

# BIA-660 Midterm Project

Analytics & Comparison of Apple Products with Twitter

By:  
Colin Mills  
Qiang Tong  
Xiangda Shi  
Xin Gao

Instructor:  
Prof. Winter Mason

# Questions

## Popularity Trends of Apple Products.

- Which Apple mobile OS is more popular?  
iOS4 ~ iOS7.
- Which generation of iPhone is more popular?  
iPhone 3GS ~ iPhone 5S.
- iPhone 5S vs. iPhone 5C?
- Most-used words for each product.



# Headlines

- Data Collection
- Data Analysis
- Data Visualization

# Data Source and Tools

- Data Source:  
Tweets containing keywords(iOS 4, iPhone 5)
- Tools:  
Python, Mysql, Selenium, BeautifulSoup, Matplotlib



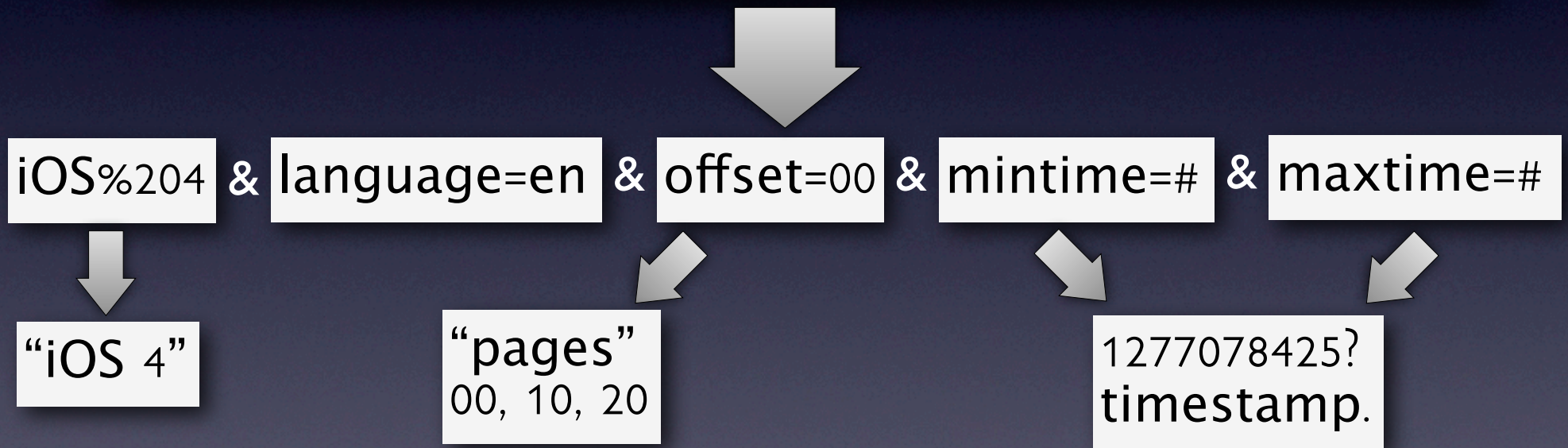
# Data Collection

- Why don't we use Twitter API?
- Twitter search URL:  
<http://twitter.com/search?q=>
- topsy.com/tweets search URL:

# Data Collection

- topsy.com/tweets search URL:

<http://topsy.com/s?type=tweet&q=iOS%204&language=en&offset=00&mintime=1277078425&maxtime=1308700814>



- Another problem, javascript?



# Data Collection

- “Selenium” to the rescue!

## HTML Code

```
<div class="media-body">
  <h5 class="media-heading">
    <a href="http://user_id_url">
      Mashable
    </a>
  </h5>
  <div>iOS 4 Has Arrived <!--Content--></div>
  <ul class="inline">
    <li>
      <a class="muted" href="http://tweet_url">
      </li>
    <li><!-- "Reply", "Forward"...--></li>
  </ul>
</div>
```

Selenium  
BeautifulSoup



## MySQL DB

iphone4_tweets	
🔑	tw_url: VARCHAR(255)
💎	usr_id: VARCHAR(45)
💎	usr_name: VARCHAR(255)
💎	usr_url: VARCHAR(255)
💎	tw_content: VARCHAR(150)
💎	tw_time: DATETIME

- More than 2,200 tweets each keyword.

# Data Analysis

- “Time” matters.

Only within 1 week of release

- Retrieve tweet content from database.
- Positive or Negative?
- Special Dictionary.

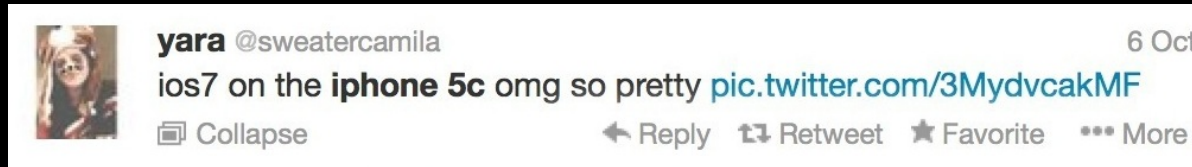
Pos\_words.txt

```
adorable accepted acclaimed  
accomplishment achievement active  
admire affluent amazing  
...  
...  
wow yummy zeal zealous
```

neg\_words.txt

```
abysmal adverse alarming angry annoy  
anxious apathy appalling atrocious  
awful bad banal barbed belligerent  
...  
...  
woeful worthless wound yell yucky
```





- “Time” matters.

Only within 1 week of release

- Retrieve tweet content from database.
- Positive or Negative?
- Special Dictionary.

Pos\_words.txt

```
adorable accepted acclaimed  
accomplishment achievement active  
admire affluent amazing  
...  
...  
wow yummy zeal zealous
```

neg\_words.txt

```
abysmal adverse alarming angry annoy  
anxious apathy appalling atrocious  
awful bad banal barbed belligerent  
...  
...  
woeful worthless wound yell yucky
```



**yara** @sweatercamila

6 Oct

ios7 on the **iphone 5c** omg so pretty [pic.twitter.com/3MydvcaKMF](http://pic.twitter.com/3MydvcaKMF)

Collapse

Reply

Retweet

Favorite

More



**11**  
RETWEETS

**13**  
FAVORITES



7:23 AM - 6 Oct 13 · Details

Flag media

- “
- 
- R
- P
- S

Pos\_wo

adorable acc  
accomplishme  
admire afflu  
...  
...  
wow yummy ze

se.

ng angry annoy  
ng atrocious  
belligerent  
yell yucky



# Data Analysis

- “Time” matters.

Only within 1 week of release

- Retrieve tweet content from database.
- Positive or Negative?
- Special Dictionary.

Pos\_words.txt

```
adorable accepted acclaimed  
accomplishment achievement active  
admire affluent amazing  
...  
...  
wow yummy zeal zealous
```

neg\_words.txt

```
abysmal adverse alarming angry annoy  
anxious apathy appalling atrocious  
awful bad banal barbed belligerent  
...  
...  
woeful worthless wound yell yucky
```

# Data Analysis

## Positive vs. Negative Sample Code:

```
conn=pymysql.connect('127.0.0.1',port,db_user,db_password,db_name)
str_sql_count='SELECT COUNT(*) FROM '+tbl_name
str_sql_get= 'SELECT TW_URL, TW_CONTENT FROM '+tbl_name'
cur = conn.cursor()
cur.execute(str_sql_count)
(all_count,) = cur.fetchone()
list_pos = []
list_neg = []
cur.execute(str_sql_get)
for row in cur:
    content = row[1].lower()
    for word in neg_words:
        if word in content:
            list_neg.append(content)
            break
    if word == neg_words[-1]:
        for word in pos_words:
            if word in content:
                list_pos.append(content)
                break
cur.close()
conn.close()
```



# Data Analysis

- Word Count
- `collections.Counter`
- list of tuples: `[(str_word, int_count), ..., ]`

# Data Visualization

positive/negative:

```
iPhone 3GS
Positive: 1048 / 3091 33.90489 %
Negative: 554 / 3091 17.92300 %
Positive/Negative: 1.89170
=====

iPhone 4:
Positive: 707 / 3027 23.35646 %
Negative: 571 / 3027 18.86356 %
Positive/Negative: 1.23818
=====

iPhone 4S:
Positive: 688 / 2758 24.94561 %
Negative: 554 / 2758 20.08702 %
Positive/Negative: 1.24188
=====

iPhone 5:
Positive: 633 / 2492 25.40128 %
Negative: 552 / 2492 22.15088 %
Positive/Negative: 1.14674
=====

iPhone 5S:
Positive: 615 / 2225 27.64045 %
Negative: 422 / 2225 18.96629 %
Positive/Negative: 1.45735
```

```
iOS 4:
Positive: 745 / 2552 29.19279 %
Negative: 530 / 2552 20.76803 %
Positive vs. Negative: 1.40566
=====

iOS 5:
Positive: 787 / 2491 31.59374 %
Negative: 451 / 2491 18.10518 %
Positive vs. Negative: 1.74501
=====

iOS 6:
Positive: 843 / 2611 32.28648 %
Negative: 492 / 2611 18.84336 %
Positive vs. Negative: 1.71341
=====

iOS 7:
Positive: 729 / 2374 30.70767 %
Negative: 595 / 2374 25.06318 %
Positive vs. Negative: 1.22521
```

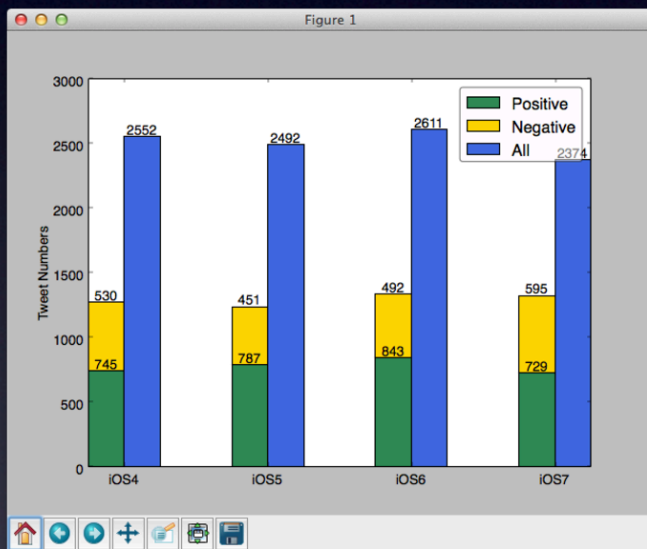
```
iPhone 5S Only:
Positive: 565 / 2207 25.60036 %
Negative: 478 / 2207 21.65836 %
Positive vs. Negative: 1.18201
=====

iPhone 5C Only:
Positive: 619 / 2234 27.6996 %
Negative: 352 / 2234 15.75649 %
Positive vs. Negative: 1.75852
```

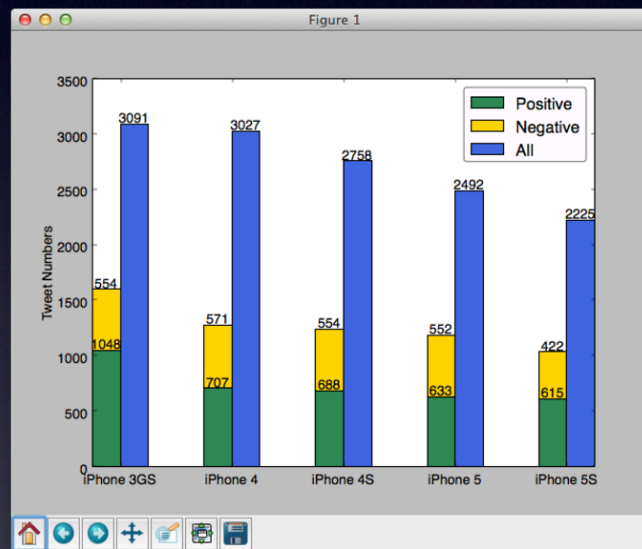


# Data Visualization

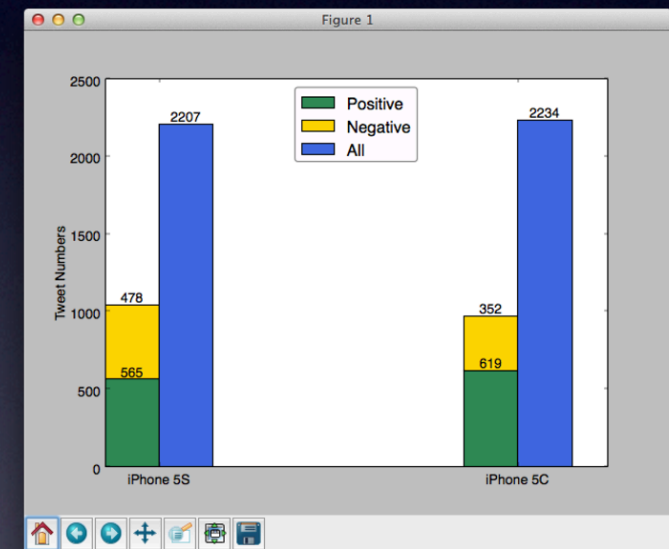
## Matplotlib - bar charts



iOS 4/5/6/7



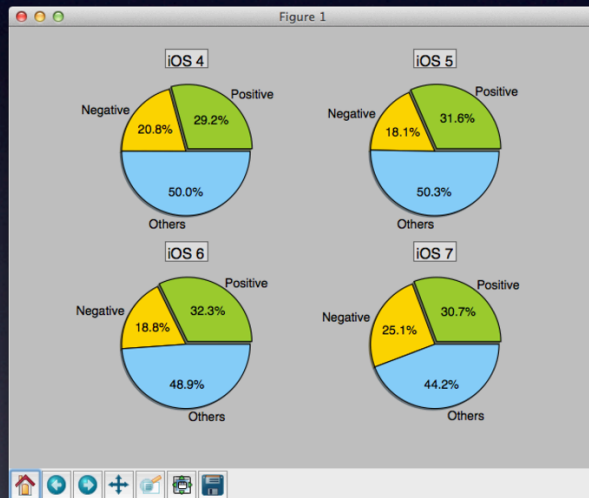
iPhone 3GS/4/4S/5/5S



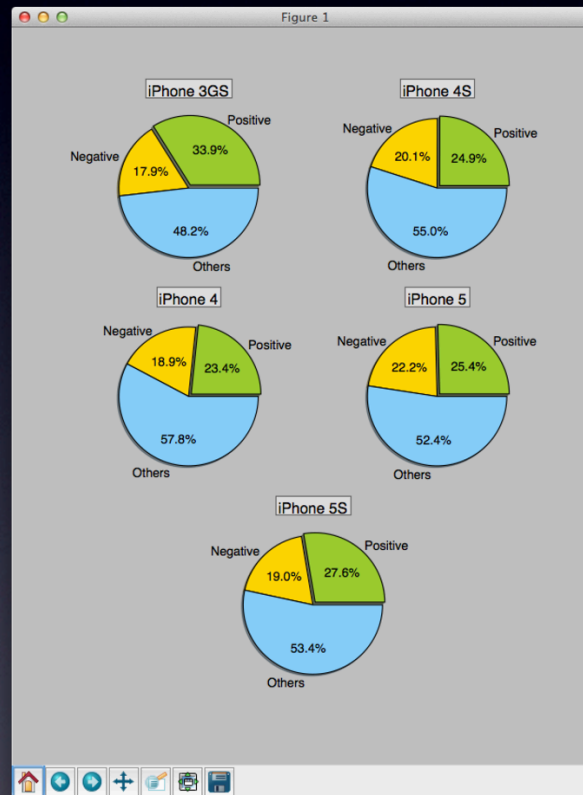
iPhone 5S/5C

# Data Visualization

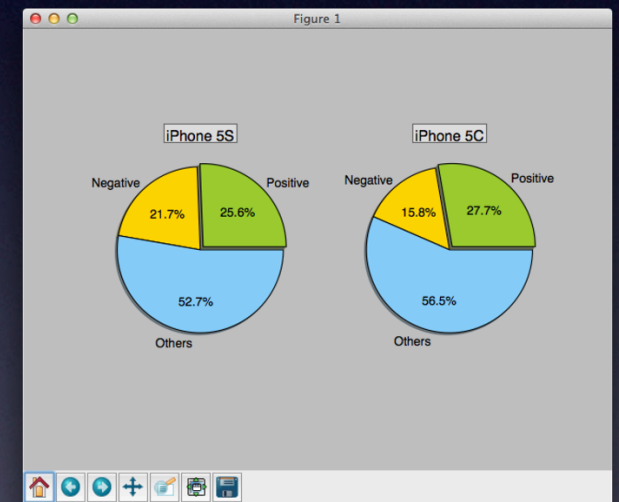
## Matplotlib - pie charts



iOS 4/5/6/7



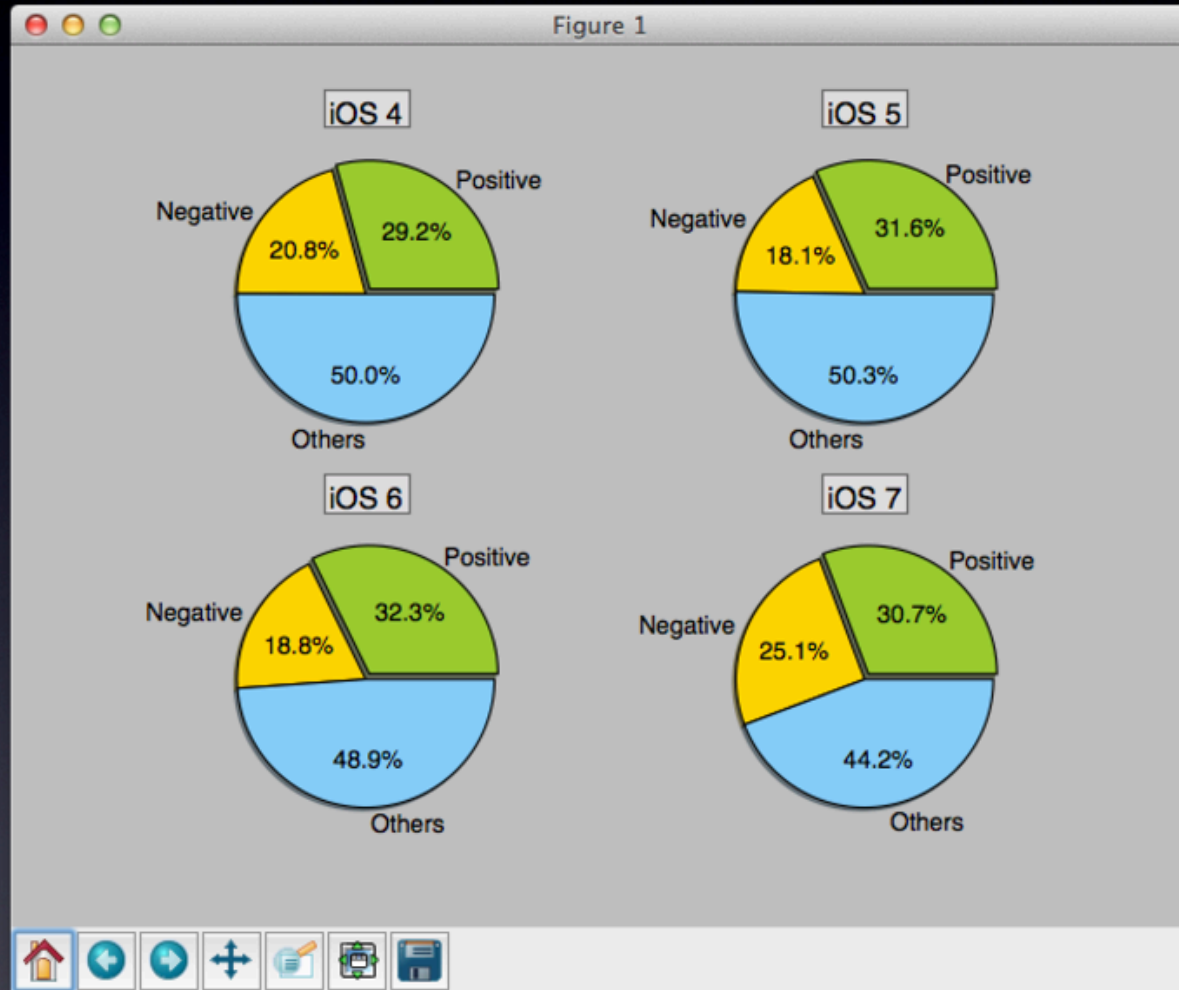
iPhone 3GS/4/4S/5/5S



iPhone 5S/5C

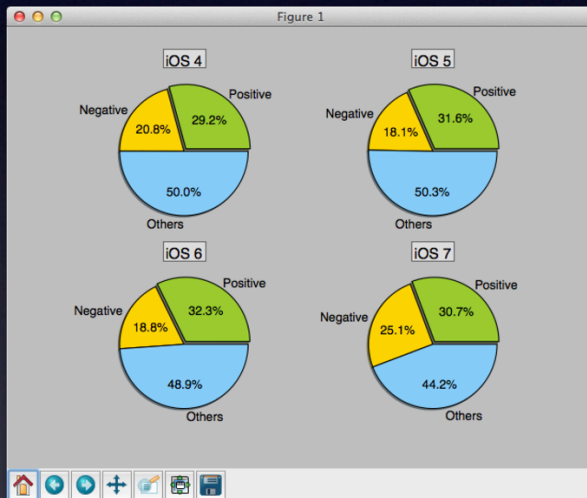


# Data Visualization

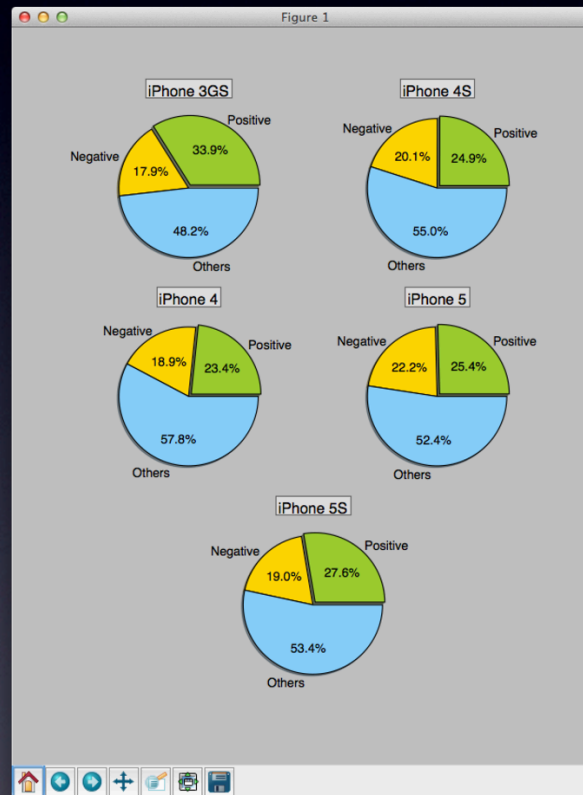


# Data Visualization

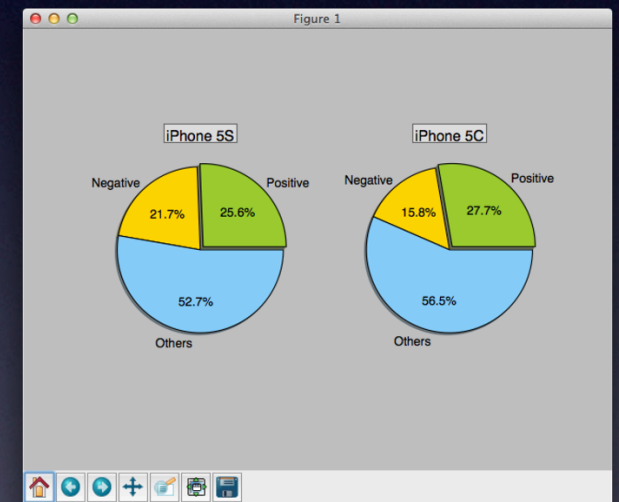
## Matplotlib - pie charts



iOS 4/5/6/7



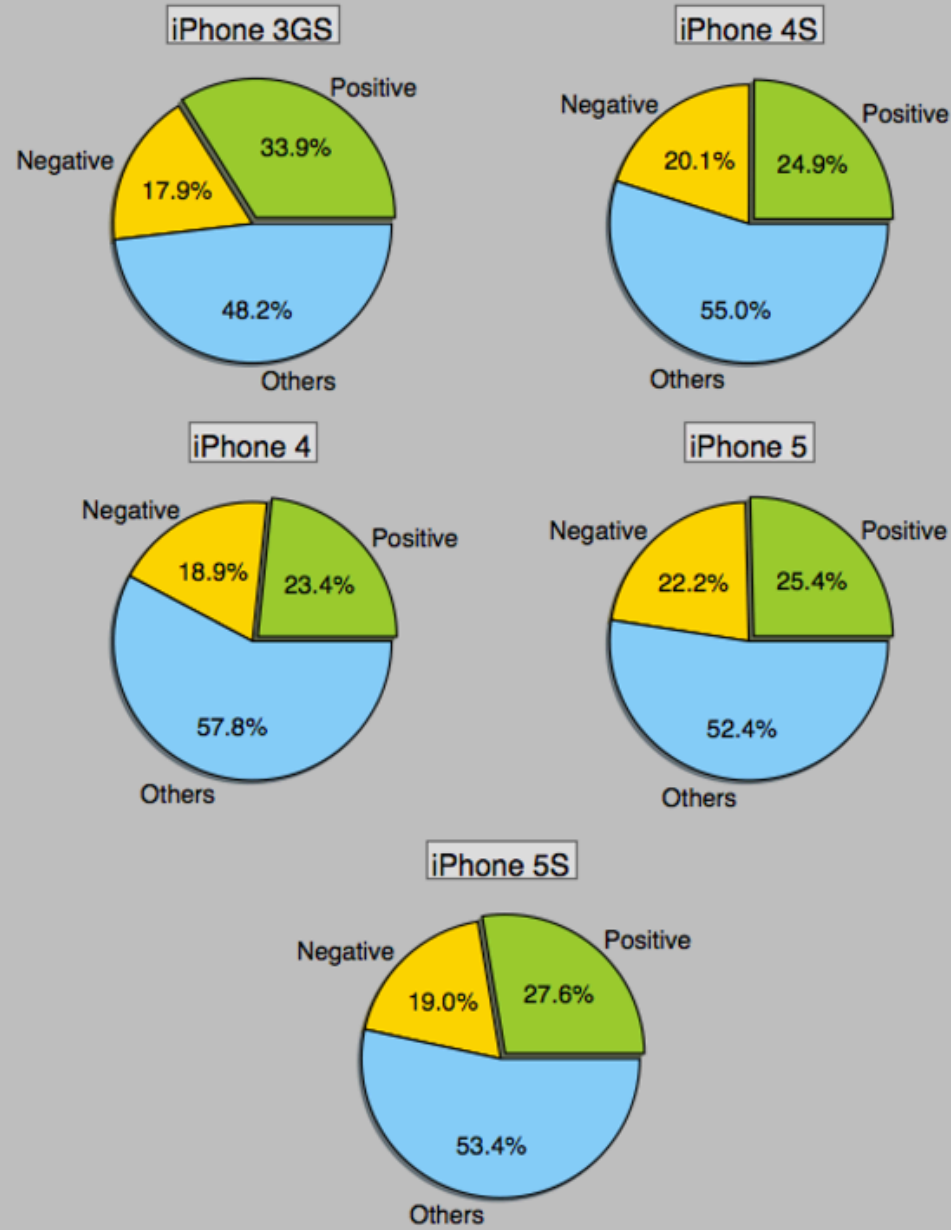
iPhone 3GS/4/4S/5/5S



iPhone 5S/5C

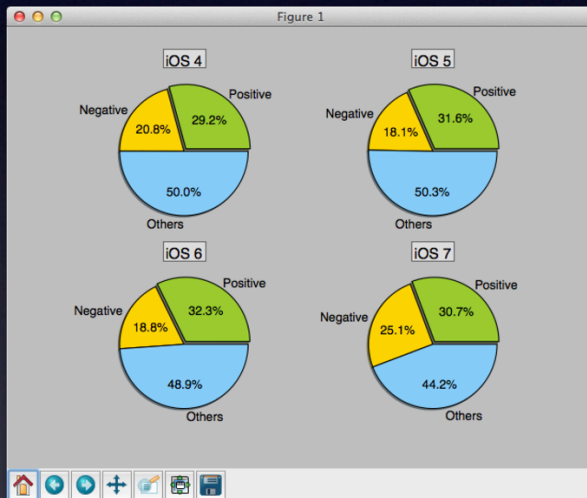


Figure 1

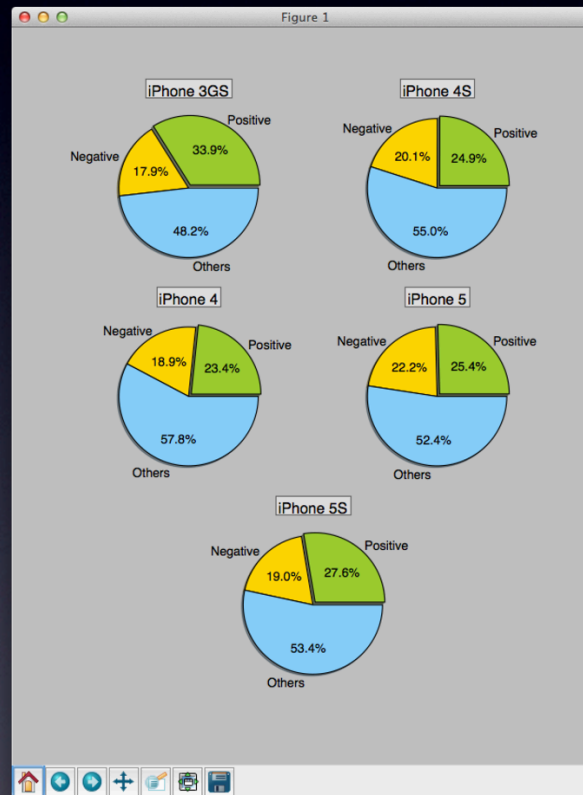


# Data Visualization

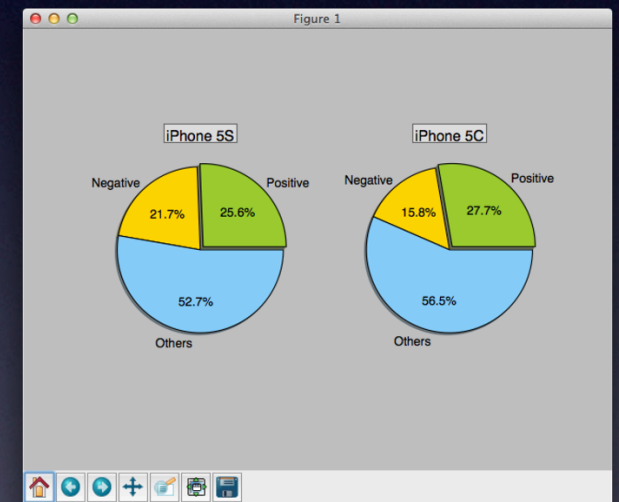
## Matplotlib - pie charts



iOS 4/5/6/7



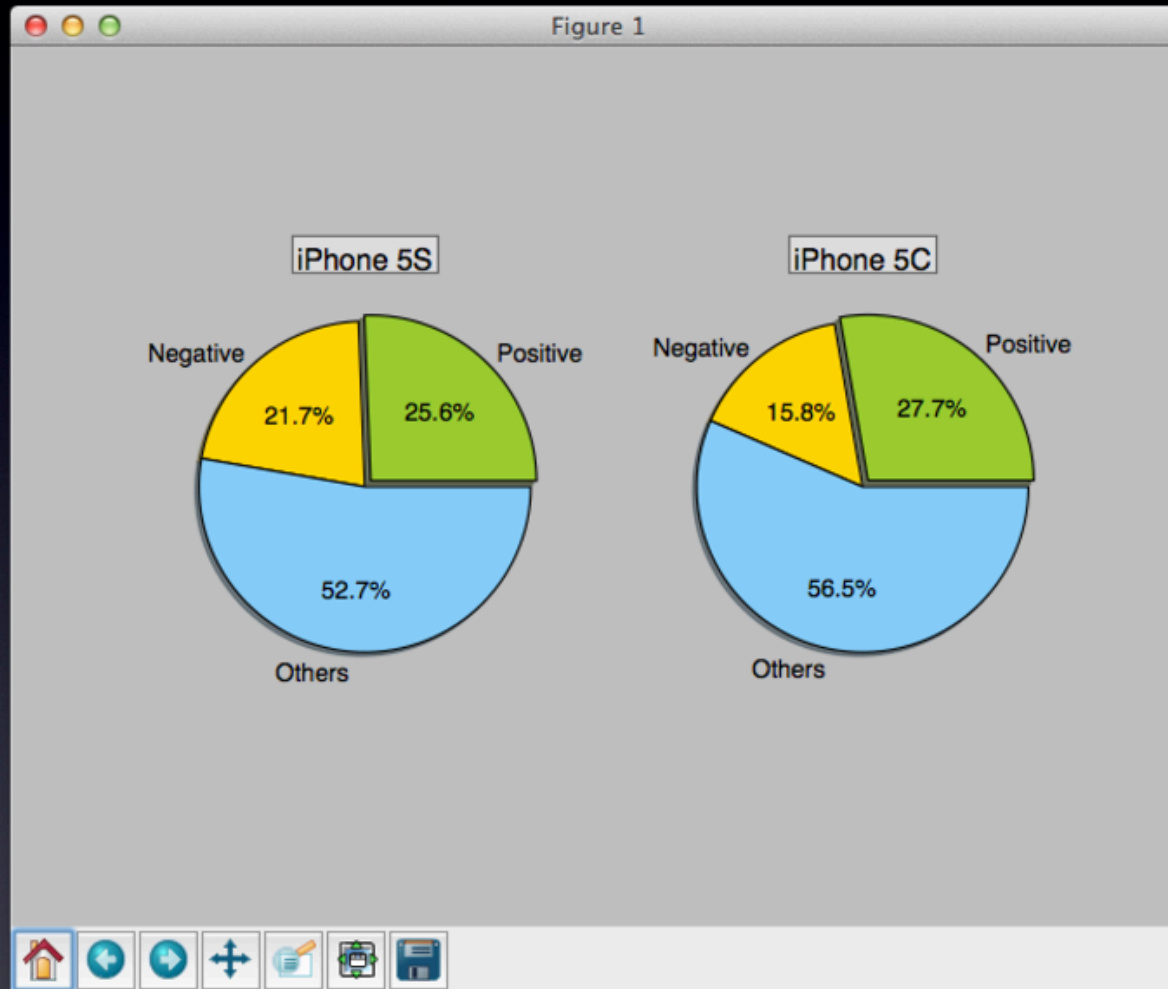
iPhone 3GS/4/4S/5/5S



iPhone 5S/5C

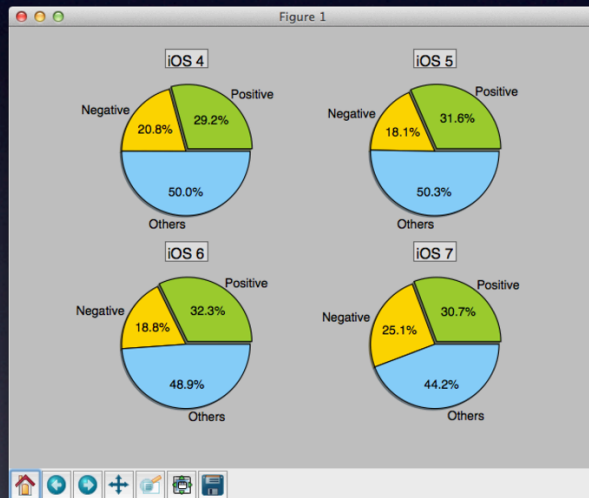


# Data Visualization

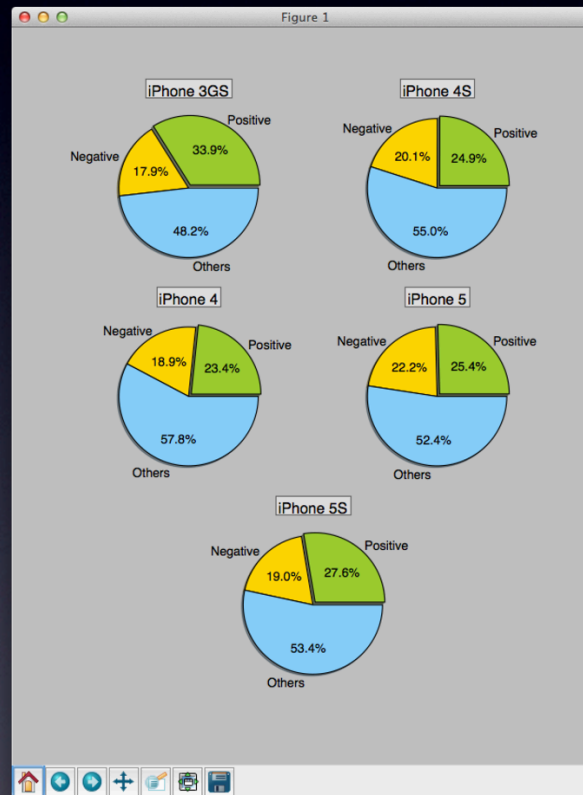


# Data Visualization

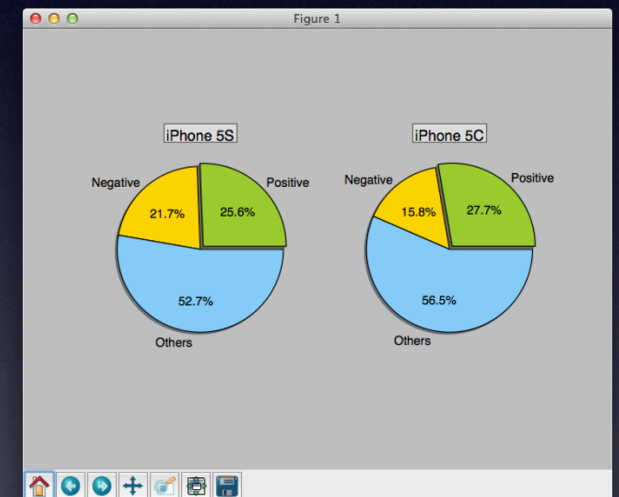
## Matplotlib - pie charts



iOS 4/5/6/7



iPhone 3GS/4/4S/5/5S



iPhone 5S/5C



# Data Visualization

## Keywords about iOS

iOS 4:

multitasking, jailbreak,  
facebook, 3g

iOS 5:

wp7, windows, jailbreak,  
mango, android, leak, icloud

iOS 6:

maps, siri

iOS 7:

battery

# Data Visualization

Wordle™ for iOS 4





# Data Visualization

## Keywords about iPhone

iPhone 3GS:

squarespace, att, win, 3g,  
faster, twitition, reasonable

iPhone 4:

att, best, case shipping

iPhone 4S:

siri, samsung, wp7

iPhone 5S:

Guess?

iPhone 5C:

Guess?

# Data Visualization

## Keywords about iPhone

iPhone 3GS:

squarespace, att, win, 3g,  
faster, twitition, reasonable

iPhone 4:

att, best, case shipping

iPhone 4S:

siri, samsung, wp7

iPhone 5S:

gold(211), fingerprint(144)

iPhone 5C:    Guess?



# Data Visualization

## Keywords about iPhone

iPhone 3GS:

squarespace, att, win, 3g,  
faster, twitition, reasonable

iPhone 4:

att, best, case shipping

iPhone 4S:

siri, samsung, wp7

iPhone 5S:

gold(211), fingerprint(144)

iPhone 5C:

plastic(137), cheap(102)

# Data Visualization

Wordle™ for iPhone 4S





# Conclusions

Q&A  
Thank you!

