**Group Members**

|  |  |  |  |
| --- | --- | --- | --- |
| S Number | Name | Contact Info | Signature |
| S5097965 | Timothy Chew | C:\Users\aytimothy\Documents\facebook-messenger.png aytimothy  C:\Users\aytimothy\Documents\discord.png aytimothy#0919 |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Fandoms Connect

**3804ICT Data Mining**

A data-mining investigation into how popular culture fandoms are interconnected, spread and related.

**Project Proposal & Investigation Report**

S5097965 FINAL 3804ICT T1/2019

# Preface

Thanks to the internet and the rise of internet message boards to the wider public such as Reddit, 4chan and not Newsgroups, the past 5 years have never been a better time for fans of pop-culture and interest groups to connect with one another. However, despite the internet being interconnected and everything out in the open, there is still the tendency for groups of people to clump up or join the same set of groups (which is still clumping up).

The aim of the investigation is to see what percentage of users belong to which fandoms, and how different fandoms overlap or avoid each other. For example: Do *Digimon* fans also interact with those who also like *Pokémon*, and vice versa[1]?

# Data Collection Methods

To get our data, I used the Python library praw to crawl Reddit using the Reddit API[2] for users, submissions and comments. These are the main building blocks of the data we are going to be analysing. It’s not 10 dimensions, but it’s already at least 8 and can be broken down into more through text analysis (more on that later). This gives us a real-world dataset straight from a live social media website as its source.

As a beginning point, I scraped myself ([/u/aytimothy](https://reddit.com/u/aytimothy)) looking at the interactions that I made on Reddit. Because I’m already part of the fandoms and communities that I want to investigate, and have interacted with a lot of people, this serves as a great starting point for building our “network” and database of users and content posted.

After scraping myself, I would look at all posts and comments regarding a particular meta-topic (anything from a subreddit that is within scope, like /r/gaming, /r/digimon, /r/anime, etc.) and scrape all users and comments who have interacted with it. Using that list of users, we would then build a network of interactions and work our way out scraping their profiles blindly. Once this queue is exhausted, we would repeat for each individual user. This exponentially grows our database of comments and cloud of users.

Many users are known to interact elsewhere, so dead-ends was not a problem. For building the extra dimensions, we do this in the data analysis, and any other data points can simply be queried from Reddit as we need them.

**In the repository, this is collect.py and collect2.py found in the /collection folder.**

# Data Analysis Methods

Using techniques taught in class, we can apply various techniques to analyse the data. I want to find out the following questions:

* If person X interacted with subreddit Y, how likely is he to interact with/knows about subreddit Z?
* If person X interacted with subreddit Y, what communities would he also interact/know about with (a set)?
* What are the ratios of people overlapping between subreddit X and subreddit Y, for all related subreddits collected?

Firstly, the term “interacted” is a bit subjective, so the first step is to clean up the data. We can very easily do this by simply finding a list of subreddits each particular person has interacted or referenced. Interaction or referenced can be inferred by simply looking for a post/comment action, being mentioned somewhere or mentioning a set of keywords related to it.

From there, we can work out the likelihood by using the Aprori Algorithm to work out common sets from the list of subreddits a user has ‘interacted’ or ‘referenced’.

The same data can then be clustered using kNN and correlation-based distance calculation between each node. Alternatively, a neural network of subreddits as input and outputs may also achieve the same effect.

From there, the outputs from the clustering analysis and classifications can be used to calculate the overlaps; the last question is simply an aggregation of the previous data.

# Preliminary Data Analysis

**Note: The data is currently being scraped, and this information does not reflect the whole dataset.  
The information is accurate as of the 6th of August, 9:01 PM. Since this is a continuous mining operation, some numbers or facts may change while this is written.**

* 635,332 comments and 6,989 submissions have been processed from Reddit, after approximately 60 hours of runtime.
* We have processed the entire post/comment history of 82 users.
* There are 340,817 unique users for comments and 3,312 for submissions in the database.
* There are 1,325 unique communities between all comments, and 1,473 between all submissions.
* There are 394 submissions and 31,985 comments created by deleted users.
* There were 3,397 comments and 311 submissions removed by moderators.
* There were 9,826 comments and 302 submissions deleted by their authors.

Submissions and comments are distributed as follows:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Property | Min | Q1 | Median | Q3 | Max | Mean |
| Submission Score | 0 | 3 | 35 | 466 - 467 | 153793 | 3139.2162 |
| Comment Score | -532 | 1 | 4 | 17 | 58234 | 116.9329 |

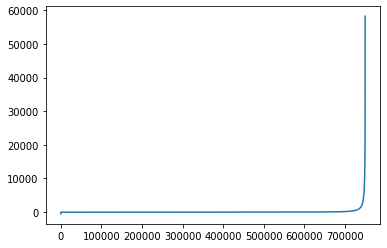
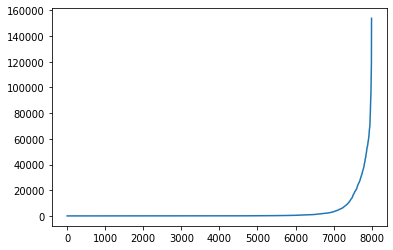
 

Figure 1 Comment Scores are very skewed. Figure 2 Submission Scores are even more skewed.

Minimums are maximums are skewed while the medians still trend in positive numbers as only the outliers generate high/low scores. Most comments/submissions get few or little upvotes.

Finally, a few more interesting information about the content that was scraped:

* The most up-voted submission is a picture of a cat from [/r/aww](https://reddit.com/r/aww) with a score of 153,793[3].
* A [Rick Roll](https://youtu.be/dQw4w9WgXcQ); Rick Astley’s song *Never Gonna Give You Up* was the fifth [highest scoring post](https://www.reddit.com/r/videos/comments/5gafop/rick_astley_never_gonna_give_you_up_sped_up_every/) collected.
* [/u/SovietRussiaBot](https://reddit.com/u/SovietRussiaBot) holds the 1st, 3rd and 5th most down-voted comment. It's a [bot](https://github.com/dneu/SovietRussiaBot) that replies to anything it considers a [Russian Reversal](https://en.wikipedia.org/wiki/Russian_reversal) joke, and also uses praw.
* Somehow, despite the whole gaming community knowing about it, EA Games’ ([/u/EACommunityTeam](https://reddit.com/u/EACommunityTeam)) [most downvoted comment](https://www.reddit.com/r/StarWarsBattlefront/comments/7cff0b/seriously_i_paid_80_to_have_vader_locked/dppum98/) (and also most downvoted across *all* of Reddit) is nowhere to be found. (7cff0b/dppum98)
* Due to recent controversies, Stars Wars Battlefront and the Hong Kong riots has appeared in the top list of comments. (Edit added on 27/08/2019)
* The [top comment](https://www.reddit.com/r/AskReddit/comments/bu1s5i/what_fact_is_common_knowledge_to_people_who_work/ep6aqhz/) is about elevators when they have a catastrophic failure.

These aren’t the highest or the lowest of the global lows, but just of the data set; sample of Reddit.

## Data Quality, Completeness and Noise

Because the data collected is *everything and anything*, there are bound to be useless or irrelevant, such as my post to [sell a Westfield Giftcard](https://www.reddit.com/r/giftcardexchange/comments/cep1a8/h_westfield_gift_card_470_aud_w_offer_paypal_cash/). Meta subreddits (such as /r/gaming) are good for diversifying your data and finding starting points, but also comes with a lot of fluff. Furthermore, as stated above, there are also posts by deleted users or were removed due to moderation reasons.

Next, it does not meet the minimum of 10 dimensions, which is possible to expand on by checking the text. References to other fandoms don’t own occur in a user submitting a post or commenting in one, but also by reference. For example, for pet monster franchises (ie. Pokémon, Tamagotchi or Digimon), we can search the bodies of text for references to their subreddit (anything beginning in an “/r/” and applies universally) or actual monster names. However, for clean usable data, it can be de-dimensionalsed; simplified to around 5 as the others are just groupings/metadata.

[1]: Original Research: Older audiences of both franchises (the 1999 to 2003 demographic who were ages 6 to 18 at the time, or parents) are aware of each other due to the virtual pet and early toy gatcha market. This later spread down the line towards the current younger generation (these people are now in their late teens to adulthood as of 2019). Also, the answer is: yes; they do.

[2]: <https://www.reddit.com/wiki/api>

[3]: There are still higher scores on Reddit. This is just a local maxima.

Fandoms Connect

**3804ICT Data Mining**

A data-mining investigation into how popular culture fandoms are interconnected, spread and related.

**Findings and Aftermath Report**

S5097965 DRAFT 3804ICT T1/2019

# Solution

## Methodology

We can find common groupings using the Apriori algorithm and subsetting our data by which stem subreddit to look at to learn about the different userbases.

Turns out that a neural network is not required as you can work out the probability by finding the fraction of userbase A who participates in userbase B;

*and are the two subreddits we want to compare.*

However, accuracy boiled down to having a big enough dataset.

## Implementation

I implemented it using scratch (library-less) Python and mlxtend (also Python) as introduced to in the labs.

Weka crashed because of a too big dataset, and scratch implementation did not wrong as it was written in pure python, and loops and set operations are extremely inefficient (especially with big datasets).

|  |  |  |  |
| --- | --- | --- | --- |
| **Full Dataset Analysis** | | | |
| **Implementation** | mlxtend (Python) | Pure Python | Weka |
| **Time** | 7509.783921 | Crashed\* | Crashed\* |
| **/r/Minecraft ???** | | | |
| **Implementation** | mlxtend (Python) | Pure Python | Weka |
| **Time** | 2144.892012 | Crashed\* | Crashed\* |
| **/r/pokemon ???** | | | |
| **Implementation** | mlxtend (Python) | Pure Python | Weka |
| **Time** | 1892.892301 | Crashed\* | Crashed\* |
| **/r/digimon ???** | | | |
| **Implementation** | mlxtend (Python) | Pure Python | Weka |
| **Time** | 1322.555212 | Crashed\* | Crashed\* |

\*We ran out of memory or was killed by the system for killing (overloading) the CPU

* **framework.py** is the implementation using mlxtend.
* **scratch.py** is the implementation using vanilla python. This implementation is extremely inefficient.
* **weka.py** opens Weka.

# Results

**You can see the results in the file output. (No file extension)**

## Overall Dataset

Due to the immense amounts of subreddits and the wide range of people and an extremely small dataset, we used a minimum support of 0.05 (approximately 400 people) and determined that, the subreddits that most are from are (this is the top of the top):

|  |  |
| --- | --- |
| **Support** | **Subreddit** |
| 0.516343 | AskReddit |
| 0.389009 | gaming |
| 0.310165 | pics |
| 0.309447 | funny |
| 0.272091 | todayilearned |
| 0.252335 | videos |
| 0.194864 | Gifs |
| 0.177622 | WTF |
| 0.171875 | Games |
| 0.165948 | worldnews |

From there, we can see associations:

|  |  |
| --- | --- |
| **Support** | **Subreddits** |
| 0.221624 | (AskReddit, gaming) |
| 0.181573 | (AskReddit, pics) |
| 0.180316 | (AskReddit, funny) |
| 0.158226 | (AskReddit, videos) |
| 0.157687 | (AskReddit, todayilearned) |
| 0.151401 | (gaming, funny) |
| 0.138290 | (gaming, pics) |
| 0.133082 | (pics, funny) |
| 0.125180 | (gaming, videos) |
| 0.108657 | (gaming, todayilearned) |

Most of the people on /r/AskReddit are also gamers, and also browse the more casual /r/pics, /r/funny, /r/videos and informational /r/todayilearned for random facts.

## /r/Minecraft ???

Since the datasets here is smaller, and we are looking at a subset, we can use higher minimum supports to find significant things. I used 0.3 as the limit to look for nicher (which will be shown later) subreddits due to how skewed to the meta-subreddits (such as /r/gaming, /r/AskReddit; the things that have high supports in the overall dataset).

|  |  |
| --- | --- |
| **Support** | **Subreddits** |
| 0.606061 | (gaming, Minecraft) |
| 0.598485 | (gaming, AskReddit) |
| 0.409091 | (AskReddit, Minecraft) |
| 0.409191 | (AskReddit, gaming) |
| 0.409091 | (gaming, Minecraft, AskReddit) |
| 0.363636 | (funny, Minecraft) |
| 0.340909 | (Minecraft, pics) |

Many of Minecraft participants are also on /r/gaming (because they’re gamers) and /r/AskReddit.

## /r/pokemon ???

|  |  |
| --- | --- |
| **Support** | **Subreddits** |
| 0.563981 | (AskReddit, pokemon) |
| 0.535545 | (gaming, pokemon) |
| 0.331754 | (funny, pokemon) |
| 0.327014 | (AskReddit, gaming) |
| 0.327014 | (AskReddit, gaming, pokemon) |

Of course many are gamers, and asks on Reddit.

## /r/digimon ???

|  |  |
| --- | --- |
| **Support** | **Subreddits** |
| 0.4 | (AskReddit, digimon) |
| 0.35 | (digimon, funny) |
| 0.3 | (digimon, gaming) |
| 0.3 | (digimon, pokemon) |

And as suspected, Digimon fans also know about Pokémon; another monster-collecting franchise, but unfortunately, not the other way around.

## Summary

Default Subreddits are subreddits that many (almost everyone) participate in and will show up everywhere. These include:

* /r/AskReddit – A subreddit where users ask the internet any question.
* /r/gaming – A subreddit all about gaming (primarily video games, but permits all, so cards too)
* /r/pics – A subreddit for posting anything that is a picture.
* /r/funny – A subreddit for posting funny things, in any medium or form.
* /r/todayilearned – A subreddit for posting interesting niche facts you’ve learnt (today).
* /r/videos - /r/pics, but for video files.
* /r/gifs - /r/videos but for anything in the .gif format.

And, there is a correlation (one person likes both) between Digimon fans and Pokémon fans, but not the other way around.

# Reflection

I could not explore a lot as the data is full of unexplored nodes, as most of my time was spent mining the major default subreddits (primarily /r/anime and /r/gaming). If I had did the Apriori Algorithm earlier on and looked at the smaller Subreddits, I would have gotten a pointer towards more specialized user behaviour quicker.

Instead, by looking at the major subreddits, I just got a lot of everything, which especially for the /r/digimon correlation – Had a very small support rate from everything else, but then again, comparing a specialization to a bit of everything still results in small support values anyway as everybody is different and not a single one of them are the same. From the decimal places, you could tell that the subset had only around 20 people/transactions in it.