Shane Cincotta
Hui Fang
CPEG 457
03/30/2020

Reddit Scraper Proposal

For my project I will create a script that continuously runs on a RasPi while scraping various subreddits on reddit for a user selected input. Before I describe how exactly my scraper will function, it's best to give a quick overview of how Reddit works.

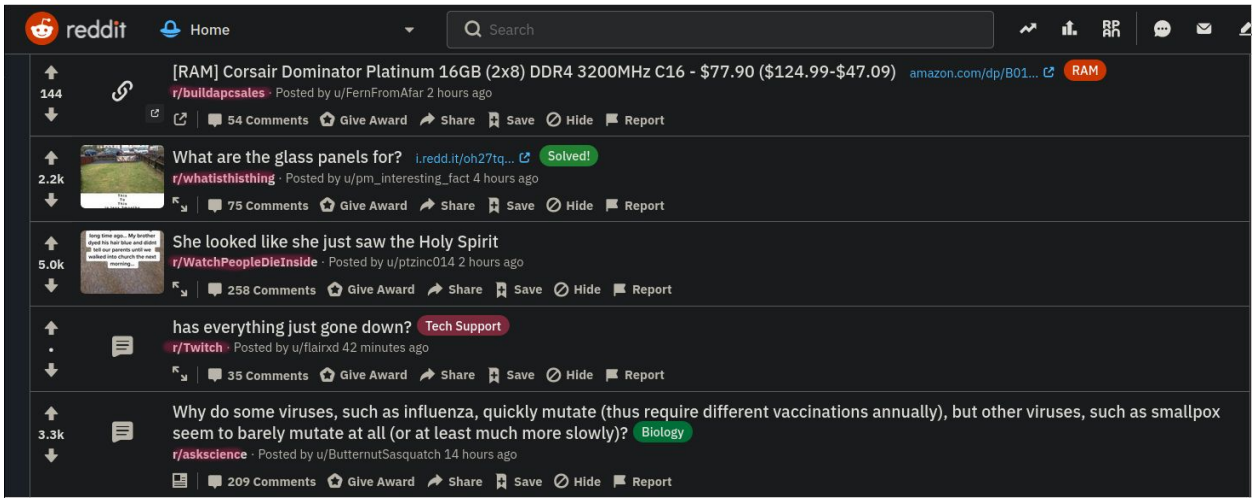

Fig 1. Reddit Front Page

Reddit is essentially a collection of forums. Forums are grouped together by category, these groupings are called "subreddits". Each Reddit post is made and contained within a subreddit, if a post within a subreddit gains enough popularity, the post will appear on my homepage. In figure 1, I show my front page with some popular posts from the day. I have also highlighted the subreddit they were originally posted in. From the figure you can see that there is a post from the subreddit "askscience", a subreddit where people ask science related questions, there's also a post from "buildapcsales", a subreddit where people post links to sales on computer parts. Reading the title of the post from buildapcsales, we can see that this post is a sale for 16GB of RAM. Let's do some more investigation into Reddit and click on the buildapcsales post.

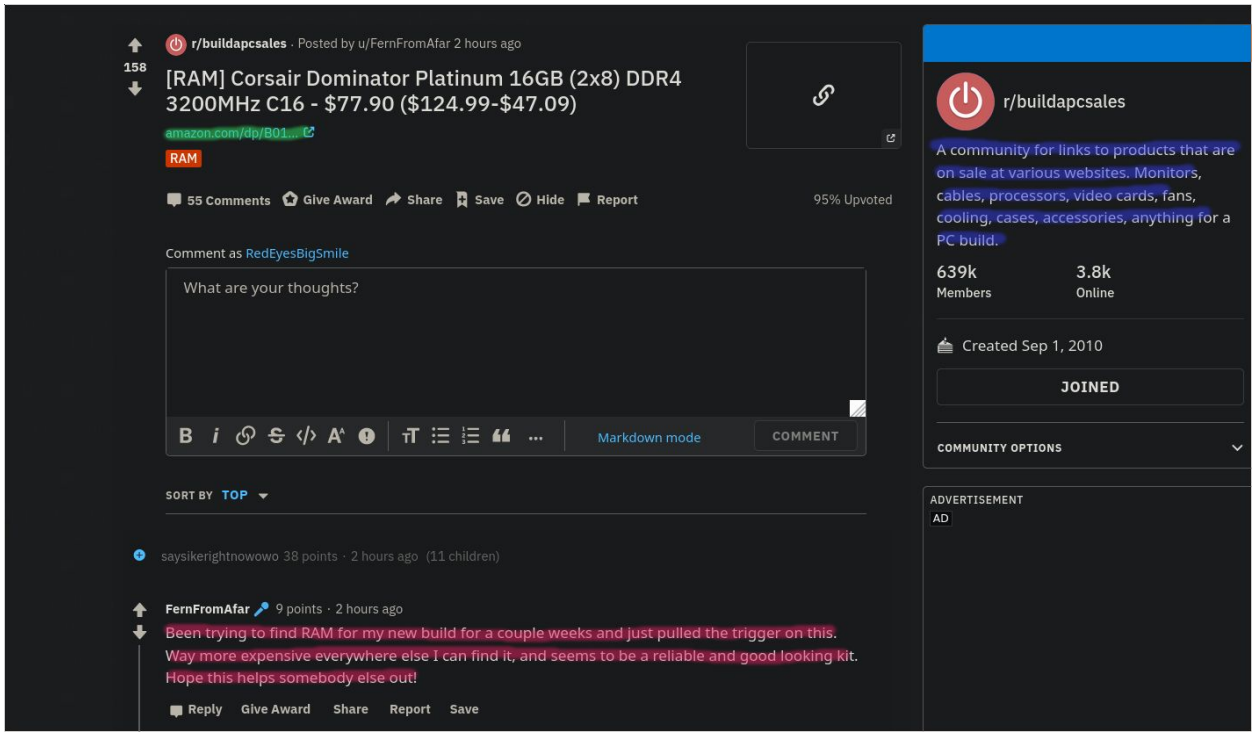

Figure 2. Buildapcsales Post

Figure 2 shows the buildapcsales post after I click on it. You can see that more information is shown to me (I have highlighted various new info). Highlighted in green is the link to the RAM sale, in blue is a description of the subreddit, and in pink is a comment from another user. In this post, there is only a title, but it is possible that there is also a body of text included in a post, that information would also be shown here.

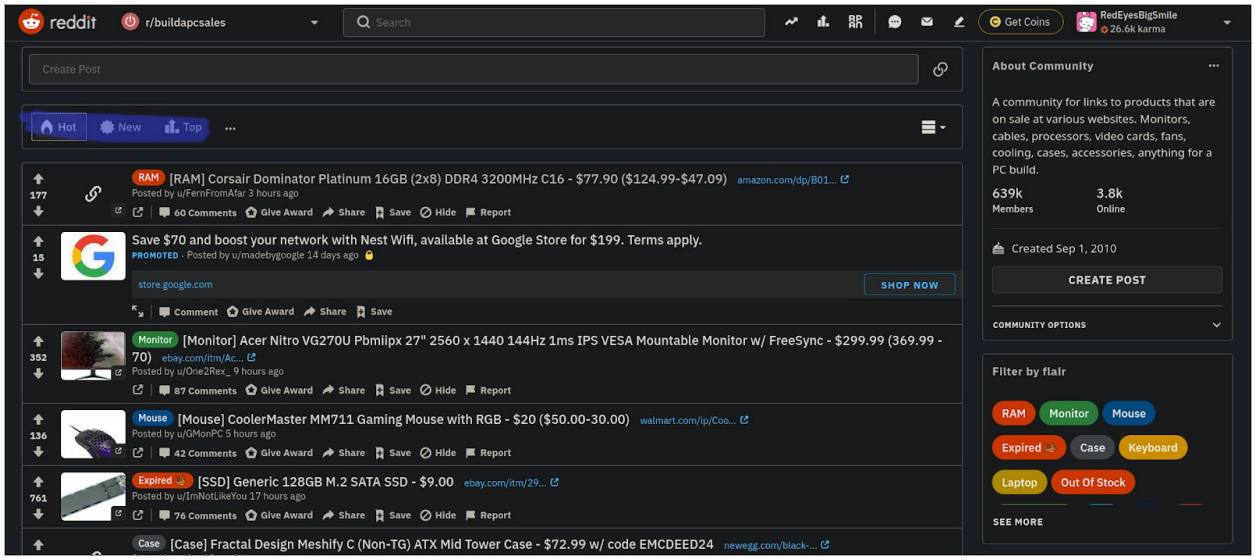

Fig 3. Viewing a specific subreddit

Finally on our tour of reddit, we can also view individual subreddits. Figure 3 shows my screen after I click on the buildapcsales subreddit link. This brings me to the buildapcsales subreddit home page. Here I can see all the posts that were posted on buildapcsales (no other subreddits will be included, unlike the homepage). You can see that I have some more options as well, highlighted in blue are options for sorting how I see the posts. I can sort them by date posted, or by the amount of upvotes (upvotes are comparable to “likes” on Facebook).

Now that we have an understanding of how Reddit works, we can start discussing my scraping tool. My scraper will search a user selected subreddit for a user selected search token. Let’s use the buildapcsales subreddit as an example. In figure 3, we can see various titles of posts from buildapcsales, there’s something interesting about their titles. All of the titles in buildapcsales are organized in the same way, the first word of the title is the type of hardware that the sale refers to. We can see various examples of this such as “[SSD]” and [“Mouse”]. What follows this is the full product name, and then the price of the item. It is actually a subreddit rule for buildapcsales that every post must be organized in this way. This makes this subreddit a great candidate to scrape from. This is because we always know that the first word is going to be the type of item, so if I am scraping for a specific set of RAM, I only need to look at the first word in the title to determine if the post is about the topic I am scraping for.

Now the organizational requirements of buildapcsales is great, but not every subreddit has those same post requirements or even requirements at all. Let's take a look at another subreddit which again has post requirements, but the requirements are different than buildapcsales.

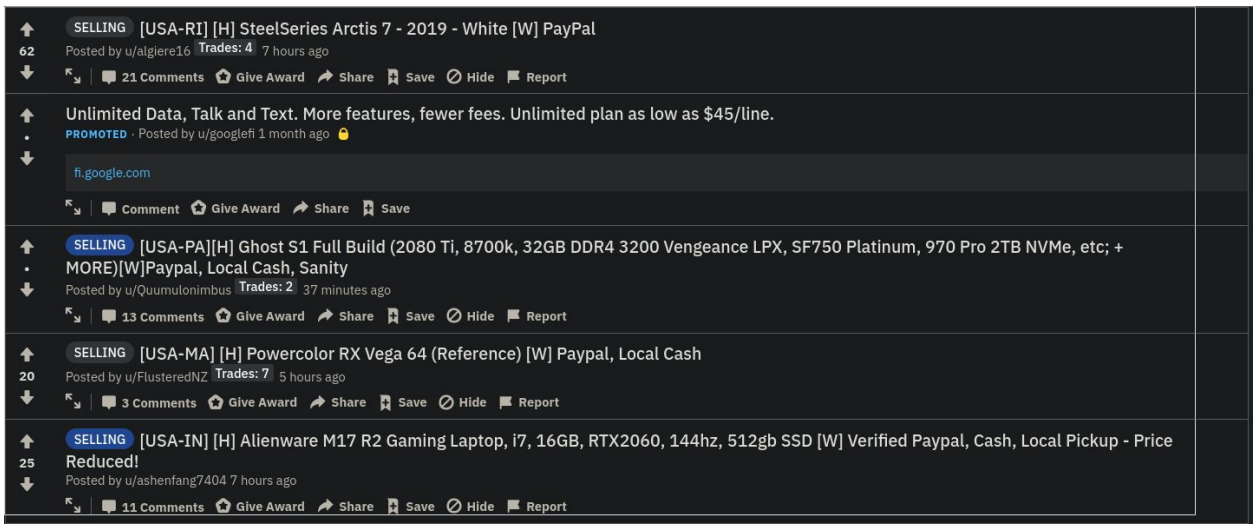

Fig 4. Hardwareswap Subreddit

Figure 4 shows the subreddit hardwareswap. Hardwareswap is a subreddit where people can buy, sell and trade various hardware. We can see from figure 4 that all the titles are again organized in the same way. Specifically, per the rules of hardwareswap, the first word of the title must include your country and state. The second word must be either [H] (which stands for “have”), which refers to which item you have for sale/trade. The third word(s) must be the item you’re buying, selling or trading. The fourth word must be [W] (which stands for “want”), which refers to what you want in return for your item that you [H]. Finally, the last words which come after [W] refer to what you want in return. Let’s look at the first post. We can see that the poster is from the USA, specifically Rhode Island. We can also see that the user has a Steel Series Arctis 7, and the user wants to receive Paypal in return for the item.

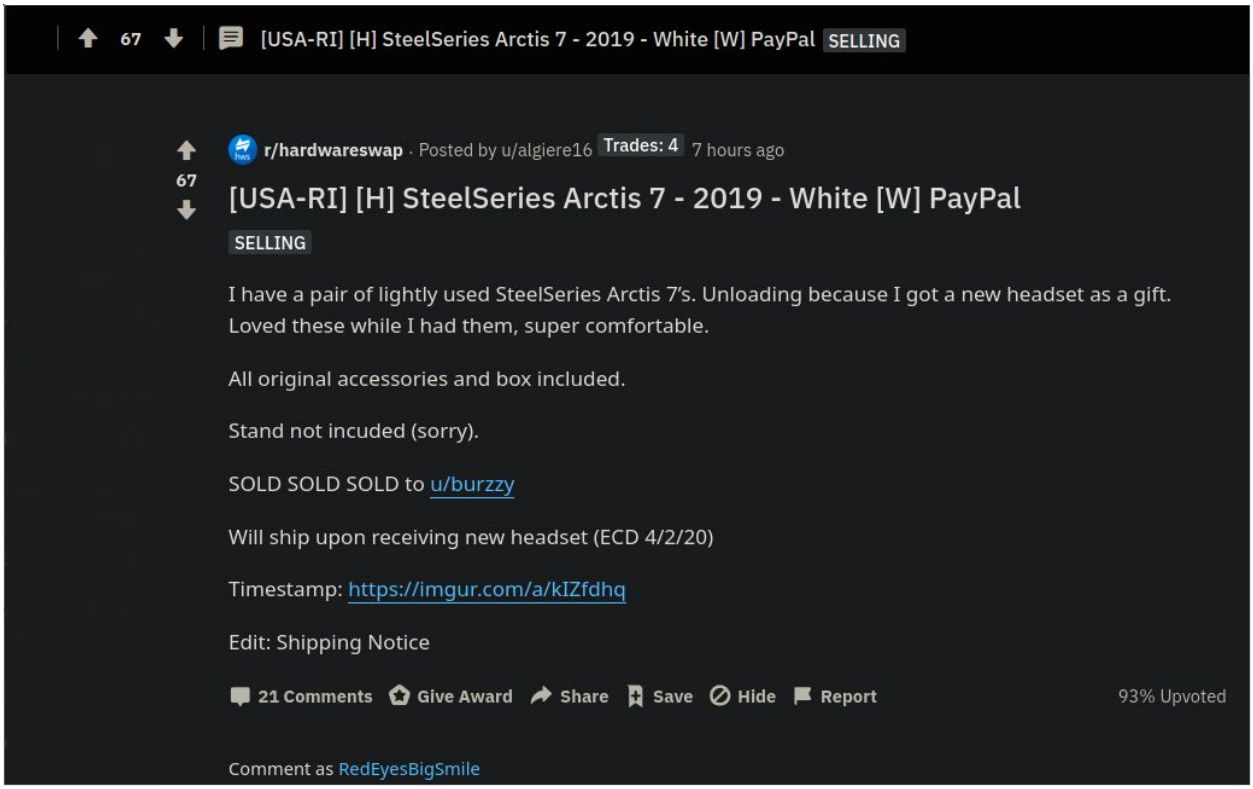

Fig 5. Hardwareswap post

Figure 5 shows what is displayed when I click on the post described above. We can see that a body is now displayed with new information. We can see that there is no price listed, as well as the date that the item was sold on. This gives my scraping bot more information to work with, it can now find if the item was sold and a description in the body.

Since some subreddits have specific rules for how posts are titled (most subreddits don't have rules at all), I would need to create a separate algorithm for each subreddit that has rules. My plan is to have a general search algorithm which will work for any subreddit, I will also have various algorithms that are subreddit specific. For example, if the user wants to scrape hardwareswap for an SSD, I know that hardwareswap has rules for how the post title is organized, so my algorithm will be tuned to work within the hardwareswap rules. If the user wants to instead search for an SSD within buildapcsales, my algorithm will use a search function specifically made to fit within the rules of buildapcsales. There is also the possibility that the subreddit we're searching doesn't have rules for how the title is organized. In that case, I would use a search algorithm that works for any subreddit which doesn't require rules in the post. In that case, the amount of information I would get from a post would be more general.

Once the scraper finds a post hit, it will record the post data depending on the subreddit. For example, if I am scraping buildapcsales, I know the price of the item is required to be in the title, while the price is not necessary to be included if I was scraping hardwareswap. So in the case of buildapcsales, I would record the price, while in the case of hardwareswap, I might not record the price if it doesn't exist. Once I have all the post data, I will use an SMS texting API (Twilio) to send all the post information to my phone. In addition, reddit includes an API called PRAW, which allows me to download realtime post information from Reddit.

The actual algorithm to determine a hit will be different for each subreddit, but will be based off information such as the item being searched for (is it the same item? If not, how close is it?), the price of the item (is there a price? If there is, what is the price? Is it lower than the maximum price I want to pay?), the time of the post (how recently was it posted?) etc.

Now there does exist various tools provided by reddit to search within subreddits for a specific search token(s), but there are two major downsides. First, the built-in Reddit search algorithm is really, REALLY bad. It's often better to use google to find a Reddit post than using Reddit's search functionality. For example, if I want to find a post about cats, it's better to use Google and search "Cats Reddit", than it is to search for "Cats" on Reddit's website. Thus I think I can improve upon the search functionality within Reddit. Second, even if Reddit's search functionality worked as intended, it doesn't continuously update. That is, I would have to manually search and refresh a subreddit to see if any new posts regarding a certain topic were posted. My bot would do the "refreshing" for me. Alot of my friends use these subreddits listed and often complain about the search functionality provided by Reddit, granted this is anecdotal,

but I believe that if they had access to my tool, they wouldn't have these issues anymore. The main people who would benefit from this tool are people who use Reddit often to buy and sell products.

To test if my scraper has worked or not, I will observe the data sent to me via SMS, and then pull up the actual Reddit post the SMS is referring to and see if the data matches up, that is, given my criteria for determining if a post is a hit (time post, item being offered, price of item, etc) should that post have been sent to me? Is the price correct? Is the item correct? Etc.

Rough timeline of what I would like to have completed and when

Since I am working on this project by myself, I will be doing all the work.

- 04/03
  - Implement PRAW to pull subreddit data and familiarize myself with the given methods
- 04/07
  - Implement first search function for one subreddit
- 04/10
  - Finish specific search functions for the rest of the subs I want
- 04/13
  - Finish a general search function which is used for subreddits without rules
- 04/17
  - Debugging time
- 04/20
  - Deploy on RasPi