CIS5200 Term Project Tutorial

Authors: Yash Choksi, Sweta Tripathi, Karolina Stachurska, Sowmya Mareedu, Sindhura Vemuri

Instructor: Jongwook Woo

Date: 05/02/2020

Lab Tutorial

Yash Choksi (ychoksi@calstatela.edu)

Sweta Tripathi (stripat@calstatela.edu)

Karolina Stachurska(kstachu@calstatela.edu)

Sindhura Vemuri (svemuri@calstatela.edu)

Sowmya Mareedu (smareed@calstatela.edu)

05/02/2020

Reddit comments analysis

Platform Spec

- Required space: 25 GB
- CPU Memory size: 2.20 GHz
- # Nodes: 3 nodes
- Hadoop cluster

Step 1: Get data from Kaggle

1. Found dataset from Kaggle.com
2. Dataset link: https://www.kaggle.com/reddit/reddit-comments-may-2015

3. Original size of data set is 30 GB, because of compression they reduced it to 16 GB. Dataset format is sql table so sqlite. So converted data from sqlite to csv using sql workbench. Data set has more than 15 columns and 21199015 rows.

4. Uploded data to Hadoop cluster using scp command: scp ./May2015.csv ychoksi@129.150.71.75:~/

5. Uploded to hdfs using put command.

6. Created external hive table and get data from csv file:

```
CREATE EXTERNAL TABLE if not exists lookup ( created_utc BIGINT,ups BIGINT,subreddit_id String, link_id String,name String, score_hidden BIGINT, author_flair_css_class String, author_flair_text String, subreddit String, id String ,removal_reason String,gilded BIGINT,downs BIGINT, archived BIGINT,author String,score BIGINT,retrieved_on BIGINT,body String, distinguished String,edited BIGINT,controversiality BIGINT,parent_id String )
ROW FORMAT DELIMITED FIELDS TERMINATED BY ','
STORED AS TEXTFILE LOCATION '/user/ychoksi/check/';
```

```
i1-bdcsce-1.compute-608214094.oraclecloud.internal:10002 failed to respond (state=08S01,code=0)
0: jdbc:hive2://bigdai1-bdcsce-1:2181,bigdai1> CREATE EXTERNAL TABLE if not exists lookup ( created_ut
c BIGINT,ups BIGINT,subreddit_id String, link_id String,name String, score_hidden BIGINT, author_flair
_css_class String, author_flair_text String, subreddit String, id String ,removal_reason String,gilded
 BIGINT,downs BIGINT, archived BIGINT,author String,score BIGINT,retrieved_on BIGINT,body String, dist
inguished String,edited BIGINT,controversiality BIGINT,parent_id String )
0: jdbc:hive2://bigdai1-bdcsce-1:2181,bigdai1> ROW FORMAT DELIMITED FIELDS TERMINATED BY ','
0: jdbc:hive2://bigdai1-bdcsce-1:2181,bigdai1> STORED AS TEXTFILE LOCATION '/user/ychoksi/check/';
No rows affected (0.176 seconds)
0: jdbc:hive2://bigdai1-bdcsce-1:2181,bigdai1>
```

7. As the table is created we can check size of database: select count(*) from lookup;

```
+-----------+--+
|    _c0    |
+-----------+--+
| 21199015  |
+-----------+--+
```

8. All column name with data types: describe lookup;

col_name	data_type	comment
created_utc	bigint	
ups	bigint	
subreddit_id	string	
link_id	string	
name	string	
score_hidden	bigint	
author_flair_css_class	string	
author_flair_text	string	
subreddit	string	
id	string	
removal_reason	string	
gilded	bigint	
downs	bigint	
archived	bigint	
author	string	
score	bigint	
retrieved_on	bigint	
body	string	
distinguished	string	
edited	bigint	
controversiality	bigint	
parent_id	string	

9. Two most important columns are author and name. If you go and check structure of reddit you can see that author is the person who asked the actual question, and name means person who made the comments. Now, to ensure privacy they **anonymized** data of name column. But they kept original form of author data.

As part of data cleaning we are removing all rows which do not have any value for name:

We are creating new table where author name NULL will be removed, table name is lookup:
Create table ltable like lookup;

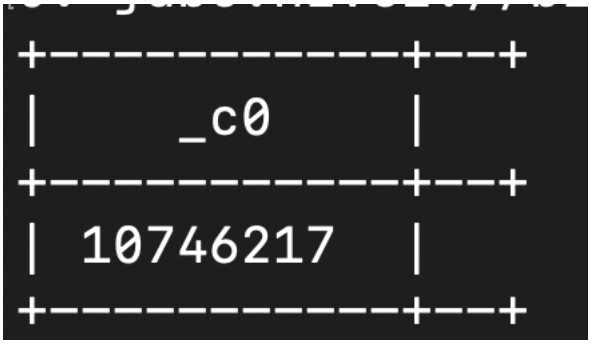
```
0: jdbc:hive2://bigdai1-bdcsce-1:2181,bigdai1> create table ltable like lookup;
No rows affected (0.362 seconds)
0: jdbc:hive2://bigdai1-bdcsce-1:2181,bigdai1>
```

tab_name
data
fcleaned
fulltable
l1
l2
l3
lookup
ltable

Now insert data in newly created table which do not have null values:

```
Insert into ltable from lookup where author is not null;
```

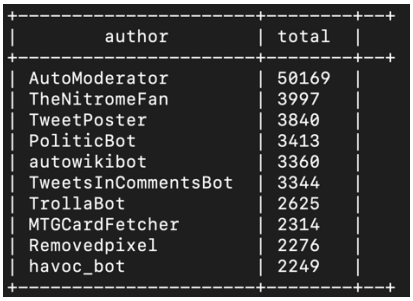
After data cleaning our number of rows will be:



_c0
10746217

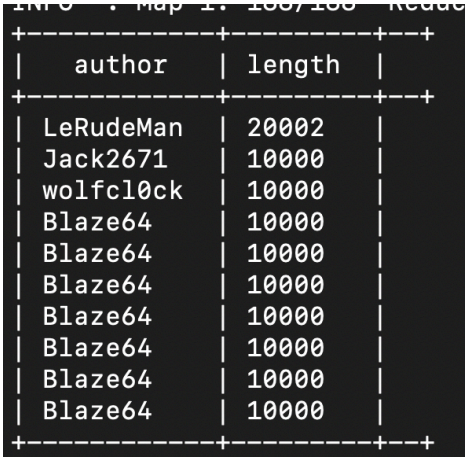
10. Top 10 authors who are asking most questions:

```
select author, count(*) as total from fcleaned where author != "[deleted]" and author != "" and author!="0" group by author order by total desc limit 10;
```



author	total
AutoModerator	50169
TheNitromeFan	3997
TweetPoster	3840
PoliticBot	3413
autowikibot	3360
TweetsInCommentsBot	3344
TrollaBot	2625
MTGCardFetcher	2314
Removedpixel	2276
havoc_bot	2249

11. Which authors asked longest questions so we can able to know who is asking how long questions:



author	length
LeRudeMan	20002
Jack2671	10000
wolfcl0ck	10000
Blaze64	10000
Blaze64	10000
Blaze64	10000
Blaze64	10000
Blaze64	10000
Blaze64	10000
Blaze64	10000

12. There is one column named as utc and which is universal time at which comment generated. Let's see top 10 times at which most of the comments are generated:

```
Select created_utc, count(*) as total from ltable group by created_utc order by total desc limit 10;
```

created_utc	total	
1430708335	94	
1430708334	90	
1430708385	89	
1430708506	88	
1430708389	83	
1430708503	83	
1430708497	79	
1430708505	75	
1430708502	74	
1430708498	73	

UTC means coordinated universal time
To convert from this strange looking time strings there is tool using programming you can convert to local time. Here is the link:
https://stackoverflow.com/questions/179940/convert-utc-gmt-time-to-local-time

13. Subreddit categories type of channels or community they want to target. Like sports or many others:

```
select subreddit, count(*) as total from fcleaned group by subreddit order by total desc limit 10;
```

subreddit	total	
AskReddit	819370	
leagueoflegends	208016	
pics	152373	
nfl	150304	
funny	146204	
nba	144457	
news	124577	
videos	110056	
todayilearned	106296	
DotA2	100526	

Let's explore more this community based channels:

14. Let's see top 10 authors who most asked questions related to nfl:

```
select author, count(subreddit) as nfl from fcleaned where subreddit = "nfl" and author != "[deleted]" group by author order by nfl desc limit 10;
```

author	nfl	
Banethoth	592	
Mister_Jay_Peg	491	
adv0589	452	
LoveRecklessly	414	
jrg114	368	
dean815	328	
Fuck-The-Modz	284	
sunstersun	281	
ImTheOnlyChipHere	271	
-Butt-Fumble-	255	

As we can see in subreddit channels Dota2 is very popular game so llet's see how many users comment about that game:

```
select count(*) as Dota2_count from fcleaned where subreddit = "DotA2";
```

dota2_count
100526

15. Top authors who received comments regrading dota2:

```
select author, count(subreddit) as dota2_count from fcleaned where subreddit = "DotA2" and
author != "[deleted]" group by author order by dota2_count desc limit 10;
```

author	dota2_count	
Dancatpro	718	
lolfail9001	368	
pankajsaraf880	349	
meme_shitter	335	
QKaraQ	293	
thegforce522	241	
ixmike88	230	
Fallen_Wings	213	
Aranys	202	
Hunkyy	200	

16. Relationship between controversy points and number of questions asked:

```
select author, SUM(Controversiality) as total_contro, COUNT(*) as sum from fcleaned where
subreddit="politics" group by author order by total_contro desc limit 100;
```

author	total_contro	sum	
fantasyfest	4291730205	138	
moving-target	2862387137	4	
ugots	2861873185	23	
ELaphamPeabody	2861863408	321	
Shnazzyone	2861695707	8	
Thorium233	2861555962	9	
jpurdy	2861454593	167	
hambonese	2861346864	4	
ridetherhombus	2861271435	27	
WildPepperoni	2861187765	40	

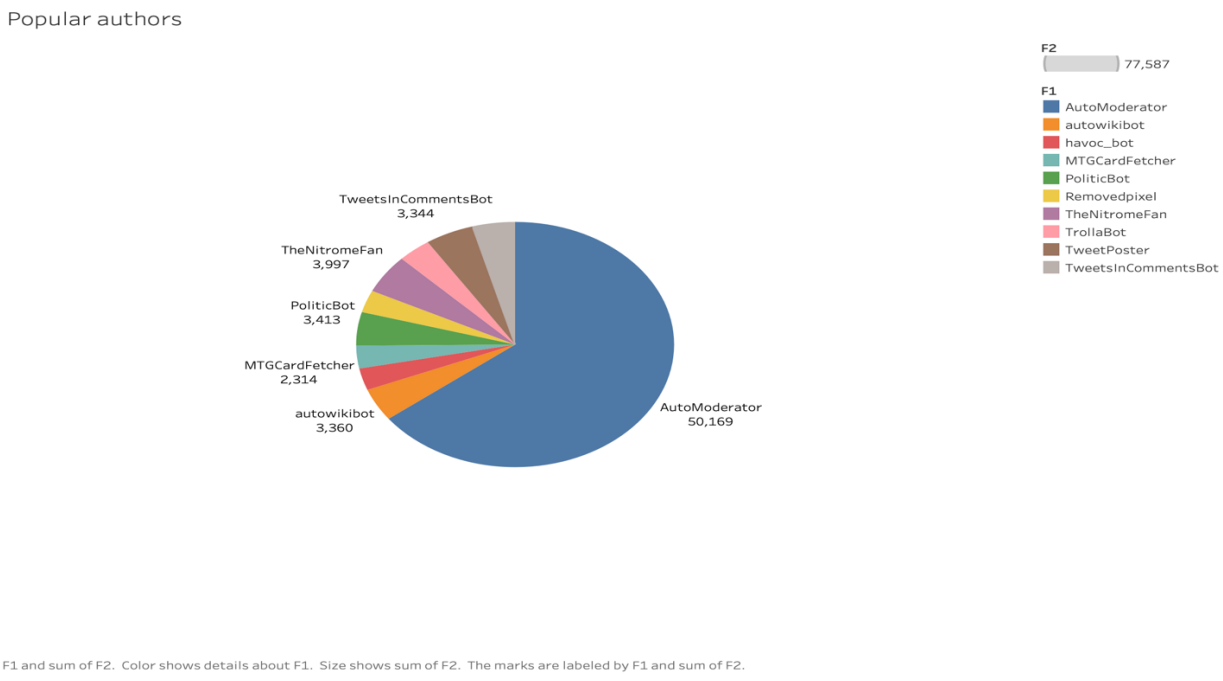
17. Edit rates to controversial posts:

```
select author, sum(controversiality) as contro, sum(edited) as edit from fcleaned where author!="0" and author!="[deleted]" and author != "" group by author order by contro desc limit 50;
```

To get this data efficiently we have converted whole different small tables and then downloaded with scp command. And by doing that we can download data and can perform visualization.

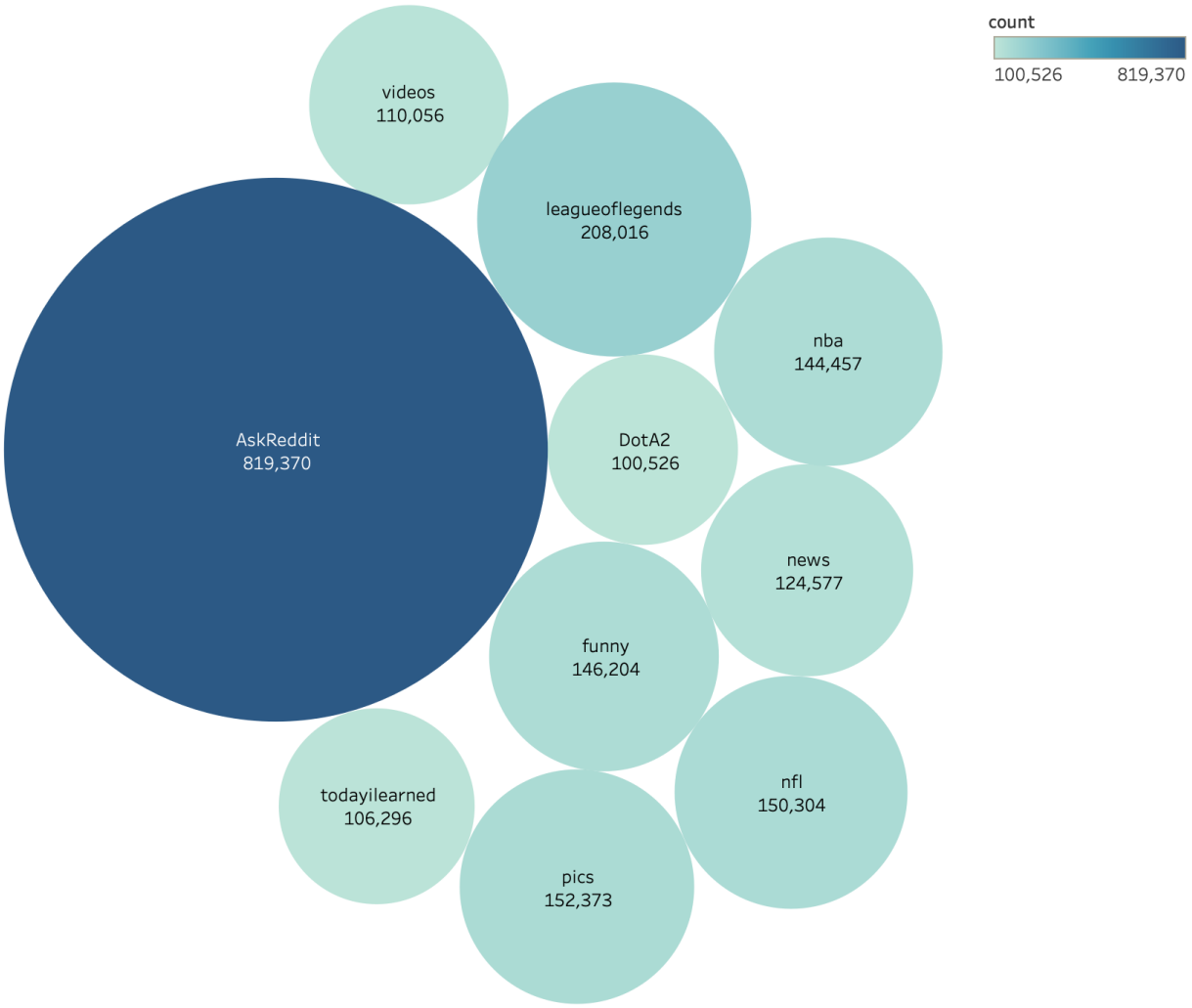
Visualizations:

We have used tableau for our visualization work.



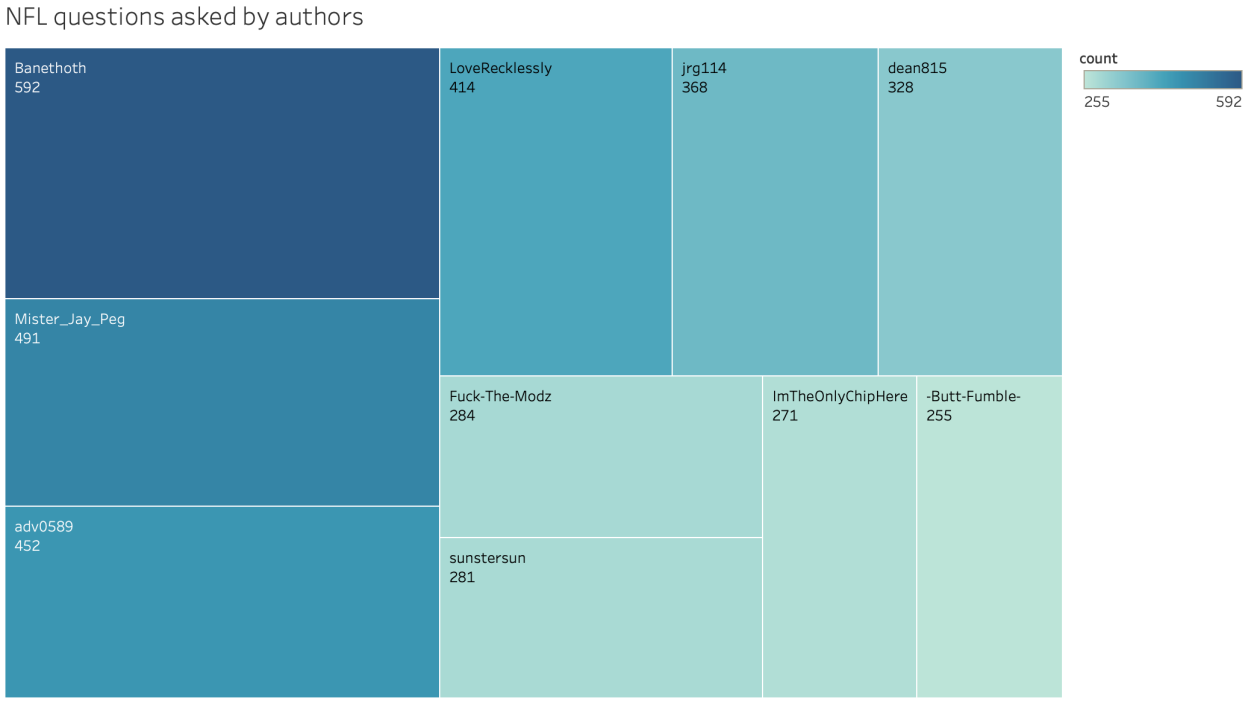
In the above figure the top 10 authors who asked most questions. The data is labelled with author name and how many questions he posted.

Categories or channels



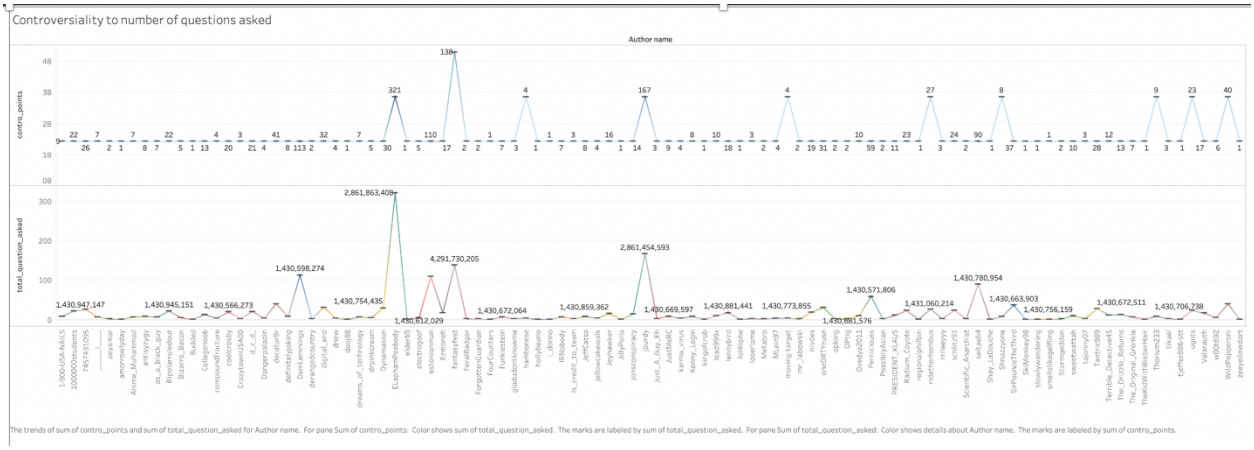
Subreddit and sum of count. Color shows sum of count. Size shows Running Sum of count. The marks are labeled by subreddit and sum of count.

There are so many different channels but out of those amny channels these are top 10 channels.

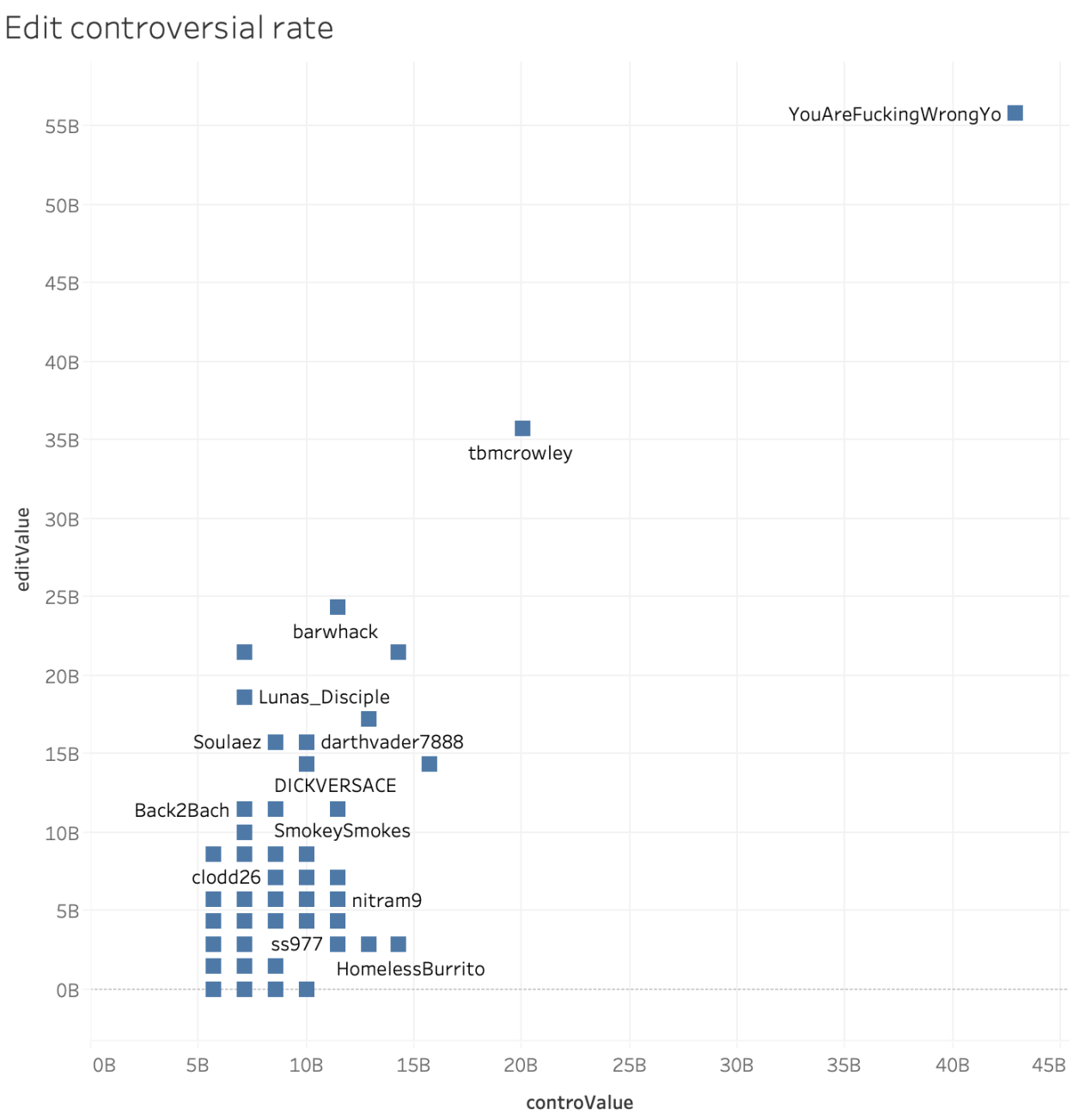


Author_name and sum of count. Color shows sum of count. Size shows sum of count. The marks are labeled by author_name and sum of count.

One of the important channel is NFL(National Football League) and these are top 10 authors who asked most of the questions. Data is labelled with names and how many times each person asked question.



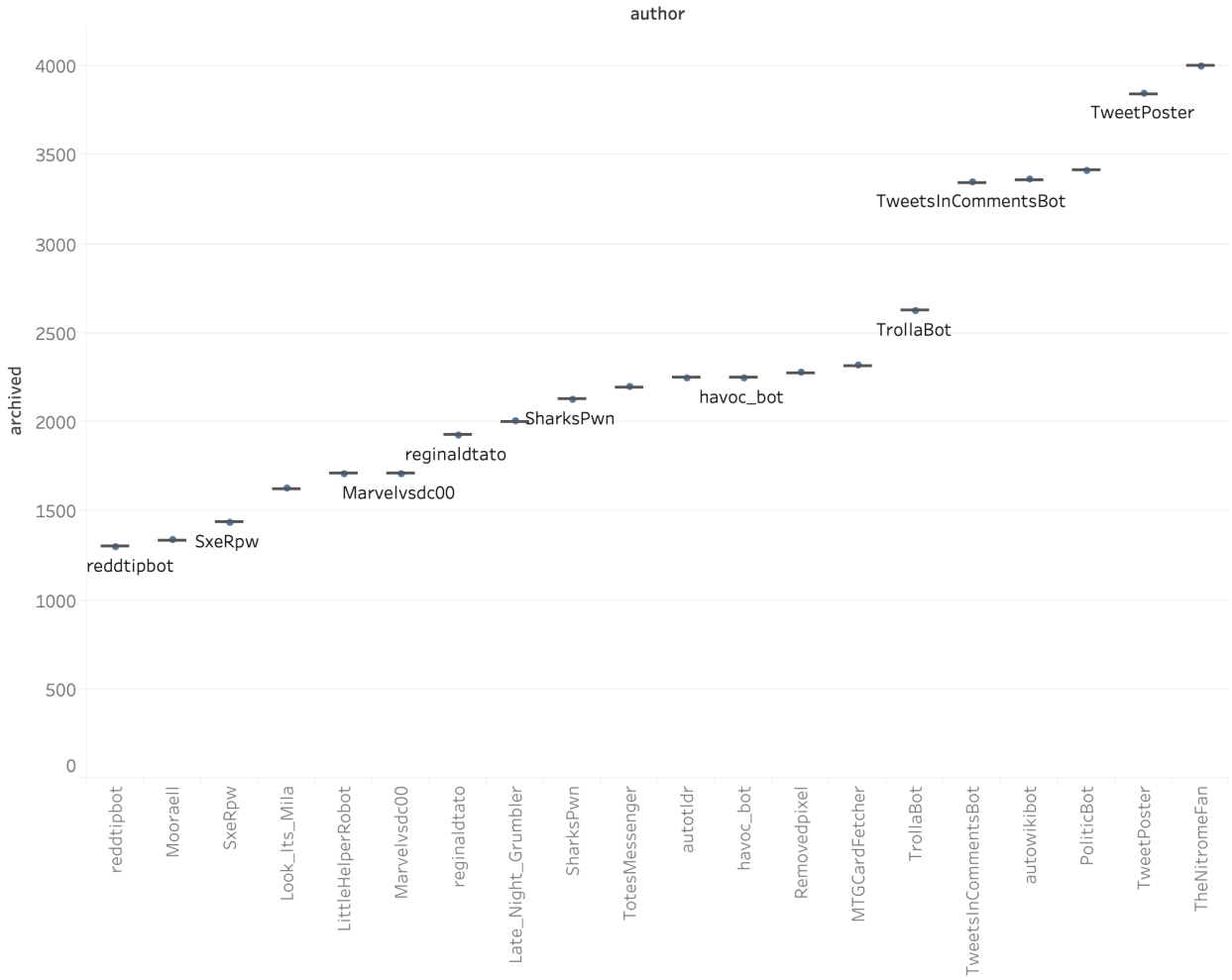
As we can see above that number of questions and controversial points has no relation in general, so by keeping that in reference wee can say that some questions are so damaging and controversial that even few questions can storm whole social network.



Sum of controValue vs. sum of editValue. The marks are labeled by Author. Details are shown for Author. The view is filtered on sum of editValue, which keeps non-Null values only.

As we can see above that as the edit rate increases the controversial that comments gets more and more. Because there is policy from reedit that if you can't improve your comments and don't make it less controversial than they will block it. So, naturally edit rate is higher for such commentns.

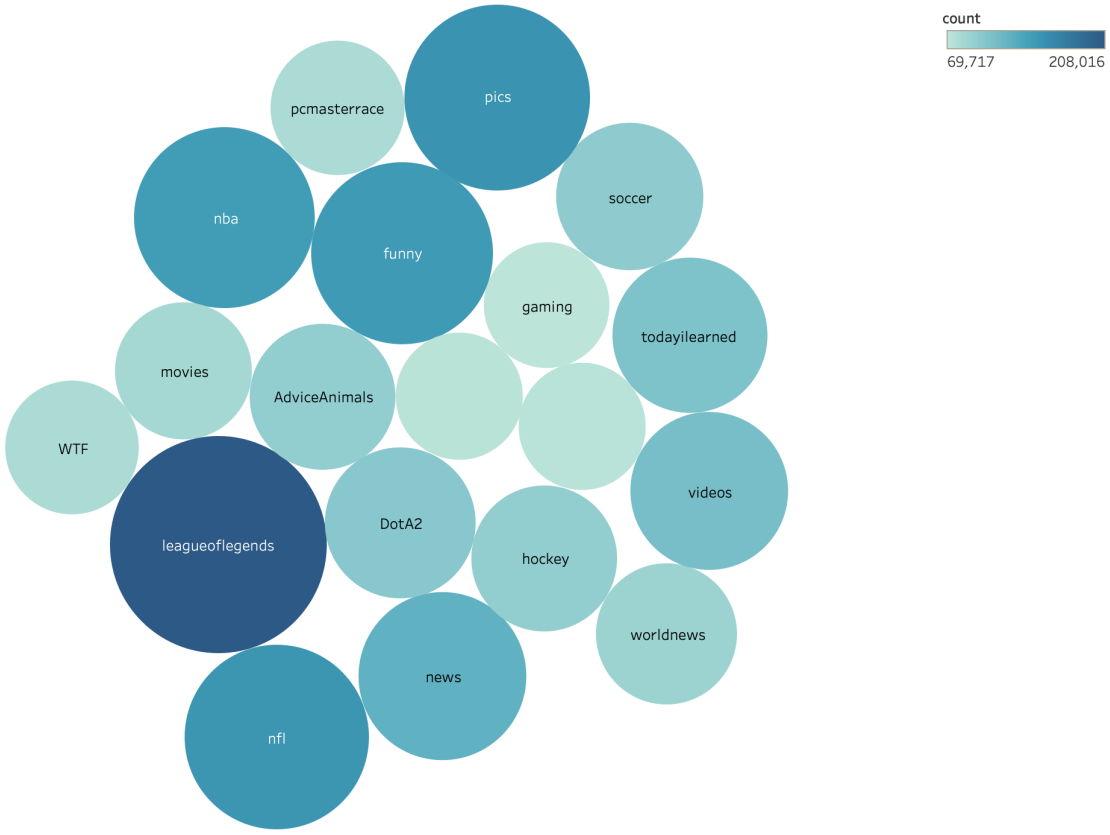
Top archived authors



Sum of archived for each author. The marks are labeled by author. Details are shown for author.

The above figure is for top 20 archived authors. Archived means when commenter gave the final fact based answer and there is nothing more can be done in that case. So, it's better idea to see which author's question is satisfactory answered.

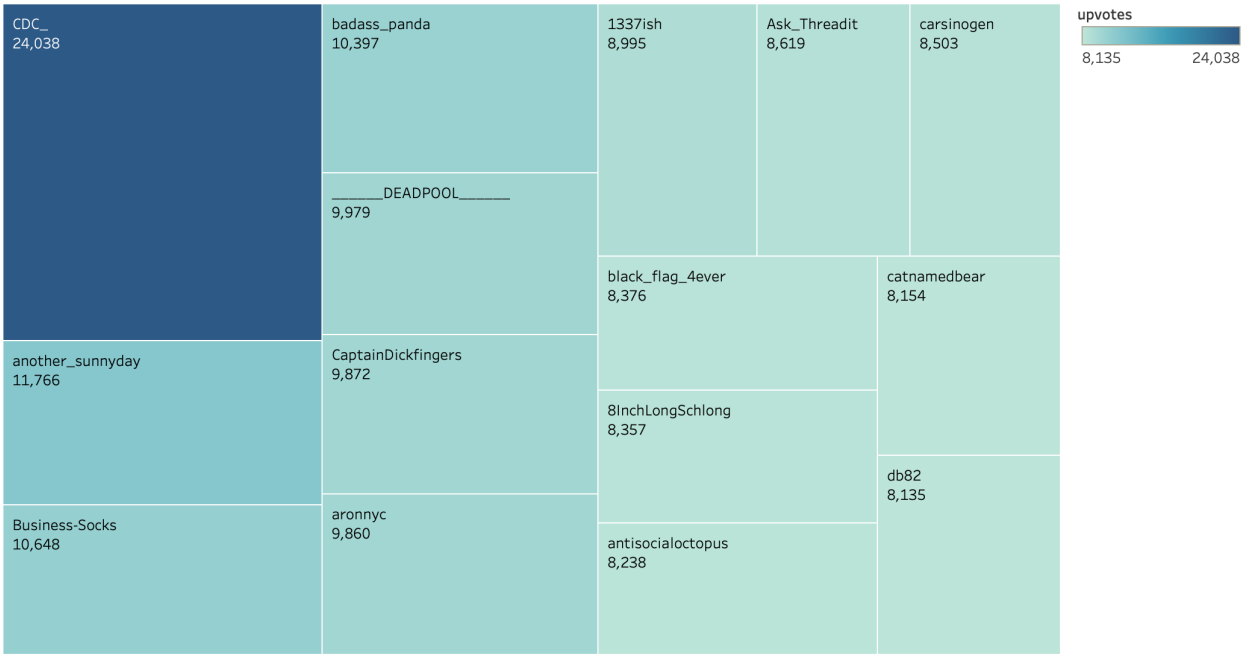
Subreddit archived



Subreddit. Color shows sum of count. Size shows sum of count. The marks are labeled by subreddit. The view is filtered on subreddit, which excludes AskReddit.

As the authors we have to see for sub redditors who have most archive rate this is very important because we have to see which redditors are answered properly.

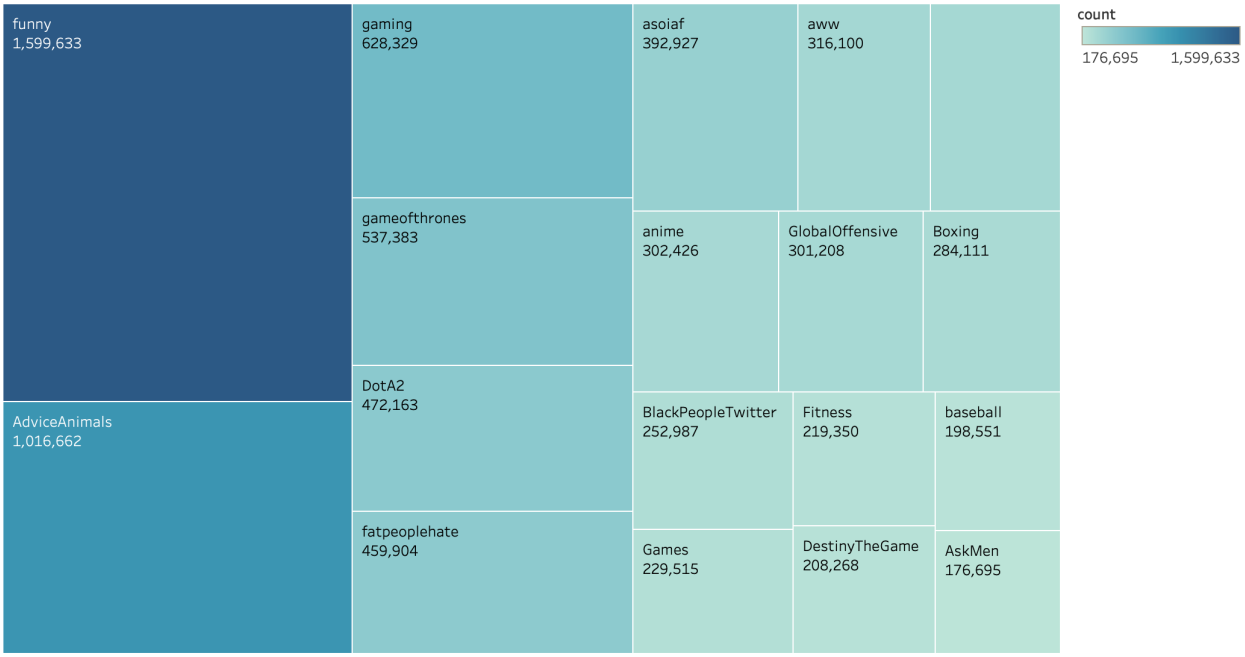
Top authors upvoted questions



Author and sum of upvotes. Color shows sum of upvotes. Size shows sum of upvotes. The marks are labeled by author and sum of upvotes. The data is filtered on sum of controversiality, which keeps non-Null values only. The view is filtered on author, which keeps 16 of 99 members.

This chart is for top authors who are upvoted all the time high. Each author above here is counted on the basiscs of total number of upvote they received over the time.

Top upvoted subreddits



Subreeddit and sum of count. Color shows sum of count. Size shows sum of count. The marks are labeled by subreeddit and sum of count. The view is filtered on subreeddit, which keeps 18 members.

As the authors received upvotes each sub redditors also received upvotes and here in this way funny is always higher than any other thing.

This is end of this tutorial.