Data Collection, Storage, and Analysis on NYU HPC

Leon Yin ~ Data Scientist | Research Engineer
at SMaPP and CDS
HPC Research Day 2017-11-08

Today I’ll talk about

1. Where is the data?
2. How does it get there?
3. How do we use it?

Where does Data live?

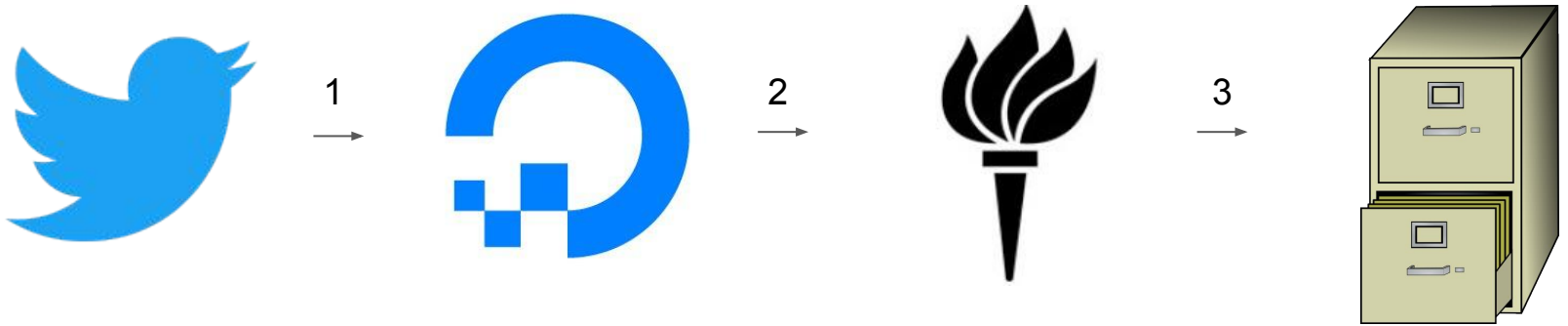
where?	What?	How Many?	How Much?	Access?
/scratch	Bzipped json files containing metadata	1.7M	8.2T	lab
/beegfs	Social Media Images and CSV metadata	1.7M	816 GB	lab
/archive	Backup scratch every week			me
Google Drive	Backup scratch and beegfs every day			me
hdfs	Chopped up metadata for queries			lab

How does data get there?

1. Twitter API hit on Digital Ocean instances 24/7.

2. Store daily files on Digital Ocean, RClone to HPC, and Bzip2.

3. Parse fields from metadata to create datasets for Images and Links.



Lessons from a Data Engineer

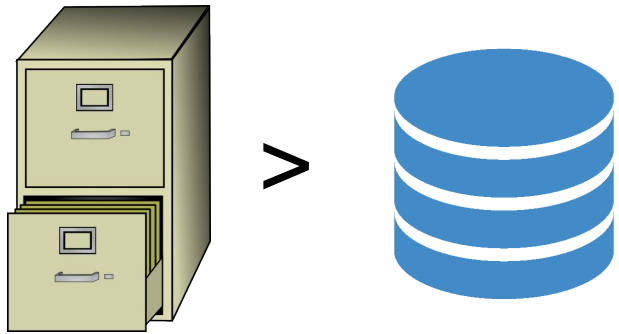
Store all data as compressed raw files.

→ Needs change, your data shouldn’t.

→ Databases are a medium, not a destination.

Crontab is unreliable for daily data pipelines.

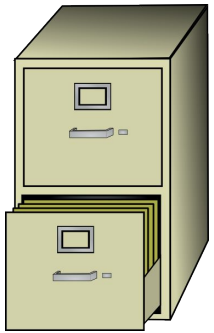
Cloud computing saves time and can be affordable!



What do we do with the data?

Run Jupyter Notebooks on Prince.
Usually w/ CPUs. Sometimes w/ GPUs. Sometimes on Dumbo.

Well-documented steps to get started on HPC in internal gDrive!



Links

Links scraped from Tweets found in `YOUGOV.TWITTER_RAW` using this scraper I made.

We join the walk `YOUGOV.USER_FRIEND_MAPPING_VIEW`, which adds the `SURVEY_USER_ID`.

In [65]:
```
%%sh
hdfs dfs -ls [redacted] | head -2
```

```
Found 9949 items
-rwxr-x---   3 [redacted]  1871984 2017-06-24 00:58 [redacted]
[redacted]csv.bz2
```

Used to generate this SQL table:
-> YOUGOV.LINKS

Cast columns from string to int, and join `survey_user_id`'s in this SQL view:
-> YOUGOV.LINKS_VIEW

In [54]:
```
overview('YOUGOV.LINKS_VIEW')
```

YOUGOV.LINKS_VIEW (N=[redacted])

Out[54]:

	col_name	data_type	comment
0	link_domain	string	
1	link_url_long	string	
2	link_url_short	string	
3	created_at	string	
4	tweet_id	bigint	
5	user_id	bigint	
6	process_date	string	
7	tweet_collection	string	
8	tweet_file	string	
9	survey_user_id	bigint	

Query 4

Read from here to a string. Search terms included through string formatting.

In [43]:
```
fake_news = '''usanewsflash.com
abcnews.com.co
denverguardian.com
rickwells.us
truepundit.com
redstatewatcher.com
worldpoliticus.com
subjectpolitics.com
conservativestate.com
conservativedailypost.com
libertywritersnews.com
worldnewsdailyreport.com
endingthefed.com
donaldtrumpnews.co
yesimright.com
burrardstreetjournal.com
bizstandardnews.com
everynewshere.com
departed.co
americanmilitarynews.com
tmzhiphop.com
winningdemocrats.com'''.replace('\n','|')
```

In [44]:
```
with open('sql/queries/q4.sql') as q:
    q4 = q.read().format(fake_news)

print(q4)
```

```
-----------------------------------------------
-- This is a query counts tweets containing
-- links to fake news from survery users'
-- friend networks
--
-- The subquery aliased as "A", contains all
-- the unique survey user ids.
-----------------------------------------------
SELECT
    A.SURVEY_USER_ID AS SURVEY_USER_ID,
    F.FAKE_NEWS AS FAKE_NEWS,
    Z.ALL_TWEETS AS ALL_TWEETS
FROM (
    SELECT DISTINCT SURVEY_USER_ID
    FROM YOUGOV.USER_FRIEND_MAPPING_VIEW
) A

-----------------------------------------------
-- TO THE LIST OF SURVEY USER IDS,
-- WE JOIN COUNTS OF COMMENTS
-- CONTAINING FAKE NEWS LINKS.
-----------------------------------------------
LEFT JOIN (
    SELECT
        SURVEY_USER_ID,
        COUNT(*) AS FAKE_NEWS
    FROM YOUGOV.LINKS_VIEW
    WHERE LOWER(LINK_DOMAIN) RLIKE 'usanewsflash.com|abcnews.com.co|denverguardian.com|rickwells.us|truepundit.com|redstatewatcher.com|worldpoliticus.com|subjectpolitics.com|conservativestate.com|conservativedailypost.com|libertywritersnews.com|worldnewsdailyreport.com|endingthefed.com|donaldtrumpnews.co|yesimright.com|burrardstreetjournal.com|bizstandardnews.com|everynewshere.com|departed.co|americanmilitarynews.com|tmzhiphop.com|winningdemocrats.com'
    GROUP BY SURVEY_USER_ID
) F
ON F.SURVEY_USER_ID = A.SURVEY_USER_ID
```

In [45]:
```
df4 = pd.read_sql(q4, conn)
df4.head()
```

Out[45]:

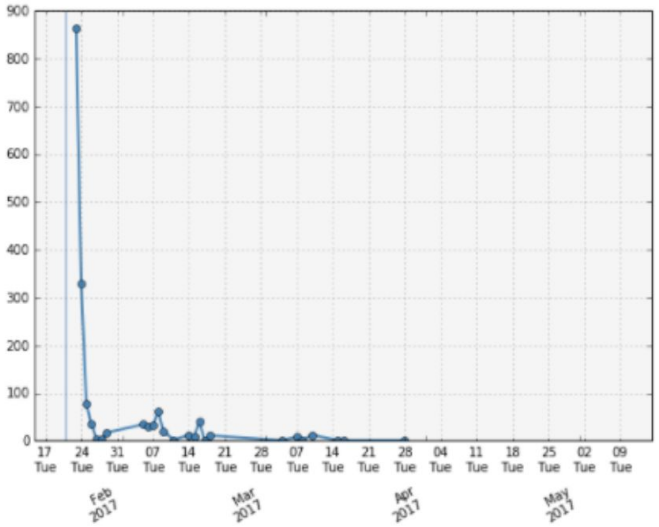
	survey_user_id	fake_news	all_tweets
0	[redacted]	2	1231315
1	[redacted]	NaN	102573
2	[redacted]	NaN	6828
3	[redacted]	1	124134
4	[redacted]	1	7351

In [46]:
```
df4.to_csv(query_csv.format(4), index=False)
```

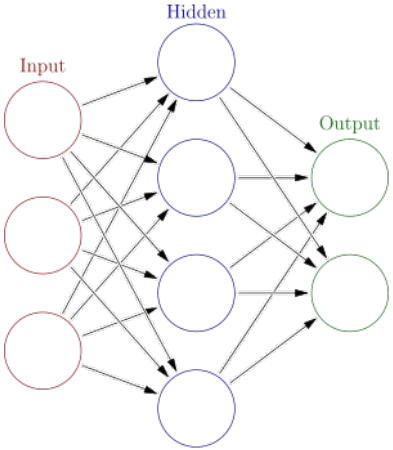
Islamophobic Propaganda during Women's March

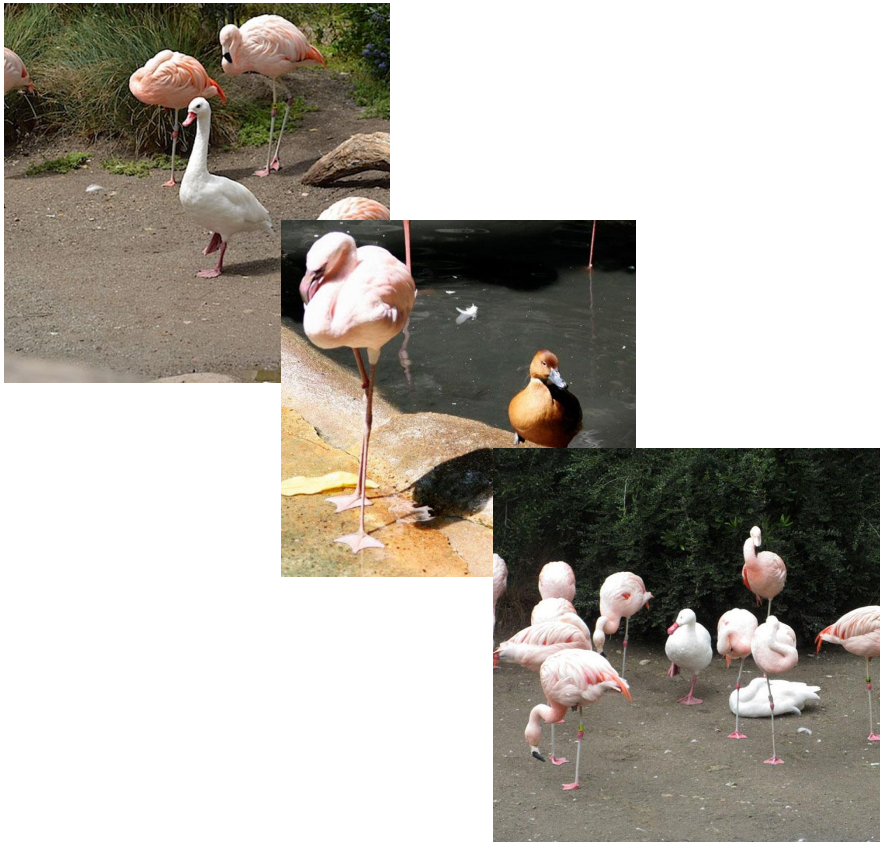
```
In [167]: get_meta(82[redacted])
```

```
82[redacted]
First retweet: 2017-01-23 00:00:00
Retweets: 1607
Link to tweet: https://t.co/[redacted]
L. Sarsour who thinks Sharia is "fair" and wants it 4 America, head of #WomensMarch -  giving
 us the (#ISIS) finger… https://t.co/[redacted]
```



Reverse Image Search using Keras and Sklearn





Questions:

How to make data available to external collaborators or researchers who leave?

How to accommodate time-sensitive data pipelines?

Thanks

@leonyin