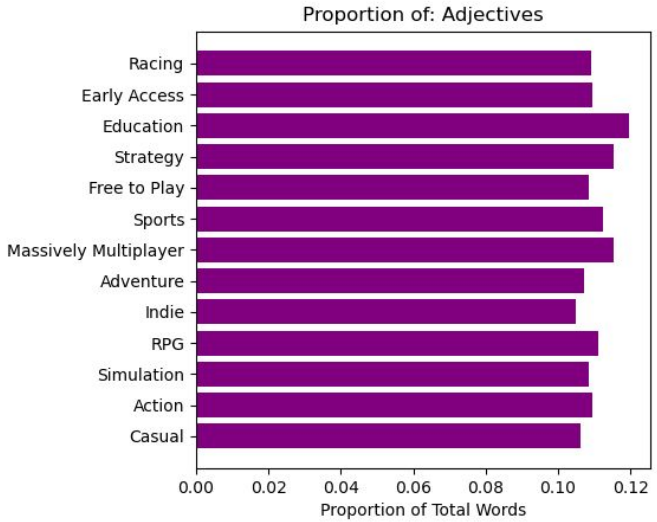


Genre & Description Words

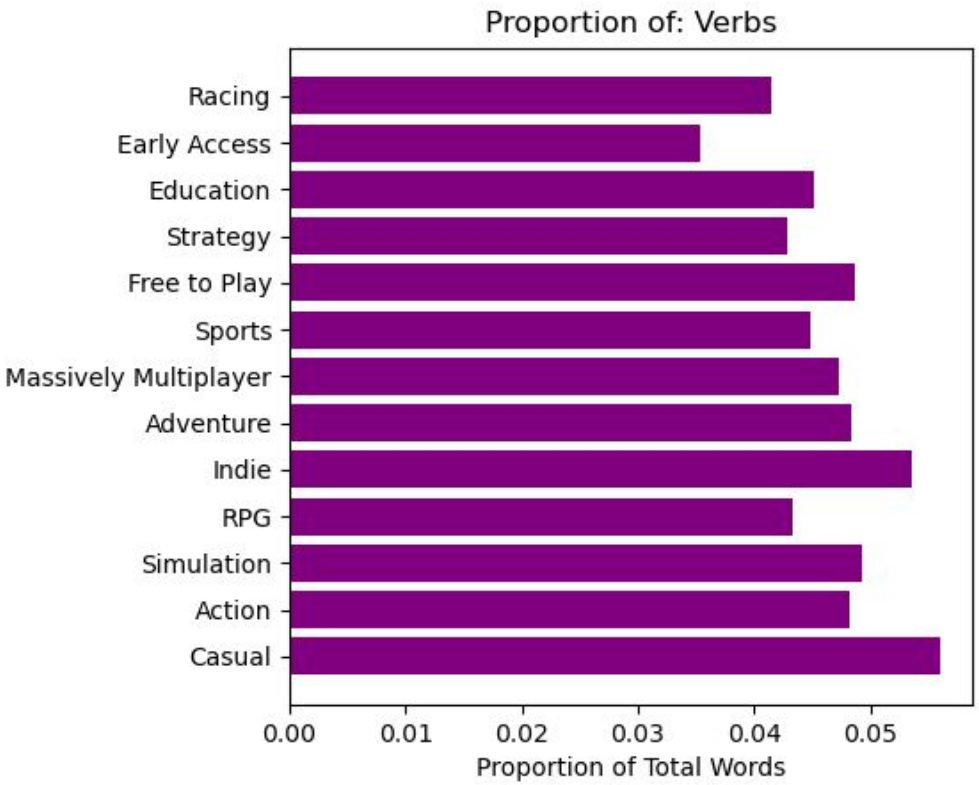


Genre & Types of Words





Genre & Types of Words



Minimum & Recommended System Requirements

- No multiprocessing function.
- No resume function.
- Let program run overnight, around 8 hours until Steam stop.
- 77400 entries finished.
- There are ⅓ entries left that I did not finish.

Example Recommended System Requirements

```
black squad - ea free timed weapon package 1,  654170
OS: Windows 7 64bit
Processor: Core i3-4170 or AMD FX-8300
Memory: 4 GB RAM
Graphics: NVIDIA GTX 760 or AMD Radeon HD 7950
DirectX: Version 9.0
Storage: 7 GB available space

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
black squad - ea free timed weapon package 2,  654171
Requires a 64-bit processor and operating system
OS: Windows 7 64bit
Processor: Core i3-4170 or AMD FX-8300
Memory: 4 GB RAM
Graphics: NVIDIA GTX 760 or AMD Radeon HD 7950
DirectX: Version 9.0
Storage: 7 GB available space
```

Example Minimum System Requirements

```
black squad - ea free timed weapon package 1,  654170
OS: Windows 7 64bit
Processor: CORE2 DUO 2.2GHZ / AMD Athlon 64 X2 2.66GHz
Memory: 4 GB RAM
Graphics: NVIDIA GEFORCE 8600 OR GT630 / RADEON HD 6750
DirectX: Version 9.0
Storage: 7 GB available space

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
black squad - ea free timed weapon package 2,  654171
Requires a 64-bit processor and operating system
OS: Windows 7 64bit
Processor: CORE2 DUO 2.2GHZ / AMD Athlon 64 X2 2.66GHz
Memory: 4 GB RAM
Graphics: NVIDIA GEFORCE 8600 OR GT630 / RADEON HD 6750
DirectX: Version 9.0
Storage: 7 GB available space
```

Limitation Minimum & Recommend Sys Requirements

- The data could not be statistical because the authors entered very different information for different games.

```
fernbus simulator - france,  654120
Requires a 64-bit processor and operating system
OS: Windows 7 / 8 / 8.1 / 10 (64bit only)
Processor: Intel Core i5 Processor or similar with at least 2.6 GHz
Memory: 6 GB RAM
Graphics: Nvidia GeForce GTX 560 or similar AMD Radeon (no support for onboard cards)
DirectX: Version 11
Storage: 20 GB available space
Sound Card: Yes

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
realms of arkania: star trail - digital deluxe content,  654140
OS: Windows 7, 8, 10
Processor: Intel Core2Duo / AMD X2, min. 2.4 GHZ
Memory: 4 GB RAM
Graphics: Nvidia Geforce GTX 560 / AMD Radeon 7850, min. 2GB VRAM
DirectX: Version 9.0c
Storage: 1700 MB available space
```

Future Work

- NLTK Tagging:
  - Tagging gets very detailed (i.e. different tense forms for verbs)
  - Include all forms of verbs in proportion calculations
- We looked at all apps sold on Steam
  - Separate analysis based on whether its a game, video, or other software.
- Making a model to predict game's genre based on its features

Questions?