COSC 445: Final Project Report

I. OBJECTIVES

The original objective was to analyze how the limited availability of streaming services affected piracy in countries where they were not available. This was not feasible since most location data when it comes to piracy is not available. Even if it was obtainable, many people use VPNs to mask their IP addresses and locations when torrenting, so there is no way to guarantee that the information would be accurate. So the objective changed.

Instead of general piracy, the focus was shifted to torrenting, which is a much easier kind of data to track. Torrenting numbers for one hundred shows across five streaming services were analyzed. Also, torrenting twenty non-exclusive shows were analyzed as a control group. From these numbers, the hope was to discover how each of these services comparatively suffers from torrenting, how torrenting for international streaming services compared to ones with limited availability, and how covid-19 and the subsequent lockdowns affect the rate of torrenting.

II. DATA

To begin the data collecting process, six lists of 20 shows were created. One for each streaming service that we were using. Those streaming services were Netflix, Hulu, Disney+, HBO max, and Amazon Prime Video, and one for a non-exclusive control group. The non-exclusive list was all shows that are not specific to one streaming platform. The data was then collected using a program called Octoparse which is a visual web data extraction software. The site the torrent data was collected from was a torrenting search engine called solidtorrents.com, which provides lists and download numbers for torrents across the internet. In Octoparse there are tasks that were made to scrape the Solid Torrents site for using the url of Solid Torrents with the title of the show. This was all saved to JSON files so that the parser could read through and interpret the data.

The next step was to clean and validate the data to ensure torrents for other shows or other pieces of media were not being included in the results.. This was done by creating a library of words that would skip lines of data that did not fit what we were searching for. An example of this would be The Witcher being a Netflix show and The Witcher 3: Wild Hunt being a video game. Words like “Wild” and “Hunt” were added to the library which would cause that line in the JSON file to be skipped.

The data was then integrated into a specially designed parser written in python, from the JSON files and stored in special objects designed for holding data. The parser would take each JSON file and iterate through it and store the name, total, date, and service to be used later on.

III. ALGORITHMS & MODELS

The processing and preliminary modeling of the data obtained by Octoparse was all done through a parser written in python. The torrent data was exported in the JSON file format, and stored in a local directory on the machine that ran the parser. The parser iterates through the directory that contains the JSON files for each streaming service, and reads the raw text for each of them into the program. The python JSON library is then used to automatically turn each JSON string into a python dictionary. Next, the parser uses a class structure to represent each show and the data that relates to that show; the class variables include the name of the show, the total number of torrent downloads for that show, the oldest torrent date found for the show, and what streaming service the show is found on.. To store this information, a new dictionary is used which is keyed on the name of the show, with the value being the class structure that holds the data. At the end of this importing process, the program has a dictionary with every show in the dataset accounted for.

The next process for the parser involves how to process the raw data that is now cleanly stored. To get the data that was needed to formulate the analysis, different checkpoints throughout the program were established where important data could be gathered. The program keeps track of the running total of many different statistics; the total number of torrents for streaming service shows compared to non-exclusive shows, the number of torrents from different time periods during the COVID-19 epidemic, and whether the shows were available internationally or were region locked.

Finally, the program made use of the Matplotlib python graphing and data visualization library in order to get a rough estimate and visualization of the data obtained throughout the project's lifespan. Although these specific graphs were not used in the end product, they were useful throughout development in order to ensure that the parser was working correctly, and to get an easy visual understanding of the data that had been obtained thus far.

IV. RESULTS

Of the 120 television shows analyzed, 40,724,907 unique downloads were found. Of these, 12% belonged to Netflix exclusives, 15% to HBO Max exclusives, 16% to Hulu exclusives, 18% to Amazon Prime Video exclusives, and 25% to Disney+ exclusives. The remaining 14% were part of the non-exclusive control group. This meant that 86% of the total number of torrents were from the streaming services. The streaming services with international availability (Disney+, Netflix, Hulu) averaged to 7,238,052 torrents per platform, while the services with limited availability (Amazon Prime, HBO Max) averaged 6,635,817 torrents per platform. The pre-lockdown average for torrents per year was 610,593 over a span of 10 years, while the post-lockdown average for torrents per year was 1,560,740 over a span of ~2 years.

The results of this data collection could imply some interesting things. Firstly, Disney+ dominated with a fourth of the total number of torrents. This was initially surprising, considering Disney+ only released recently and finding twenty released exclusives for the platform was a struggle. However, it makes more sense when compared to the post-covid numbers.

Torrenting nearly tripled after the lockdown, and Disney+ released only a few months prior to that. Meaning nearly all of their exclusives would benefit from this boost. Also, while Disney+ is available to a more international audience now, it was fairly limited at its release, meaning less people had access to their exclusives.

Netflix was another surprising statistic. The streaming service was the oldest of the ones selected for this analysis, and had the widest array of exclusives to choose from. Despite this, it had the lowest number of torrents. No doubt if the entire exclusive library for each service was included in this analysis, Netflix would be number one in torrents, but on average it seems to suffer less than the other services. Its age could be the reason why it suffers less than others. Since it was the first popular streaming service, it was the one most people owned, removing most people's need to pirate its exclusive content.

The control group had about the same number of downloaded torrents as any individual streaming service, meaning that exclusivity may have less to do with torrenting numbers than initially assumed. Something that was not considered during the research was how internationally available some of these non-exclusives were. If they were only available through cable TV channels that did not run internationally, then it makes sense that torrenting numbers would be high.

The numbers for international availability vs region locked content seems to imply there is not a strong correlation between limited regional availability and torrenting. However, much of this hinges one how Disney+ was categorized. The service and its torrent numbers were placed in the "internationally available" category, but that was not only true. There was around a six month period of time when the service was released where it was region locked. That is about one fifth of the service's total lifetime. More research may be required to tell if this lack of correlation is accurate.

The pre-covid and post-covid numbers were not surprising. One average, torrenting numbers nearly tripled on a yearly basis after the lockdown began in March 2020. Many people were laid off work and locked inside with not much to do, so all forms of entertainment most likely skyrocketed.

Overall, most of the results differed from expectations. Although, these results may benefit from more research.

V. ISSUES

The majority of this project ran without running into any major problems, but there were a few to make note of. The first and most impactful issue involved the difficulty of acquiring accurate piracy statistics, especially on a global scale. This data is hard to acquire without outright paying for it, or somehow gathering it yourself. The difficulty of this task led to the re-evaluation of the original research objective and coming up with a much more feasible one. Specifically, one whose data could be more easily gathered by an independent team.

The next two issues encountered both involved processing the data that had already gathered. The first issue involved an error in how Octoparse exported the scraped data into JSON files. There was either some compatibility issue between their JSON format and the one that python parser, or just an actual error in how they formatted their JSON files. To fix this, some of the JSON files had to be manually corrected to be

in a proper format recognizable by the python JSON library. The final issue encountered was that the scraper had no way of differentiating between torrents that shared a similar name to the shows that were in the data set, but were actually torrents for completely different things. To fix this, specific torrents' names that generated the most interference were identified, and a list of phrases associated with those torrents was created in order to filter out the irrelevant results. The parser ignores data entries which contain these blacklisted phrases.

VI. FUTURE WORK

A possibility to further improve the data is to more thoroughly comb through the data collected and throw out redundant and irrelevant data.The data cleanup is done by throwing out other media that was included in the dataset when the data is collected. The way the irrelevant data is detected is by going through the data manually and finding words that are exclusive to the irrelevant data then run the words through a parser and remove the datasets that contain the chosen words. The dataset could also be improved by choosing more relevant shows. The chosen shows in the data could have been all available internationally. For example some shows are only available in the US and therefore for other consumers in other countries, the only way to watch the show is to pirate. The international nonexclusive shows will show the pirating more accurately in that consumers would have been given a choice of pirating or going the official route.

VII. ORG CHART

Date	Activity	Goals
Oct 20	Start Sprint 1	o Get the shows needed o Start parser
Oct 27	Start Sprint 2	o JSON files o Work on parser
Nov 3	Start Sprint 3	o Cleaning data o Work on parser
Nov 10	Start Sprint 4	o Finish parser
Dec 01	Final meeting	o Finalize project o Report

At the start of each there was a meeting. After sprint 4, meetings were changed to twice a week. The parser was started during sprint 1 and was finished during sprint 4.

Andrew

- o Project Manager ensuring the timeline is kept
- o Data Collection using octoparse

Riley

- o Created Parser using python

Franklin

- o Data Collection using octoparse
- o Data Cleaning making a library of words to omit

Anthony:

- o Data Collection using octoparse