Big-Data Systems and Intelligent Analytics

Twitter based Movie Ratings

Team 4:
Samir Sharan
Harshit Shah
Jeevan Reddy

Goal

The goal of this project is to provide a user-sourced and always up-to-date movie ratings. The ratings will be updated as much as possible by incorporating data from the newest tweets available.

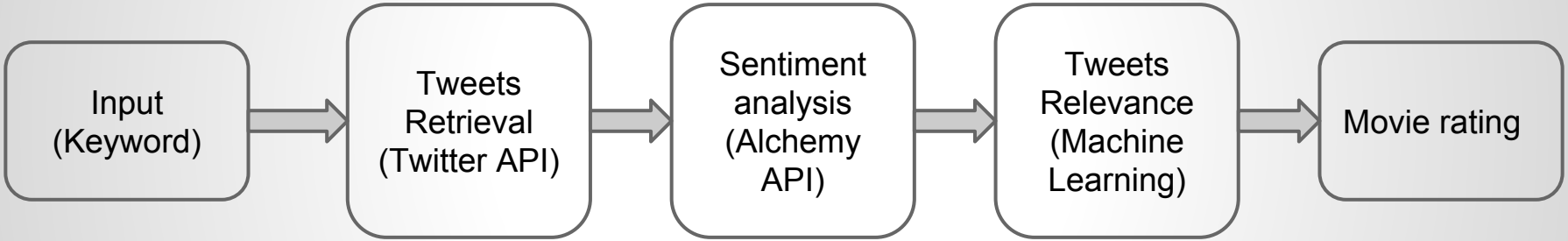
Why Twitter?

- Twitter has 302 million active users
- 500 million tweets per day
- Tweets reflect user’s real thoughts and opinions
- Real time information
- Can capture general audiences opinion

Methods:

- Alchemy API
- AWS Sentiment Analysis

Flow


Input
(Keyword)
Tweets
Retrieval
(Twitter API)
Sentiment
analysis
(Alchemy
API)
Tweets
Relevance
(Machine
Learning)
Movie rating

Tweet Sentiment

{"verified": false,
"screen_name": "hiyaimdemi",
"RT_count": 0,
"text": "southpaw is such a good film\ud83d\ude33",
"coordinates": null,
"**sentiment": "positive**",
"language": "en",
"**score": 0.629288**,
"location": null,
"time": "Wed Aug 19 21:30:24 +0000 2015",
"movie": "Southpaw",
"user_followers_cnt": 1078,
"id": 634115221662134272,
"fav_count": 1}

- **Score**: Degree of sentiment

Tweet Relevance

{**"verified": false**,
"screen_name": "hiyaimdemi",
"RT_count": 0,
"text": "southpaw is such a good film\ud83d\ude33",
"coordinates": null,
"sentiment": "positive",
"language": "en",
"score": 0.629288,
"location": null,
"time": "Wed Aug 19 21:30:24 +0000 2015",
"movie": "Southpaw",
"user_followers_cnt": 1078,
"id": 634115221662134272,
"fav_count": 1}

Relevance Score: W1*RT_count + W2*fav_count + W3*user_followers_count + W4*verified

Machine Learning

- Train on a set of 21,000 tweets
  - features - RT_count, fav_count, user_follower_count, verified
  - label - Relevance score
- Test on 14,000 tweets
  - Predict Relevance Score
  - Calculate Mean Squared Error

Rating

- Depends on Sentiment Score (SS) and Relevance Score (RS)

- R = (SS * RS) -> scale on 5 -> add to 2.5
- R = [(0.1632) * 5] + 2.5
- R = 3.316

Rating

Day Wise Rating

Date

Movie	19	20	21
FantasticFour	3.018	3.105	3.242
InsideOut	2.966	2.908	2.858
Minions	2.721	2.518	2.702
RogueNation	2.308	2.193	2.352
Southpaw	2.582	2.585	2.574

Avg. Rating

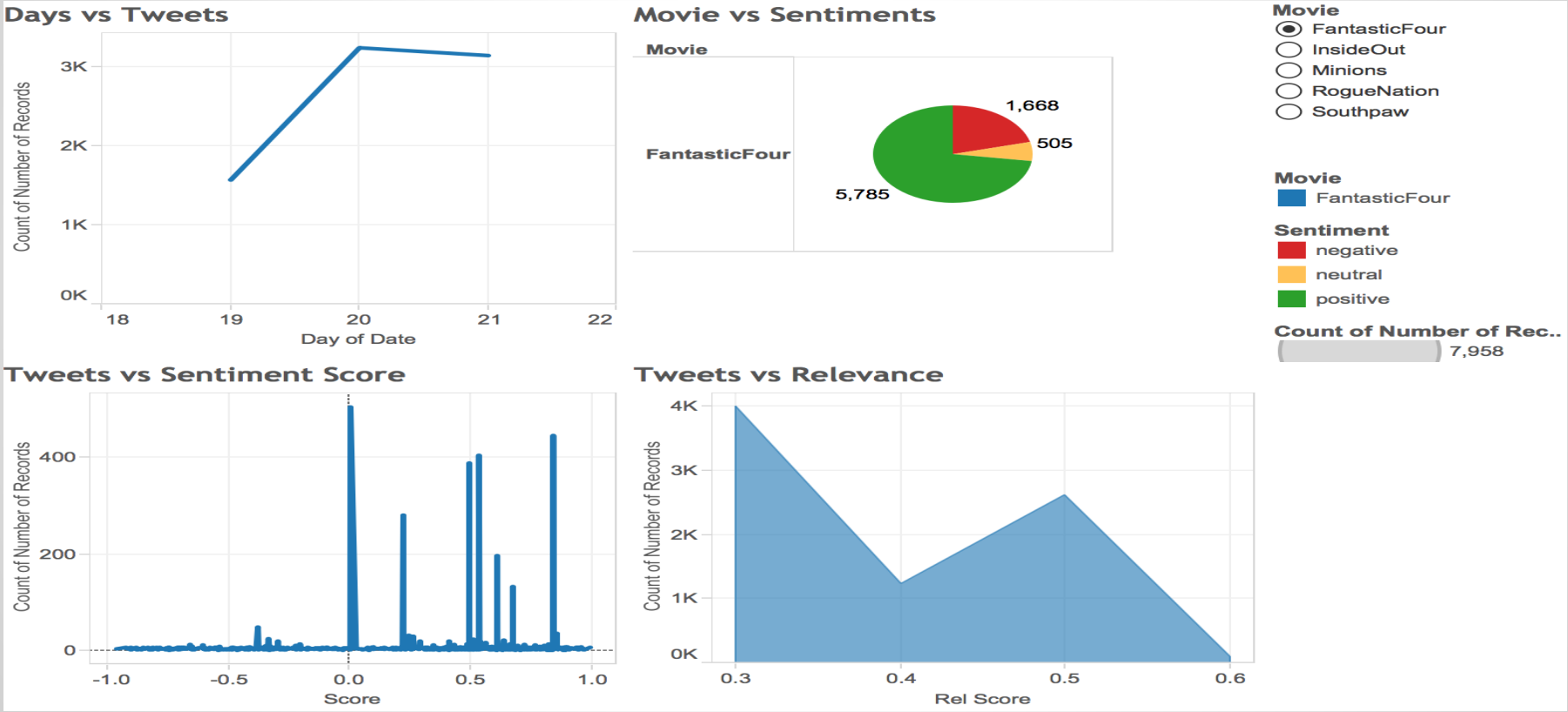
2.193 — 3.242

Average of Rating broken down by Date Day vs. Movie. Color shows average of Rating. The marks are labeled by average of Rating. The view is filtered on Movie, which keeps FantasticFour, InsideOut, Minions, RogueNation and Southpaw.

Rating


Average Rating Overall
Movie
FantasticFour
InsideOut
Minions
RogueNation
Southpaw
3.1420
2.8904
2.6059
2.2799
2.5796
Avg. Rating
2.2799
3.1420

Dashboard


Days vs Tweets
Count of Number of Records
3K
2K
1K
0K
18
19
20
21
22
Day of Date
Movie vs Sentiments
Movie
FantasticFour
1,668
505
5,785
Movie
FantasticFour
InsideOut
Minions
RogueNation
Southpaw
Movie
FantasticFour
Sentiment
negative
neutral
positive
Count of Number of Rec..
7,958
Tweets vs Sentiment Score
Count of Number of Records
400
200
0
-1.0
-0.5
0.0
0.5
1.0
Score
Tweets vs Relevance
Count of Number of Records
4K
3K
2K
1K
0K
0.3
0.4
0.5
0.6
Rel Score

AWS Sentiment Analysis

- Create an Amazon S3 Bucket
- Collect and Store the data
- Build the mapper
- Create an Amazon EMR cluster
- Examine the output

Steps in EMR:

Add Step

Step type Streaming program

Name* Streaming program

Mapper* sentiment.py — S3 location of the map function or the name of the Hadoop streaming command to run.

Reducer* aggregate — S3 location of the reduce function or the name of the Hadoop streaming command to run.

Input S3 location* s3://aws-logs-202509000957-us-east-1/input/

s3://<bucket-name>/<folder>/

Output S3 location* s3://aws-logs-202509000957-us-east-1/final_output/

s3://<bucket-name>/<folder>/

Arguments

```
-files
s3://awsdocs/gettingstarted/latest/sentiment/classifier.p#classifier.p,s3://aws-logs-202509000957-us-east-1/mapper/sentiment.py
```

Action on failure Continue — What to do if the step fails.

Cancel | Add

S3 Bucket:

Upload | Create Folder | Actions

All Buckets / aws-logs-202509000957-us-east-1

Name
elasticmapreduce
input
mapper

Output:

Name	Storage Class	Size	Last Modified
_SUCCESS	Standard	0 bytes	Mon Aug 17 19:53:03 GMT-400 2015
part-00000	Standard	0 bytes	Mon Aug 17 19:52:53 GMT-400 2015
part-00001	Standard	0 bytes	Mon Aug 17 19:52:54 GMT-400 2015
part-00002	Standard	23 bytes	Mon Aug 17 19:52:55 GMT-400 2015
part-00003	Standard	0 bytes	Mon Aug 17 19:52:56 GMT-400 2015
part-00004	Standard	15 bytes	Mon Aug 17 19:52:58 GMT-400 2015
part-00005	Standard	0 bytes	Mon Aug 17 19:53:01 GMT-400 2015
part-00006	Standard	0 bytes	Mon Aug 17 19:53:02 GMT-400 2015

```
No match:        12
```

```
minions: positive        364
```

References

- Alchemy API: http://www.alchemyapi.com/products/alchemylanguage/sentiment-analysis
- AWS Sentiment Analysis: http://docs.aws.amazon.com/gettingstarted/latest/emr/getting-started-emr-sentiment-tutorial.html

Thank You!