**README**

**Files used:**

game\_inventory.csv, developer.csv, genre.csv, platform.csv, game\_inventory\_mongo.csv

These are the CSV files that were pulled in from the MySQL database that holds the data to be loaded into the MongoDB.

**Test Cases**:

1. **What are tags are associated with a person, place or thing?**

Here, we used 'Final Fantasy VII', 'final fantasy 7','#ff7r','#FF7R','finalfantasy7' as tags to scrap the tweets for the game 'Final Fantasy VII'. Tags associated with a game can be related to name, genre and platform too. Tags used: 'Final Fantasy VII', 'final fantasy 7','#ff7r','#FF7R','finalfantasy7'.

tags = pd.DataFrame(columns=['Name','Genre','Platform'])

i=0

for doc in db.Game\_Inventory.find():

tags.loc[i] = [doc.get('G\_NAME'),doc.get('GENRE'), doc.get('PLATFORM')]

i += 1

| tags **Name** | **Genre** | **Platform** |
| --- | --- | --- |
| 0 | Mario Kart Wii | Racing | Wii |
| 1 | Wii Sports Resort | Sports | Wii |
| 2 | New Super Mario Bros. | Platform | DS |
| 3 | New Super Mario Bros. Wii | Platform | Wii |
| 4 | Wii Play | Misc | Wii |
| ... | ... | ... | ... |
| 59 | Red Steel | Shooter | Wii |
| 60 | Phantasy Star Portable | Role-Playing | PSP |
| 61 | Bionic Commando | Adventure | X360 |
| 62 | The Conduit | Shooter | Wii |
| 63 | Submarine Titans | Strategy | PC |

here we get 64 rows × 3 columns as a result for the tags associated with the 'Final Fantasy VII'

1. **What social media users are like other social media users in your domain?**

Here in our scope we used the Twitter data to gather the social media users

col = ['user\_id','user\_name','user\_description','location','followers\_count','verified']

location = pd.DataFrame(columns = col)

i=0

for tweet in db.twitter.find():

location.loc[i] = [tweet.get('\_id'),tweet.get('user\_name'),tweet.get('user\_description'),tweet.get('tweet\_loc'),tweet.get('followers\_count'),tweet.get('verified')]

i += 1

loc\_final = location.groupby('location')['location'].agg('count').sort\_values(ascending = False)

loc\_df = pd.DataFrame({'location':loc\_final.index,'count':loc\_final.values})

loc\_df.head(10)

| **location** | **count** |
| --- | --- |
| 0 | Nibelheim | 17 |
| 1 | ツイートをプッシュ通知で受け取るのがオススメです！ | 5 |
| 2 | Los Angeles, CA | 5 |
| 3 | England, United Kingdom | 5 |
| 4 | In your heart | 3 |
| 5 | 神の祠 | 3 |
| 6 | Everywhere and always. | 3 |
| 7 | New York, USA | 3 |
| 8 | Chez Moi | 2 |
| 9 | United States | 2 |

From the above table we can see that almost 17 of the users have put Nibelheim as their tweeting location. It's pretty interesting since Nibelheim is a fictional city in Final Fantasy VII. This can show that most of the users who have tweeted about Final Fantasy VII are fans of the game.

1. **What people, places or things are popular in your domain?**

We can check the no. of followers count of each users to determine the popular people in our domain. i.e. in Twitter.

import pprint

flag = 0

col = ['user\_name','followers\_count']

foll = pd.DataFrame(columns = col)

i=0

for tweet in db.twitter.find():

foll.loc[i] = [tweet.get('user\_name'),tweet.get('followers\_count')]

i=i+1

foll = foll.sort\_values(by ='followers\_count',ascending = False)

foll.head(5)

|  | **user\_name** | **followers\_count** |
| --- | --- | --- |
| 285 | BRKsEDU | 1552487 |
| 187 | VG247 | 210100 |
| 396 | Pixelania | 89891 |
| 367 | flaviagasi | 43022 |
| 250 | Ligero1 | 40240 |

1. **What people, places or things are trending in your domain?**

We can check the trending tweets of twitter by calculating various tweets with an average of favourites count and retweets count.

trends **=** pd.DataFrame(columns**=**['user\_id','user\_name','favourites\_count','retweets\_count','followers\_count'])

**for** tweet **in** db.twitter.find():

trends.loc[i] **=** [tweet.get('\_id'),tweet.get('user\_name'),tweet.get('fav\_count'),tweet.get('retweets'),tweet.get('followers\_count')]

i **+=** 1

trnd\_fav **=** trends.groupby('favourites\_count')['favourites\_count'].agg('count').sort\_values(ascending **=** **False**)

trnd\_retweets **=** trends.groupby('retweets\_count')['retweets\_count'].agg('count').sort\_values(ascending **=** **False**)

trnd\_fav\_df **=** pd.DataFrame({"fav\_count":trnd\_fav.index, "count":trnd\_fav.values})

trnd\_retweets\_df **=** pd.DataFrame({'retweet\_count':trnd\_retweets.index,'count':trnd\_retweets.values})

trnd\_fav\_df

|  | **fav\_count** | **count** |
| --- | --- | --- |
| 0 | 0 | 411 |

trnd\_retweets\_df

|  | **retweet\_count** | **count** |
| --- | --- | --- |
| 0 | 0 | 411 |

From the above data, we can see that all the tweets that we have scrapped only have 0 favourite count and 0 retweet count. Since we scrapped tweets that were created recently, that might be one reason why the tweets have 0 favourite count and 0 retweet. Instead of using favourites count and retweet count, we can count the no. of followers a user has to show that since they have a large no. of followers their post will reach a lot of audience.

foll.head(5)

|  | **user\_name** | **followers\_count** |
| --- | --- | --- |
| 285 | BRKsEDU | 1552487 |
| 187 | VG247 | 210100 |
| 396 | Pixelania | 89891 |
| 367 | flaviagasi | 43022 |
| 250 | Ligero1 | 40240 |

loc\_df.head()

|  | **location** | **count** |
| --- | --- | --- |
| 0 | Nibelheim | 17 |
| 1 | ツイートをプッシュ通知で受け取るのがオススメです！ | 5 |
| 2 | Los Angeles, CA | 5 |
| 3 | England, United Kingdom | 5 |
| 4 | In your heart | 3 |