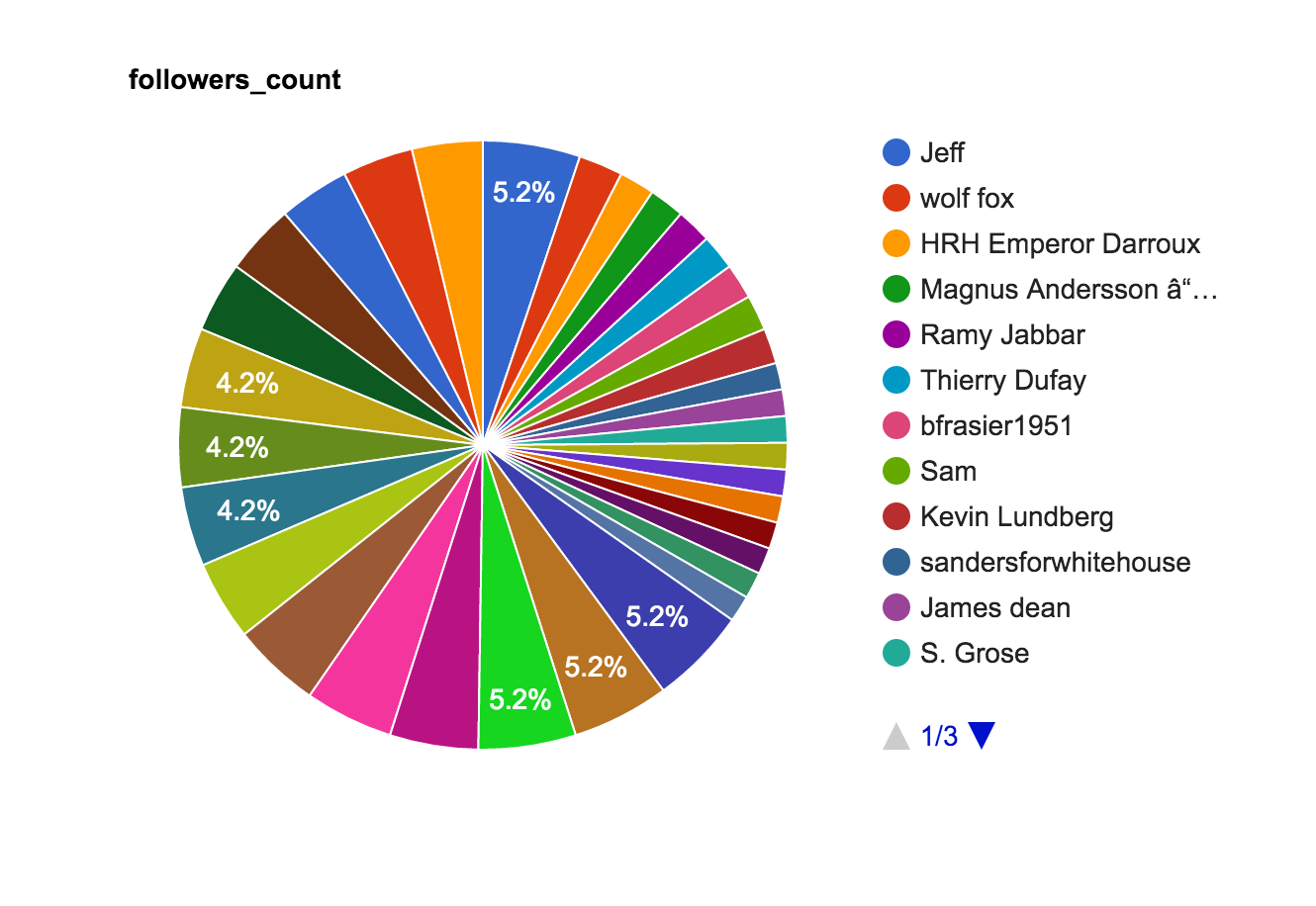
Query 1:

This query finds followers count of every politician and gives the output of the top people who have the highest followers count.

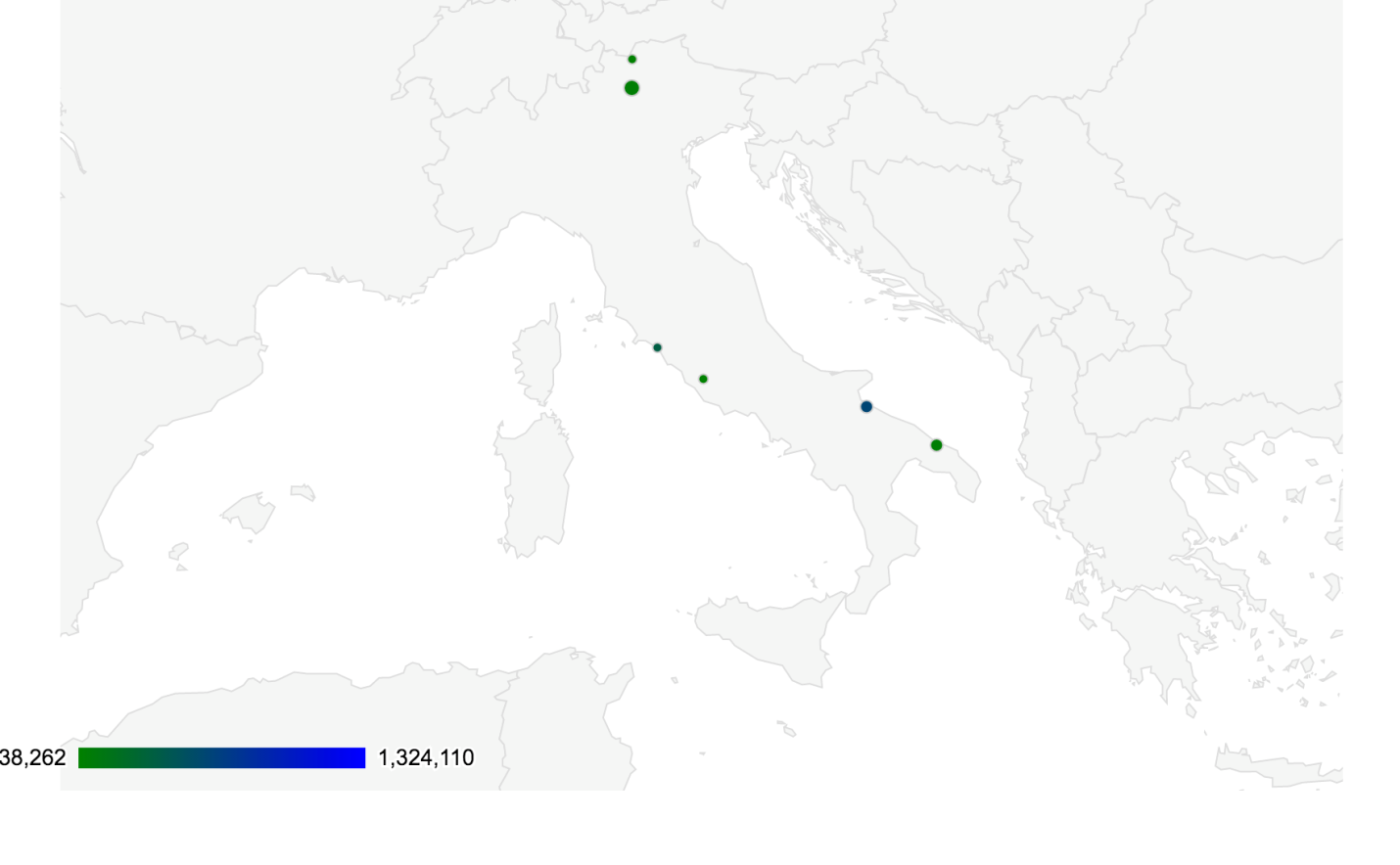
*val query1= sqlContext.sql("select user.name, count(user.followers\_count) as followersCount from tweettable group by user.name order by followersCount desc limit 20")*



Query 2:

This query gives the users location. This helps us know the location of the user and tells the different regions from where people tweet.

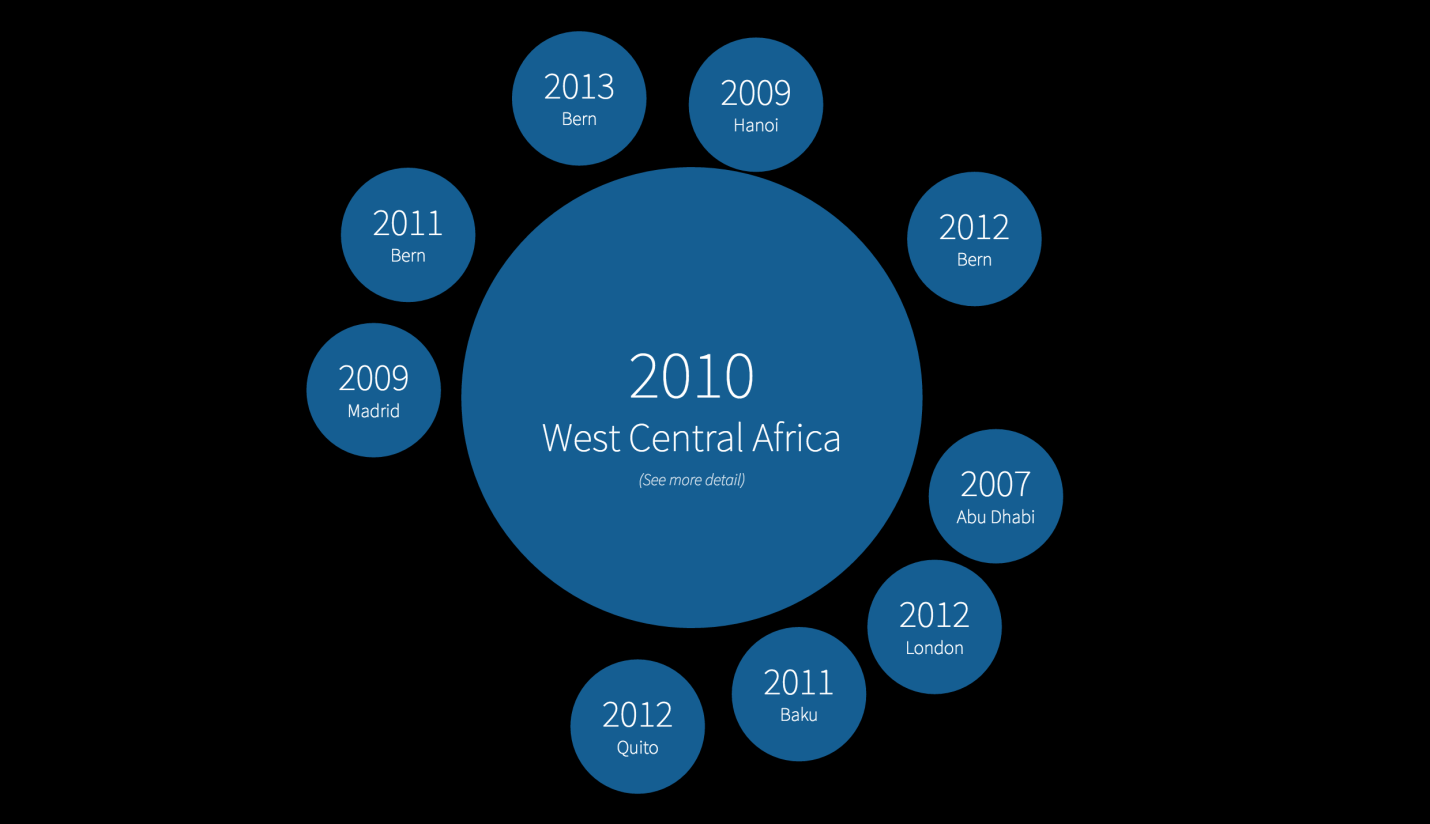
*Val query2=sqlContext.sql("select count(\*), user.location from tweettable where user.location is not null group by user.location limit 500")*

**

Query 3:

This query gives the top 10 time zones. This helps know the popular tweet created in different time zones.

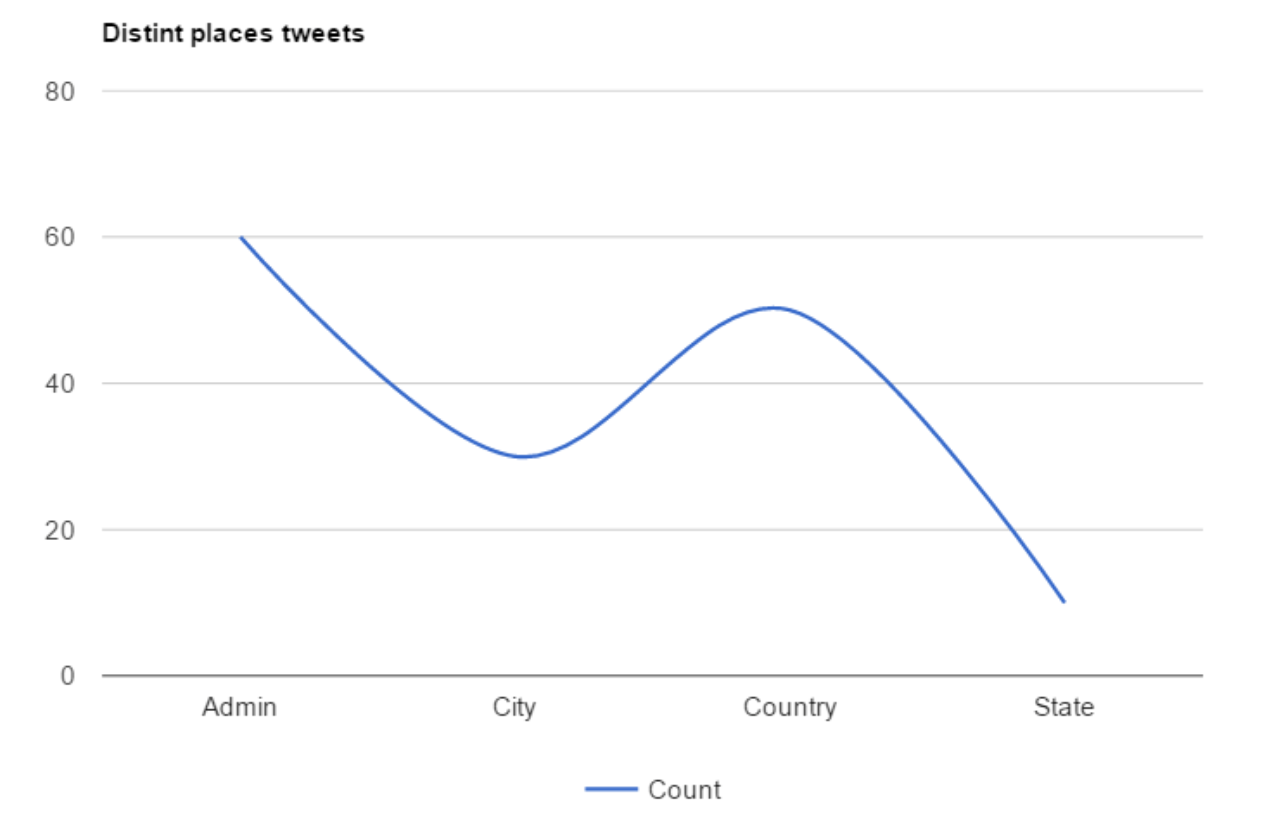
*val query3= sqlContext.sql("select user.time\_zone, SUBSTR(user.created\_at,27,31), count(\*) as total\_count from tweettable where user.time\_zone is not NULL group by user.time\_zone, SUBSTR(user.created\_at,27,31) limit 10")*



Query 4:

This tweet gives the number of cities , countries which are recorded in tweet data.

*val query4 = sqlContext.sql("SELECT DISTINCT place.place\_type, COUNT(place.place\_type) AS tweet\_count FROM tweettable GROUP BY place.place\_type limit 100")*

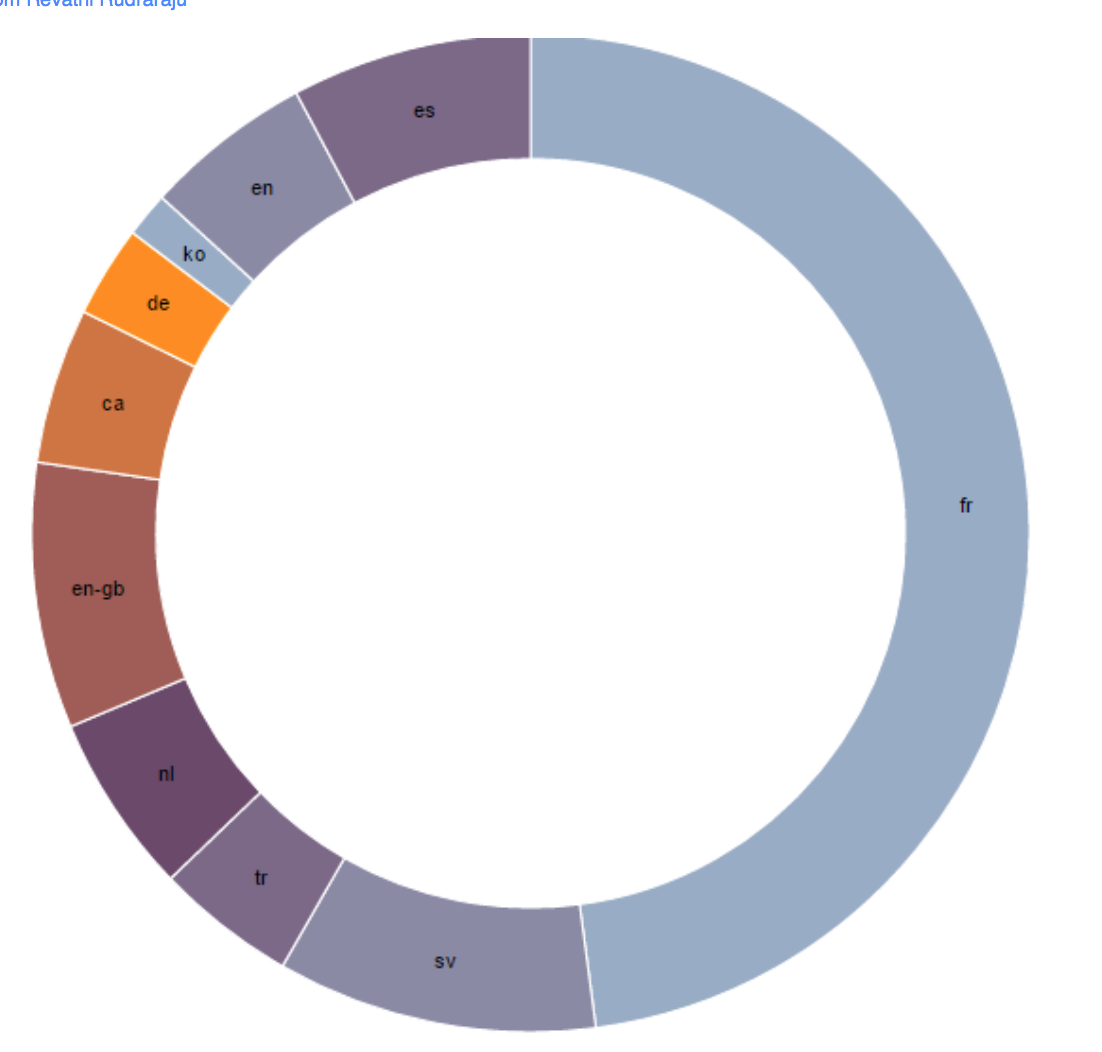
**

Query 5:

This query finds the languages in which they tweeted through users followers count and friends count.

*val query5= sqlContext.sql("select sum(user.statuses\_count), user.lang from tweettable where user.followers\_count>4000 and user.friends\_count>2000 and user.lang is not*

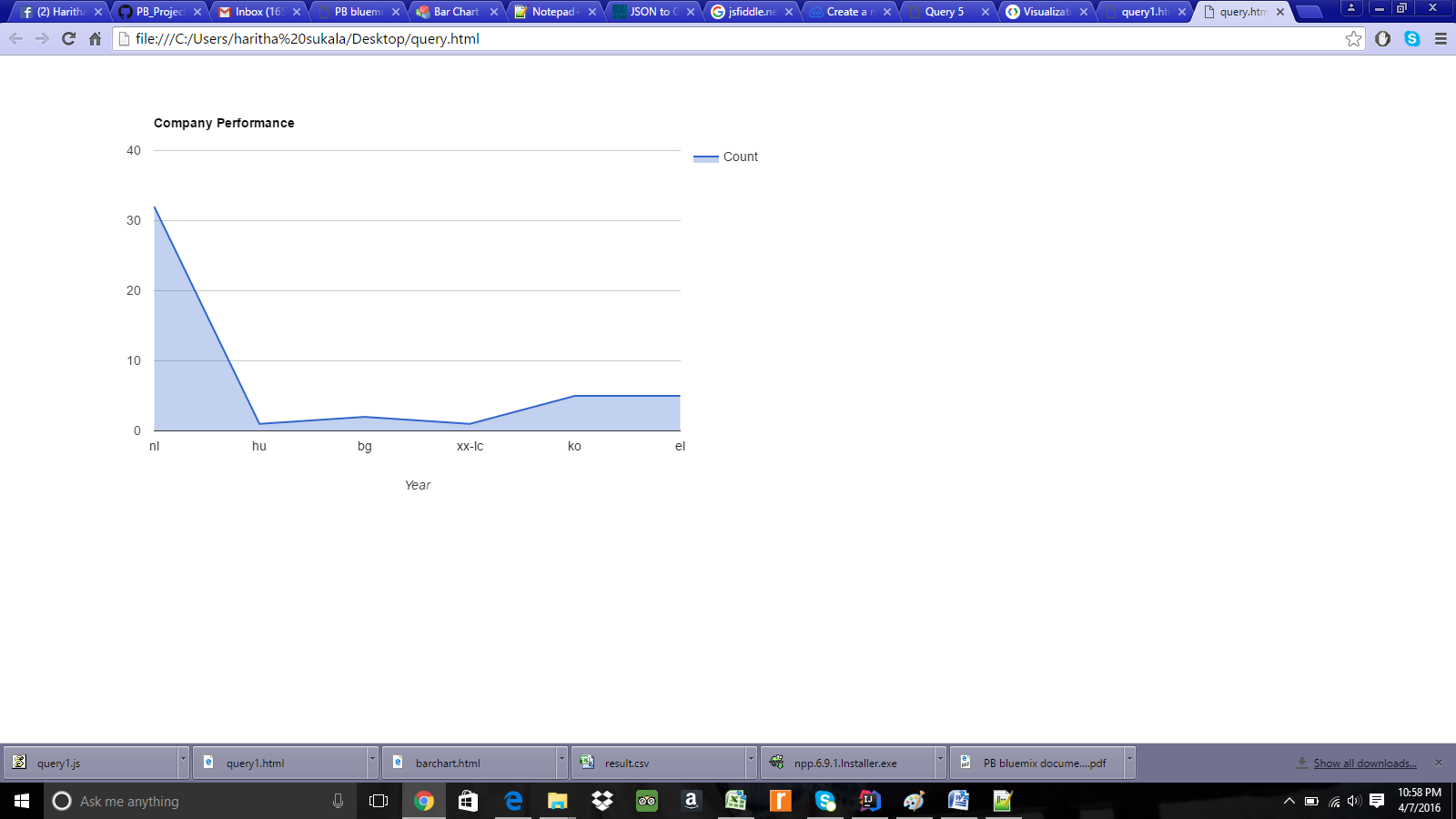
*null and user.statuses\_count is not null group by user.lang limit 100")*



Query 6:

This query finds the top 7 languages that users use to tweet and there count .

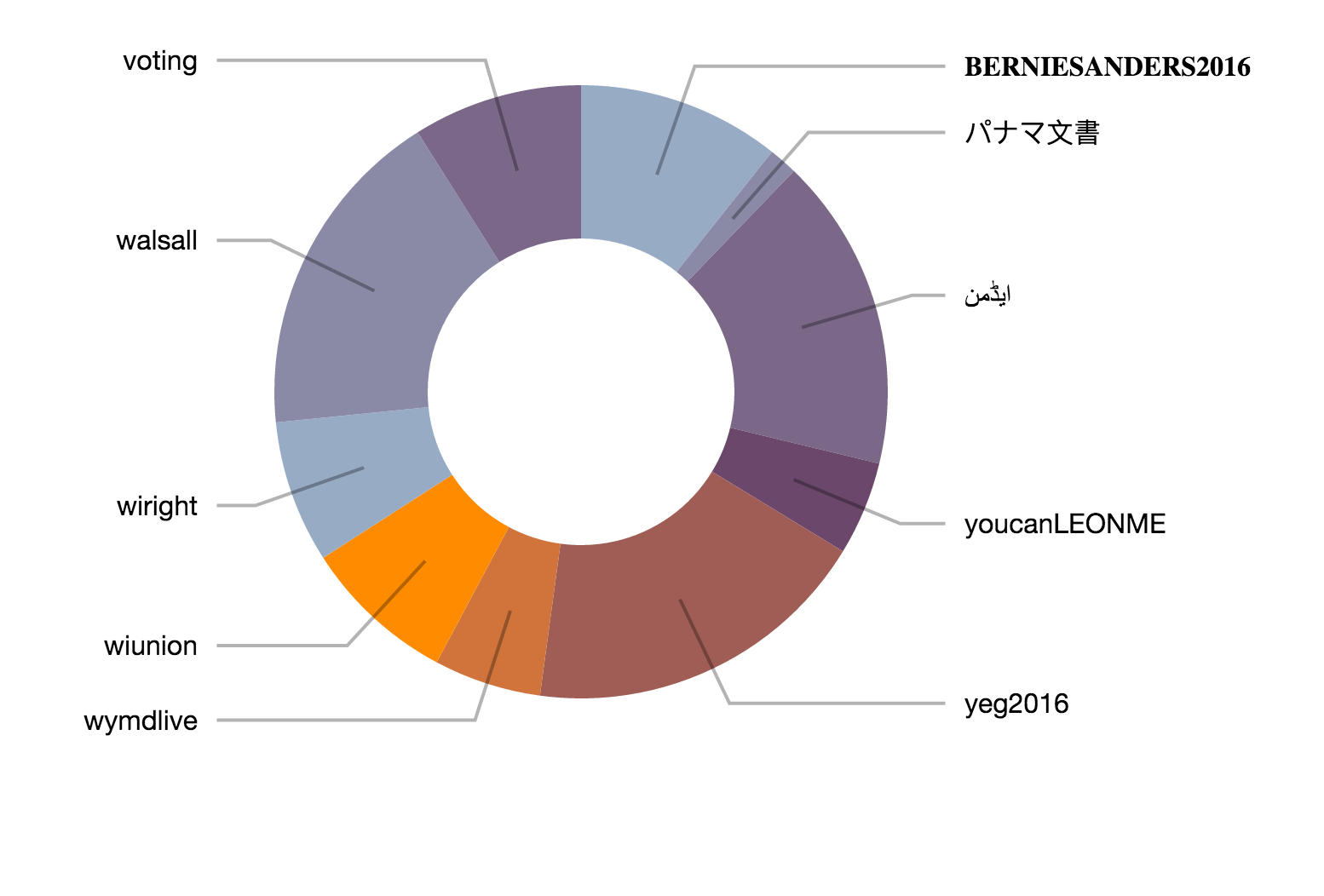
*val query6= sqlContext.sql("select distinct(user.lang),count(\*) from tweettable group by user.lang limit 7")*

**

Query 7:

This query tells the most famous hashtags regarding the elections.

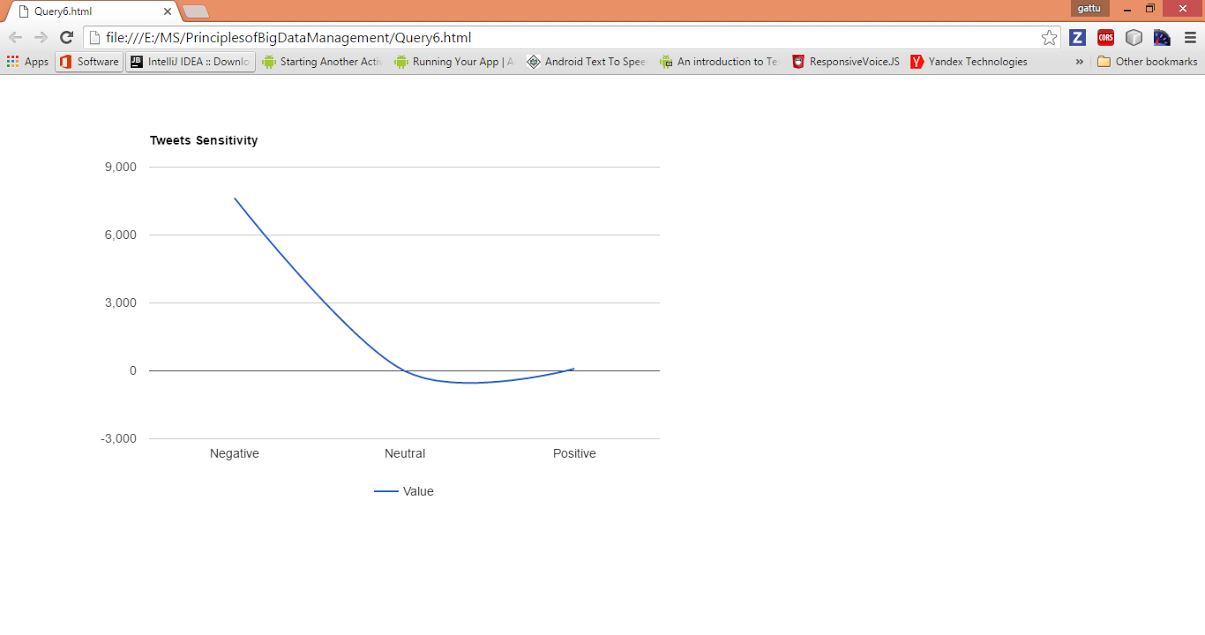
*val query7= sqlContext.sql("select entities.hashtags[0].text as famous\_tags from tweettable group by entities.hashtags[0].text order by famous\_tags desc limit 10")*



Query 8:

This query finds tweets and its count which are possibly sensitive

*val query8= sqlContext.sql("select possibly\_sensitive as sensitive,count(\*) as cnt from tweettable group by possibly\_sensitive limit 3")*

**