DLRL CLUSTER

CS4624 Spring 2014
Virginia Tech
Blacksburg, VA
Client: Sunshin Lee

Adam Lech
Joseph Pontani
Matthew Bollinger

Outline

- Deliverables
- Data Preprocessing
- Hive
- HBase
- Impala
- Mahout
- Future Work

Deliverables

- Tutorials
- Video demos
- Report generation
  - HBase
  - Hive
  - Impala
  - Mahout

Data preprocessing

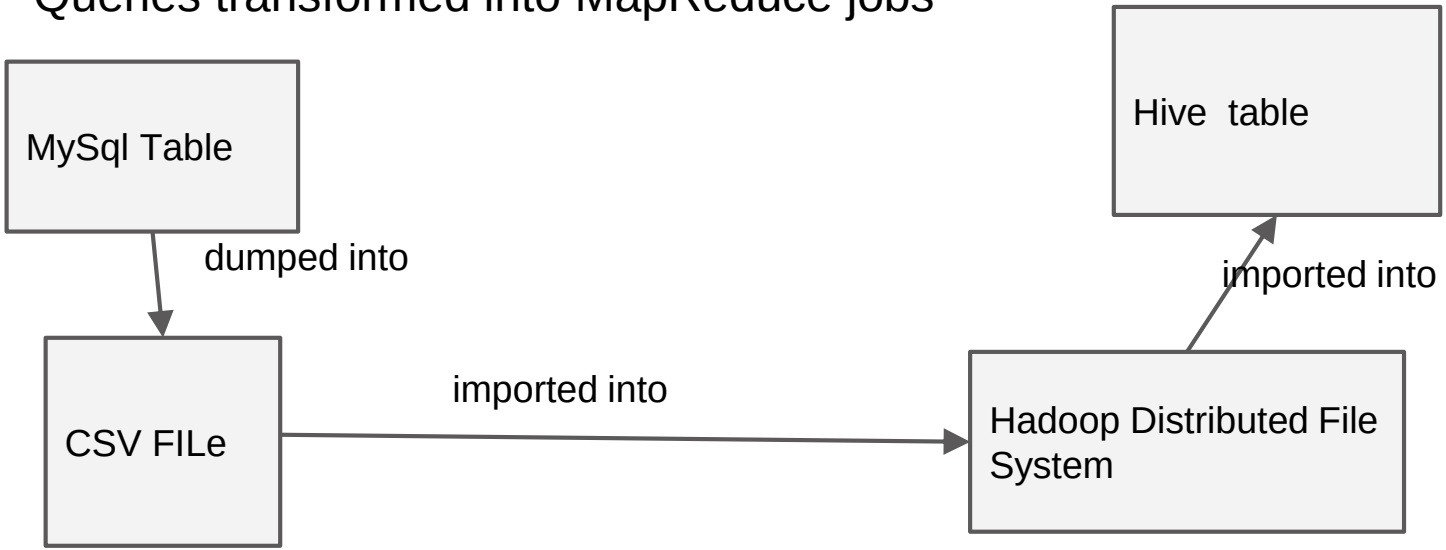
- Needed to preprocess data for meaningful analysis
  - source
    - strip excess URL information
    - Tweetbot for iOS
    - tapbots.com
  - tweet date
    - separate into fields for year, month, and date
  - tweet text
    - remove stop words (‘the’, ‘and’, etc.)
- Input CSV into Python script
- Dumped out to CSV file for use

Preprocessing Challenges

- Tweets are from two different sources
  - twitter-stream
  - twitter-search
- Different formats
  - Tweetbot for iOS
  - Twitter for Android
- Full of weird characters that threw script off
- Large datasets take FOREVER to process

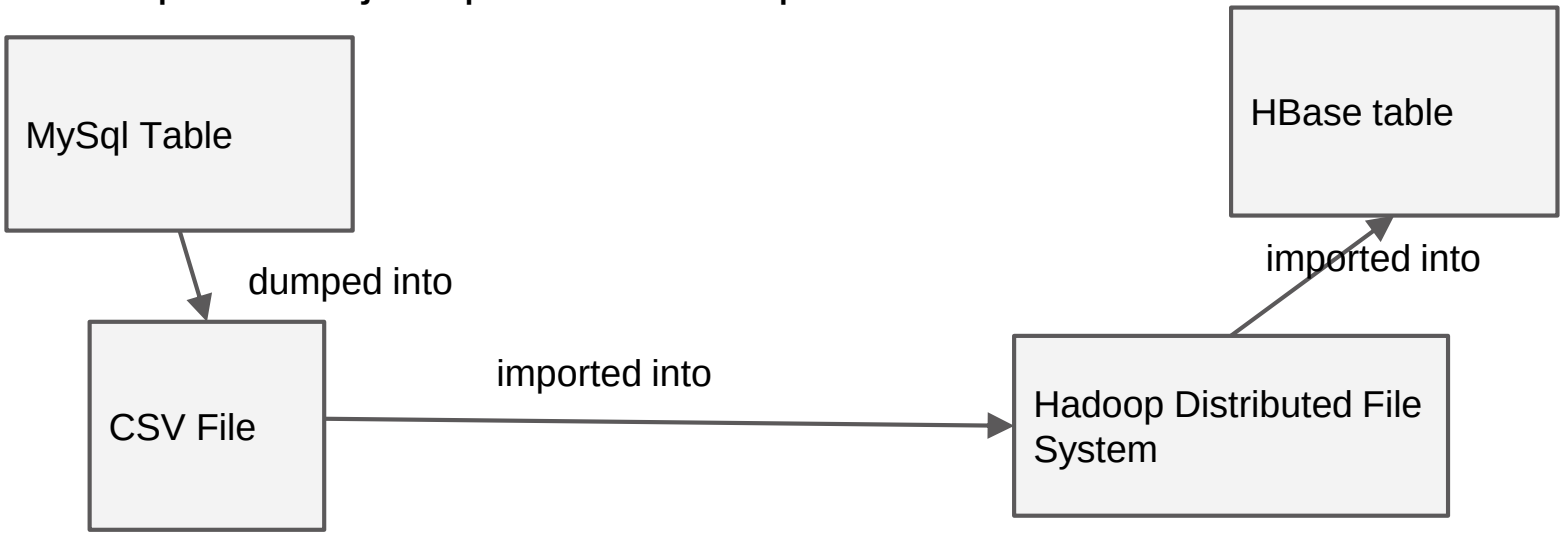
Hive

- Importing pothole dataset to Hive
- Statements similar to loading data in MySql
- Data still stored in file
- Queries transformed into MapReduce jobs



HBase

- Importing pothole dataset to HBASE
- HBase requires KEY_ROW for exactly one column
- HBase organizes fields into column family
- MapReduce jobs performed - ImportTsv tool used



Impala

- Setup tables in Impala with new datasets
- Create benchmark queries to test Impala vs Hive

```
	select count(*) as num_tweets, from_user from twitter group by from_user order by num_tweets desc limit
15;
	select count(*) as num_tweets, source from twitter group by source order by num_tweets desc limit 15;
	select count(*) as num_tweets, created_at from twitter group by created_at order by num_tweets desc limit
15;
select count(*) as num_tweets, month from twitter group by month order by num_tweets desc limit 15;
select count(*) as num_tweets, day from twitter group by day order by num_tweets desc limit 15;
select count(*) as num_tweets, year from twitter group by year order by num_tweets desc limit 15;
```

Impala - Example Report

```
select count(*) as num_tweets, from_user from twitter group by from_user order by num_tweets desc limit 15;
select count(*) as num_tweets, month from twitter group by month order by num_tweets desc limit 12;
select count(*) as num_tweets, month from twitter group by month order by month desc limit 12;
```

num_tweets	from_user
2912	GrandRapids311
2714	mrpotholeuk
1720	citizensconnect
1435	NJI95thm
1202	baltimore311
1189	NYI95thm
1135	NYI78thm
843	NJI78thm
656	MarquelatTPV
576	FixedInDC
498	NYI87thm
497	csreports
374	BridgeviewDemo
355	MPLS311
340	edm_pothole

num_tweets	month
61243	2
60555	3
25212	1
23009	4
12706	5
11897	12
10947	8
9906	10
9779	9
9538	6
8602	11
7809	7

num_tweets	month
11897	12
8602	11
9906	10
9779	9
10947	8
7809	7
9538	6
12706	5
23009	4
60555	3
61243	2
25212	1

Mahout

- How to use Mahout
  - preprocess dataset
    - remove ‘stop words’ and other unnecessary text
  - import dataset to HDFS
    - pothole and shooting dataset are 1 tweet per line
  - datamine using FPGrowth algorithm to get frequent patterns
    - specify word separator, in this case a space
  - view/dump results
- Deliverable: Tutorial (PDF) and Demo video (Youtube)
  - tweets about potholes, 20MB CSV file
  - how to run Mahout with FPGrowth on a dataset
- Finally, run FPG on actual cluster with a much larger dataset
  - tweets about shootings, 8GB CSV file

Mahout

- Issues with ‘Shooting’ dataset
  - FPG only needs the tweet text, how to preprocess dataset to remove all other columns, 8GB CSV file took forever to preprocess via Python script
    - solution was to just export only the tweet from MySQL
  - Java heap space is exhausted when running Mahout using mapreduce on a large dataset
    - lower the requested heap size (top k values are kept) when running FPG via the k switch (from -k 50 to -k 10) and increase minimum groupings via s switch (from -s 3 to -s 10)

Future Work

- Preprocess tweets on their way in, not after the fact
- Leverage different technologies for specific tasks in DLRL Cluster