*Rogelio Zuniga R.*

# Travel Reviews ...

March 2016 | DAT-30

# The Question

- "What is the relationship between perceived destination cleanliness and travel to that location?"
- "How much can we trust social media reviews?"

# The Data: Twitter

- **Twitter API search**: Geolocation coordinates and key words determined to return relevant comments :

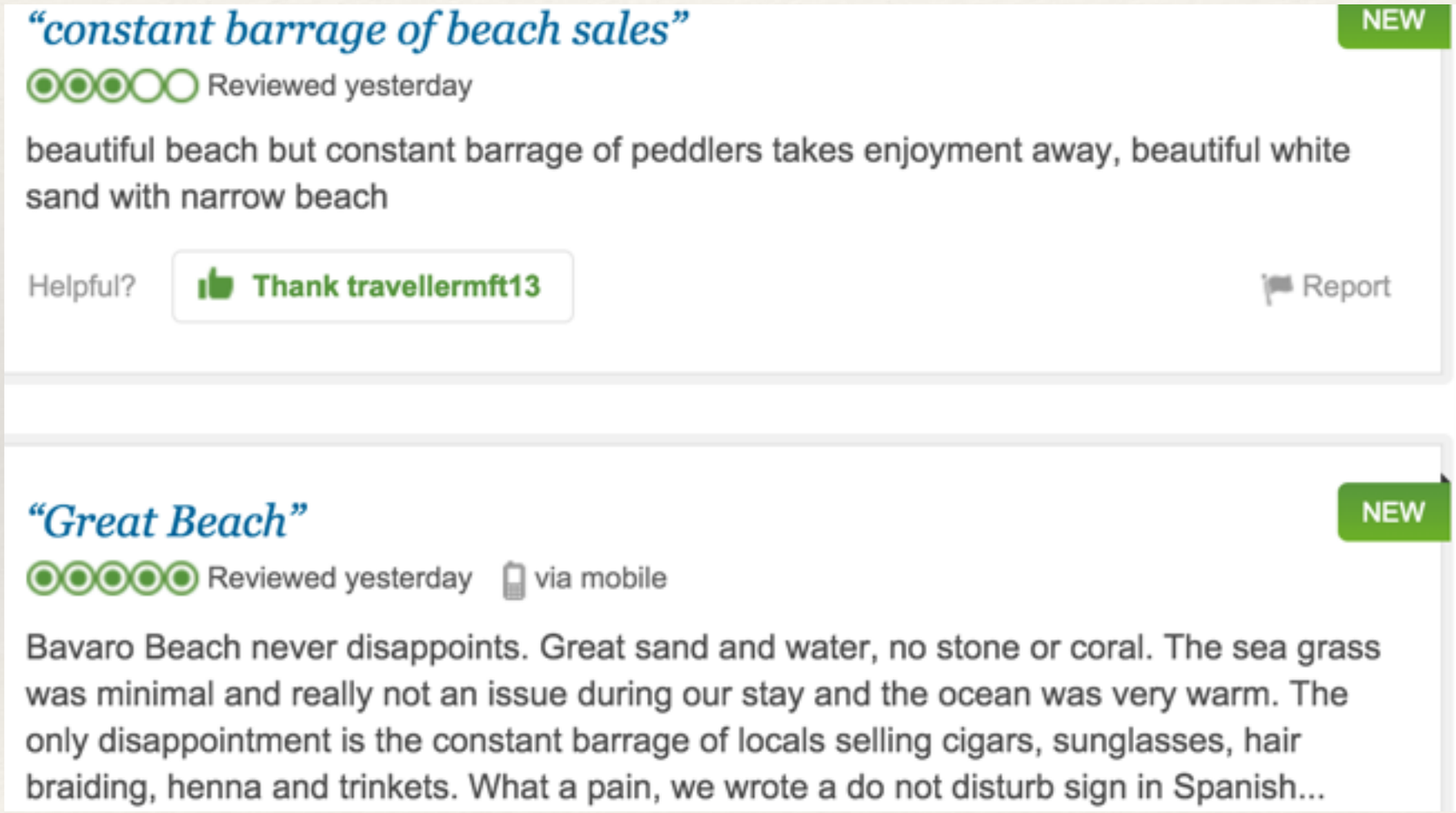
"https://twitter.com/search?q=beach%20trash%20jamaica&src=typd"

**Dr. Claire Nelson** @DrClaireNelson · 12 Nov 2015

Now that would be nice.. I am sick of seeing people leave their **trash** on the **beach** in **Jamaica**. We mustban Styrofoam

# The Data: TripAdvisor

- **TripAdvisor**: TA "place" codes



# Exploring Social Media is Hard!

**Twitter: search API using TwitterSearch**

- Geocodes don't help
- Keywords lead to irrelevant returns
- Very hard to search anything except new tweets

# Exploring Social Media is Hard!

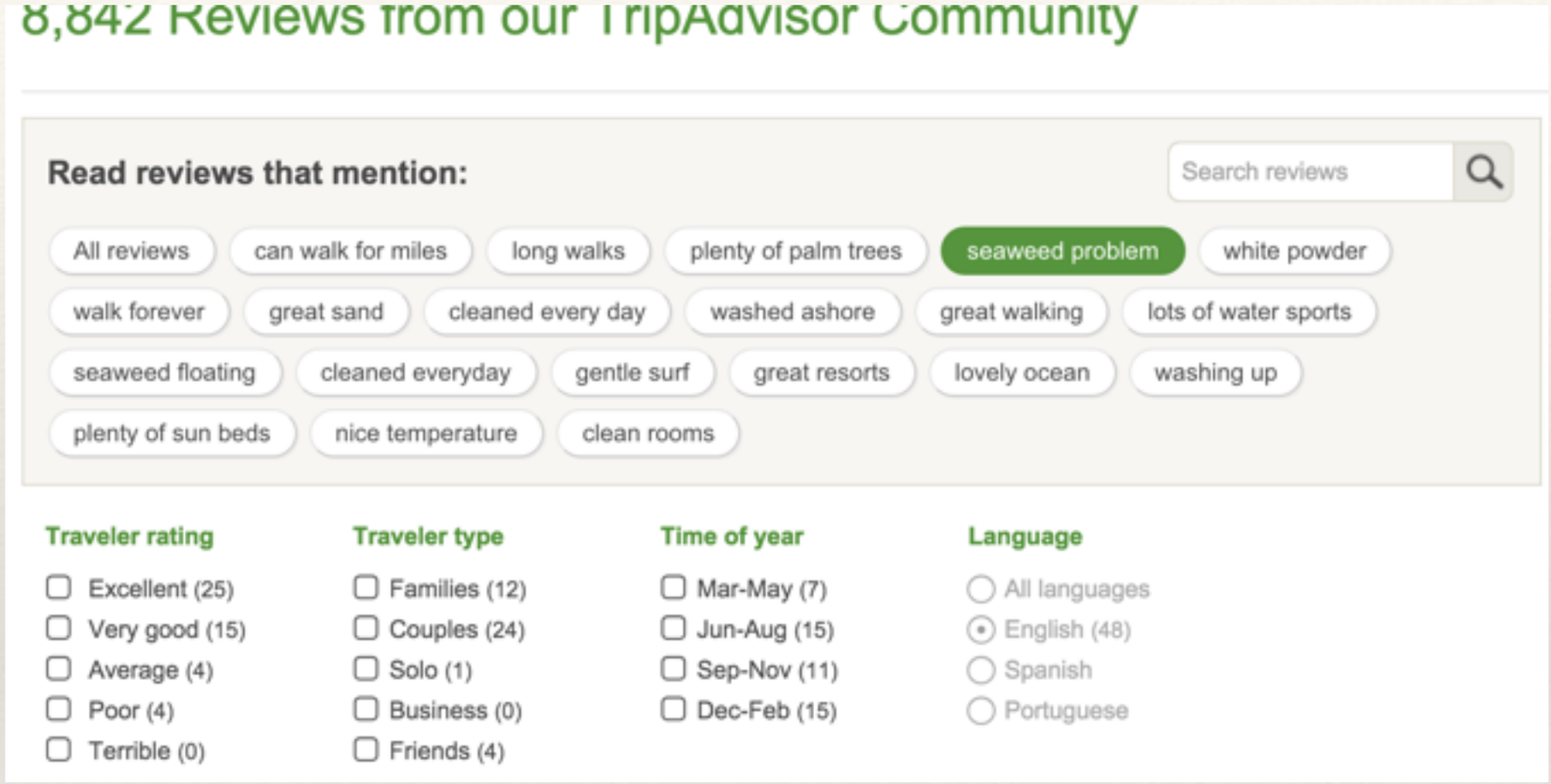
**TripAdvisor: direct from the website using BeautifulSoup**

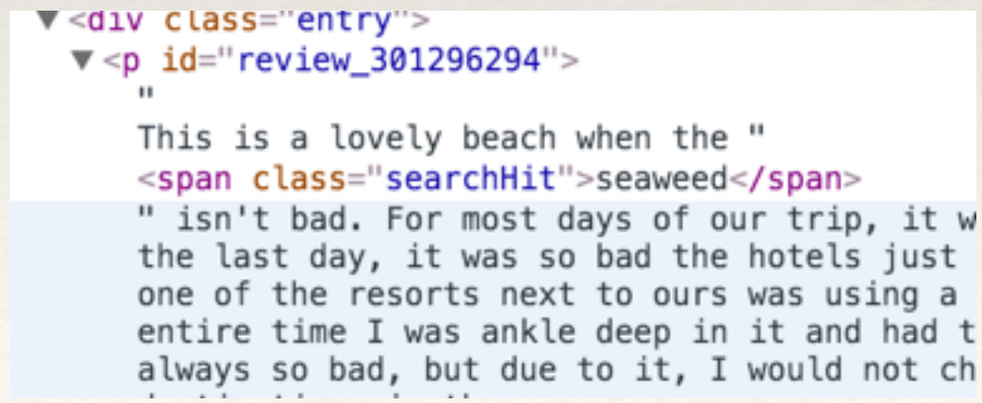
- Keywords lead to irrelevant returns
- Huge results!
- Need to organize a lot of results
- Not allowed to use API for mining data!

# Yikes!

- Without being able to search Tweets far into the past I can't compare sentiment scores over time!
- On the other hand TripAdvisor reviews are far better qualitatively!

# Better Quality but More Work



# Decisions…

- Use only TripAdvisor?
  - Decided to use only TripAdvisor reviews
- How to deal with reviews that include keywords but are not relevant? Naive Bayes?
  - Use LDA and k-means to see if there are relevant topics
- How long will it take to gather enough data, scrape it and process it?
  - It takes much longer than expected to setup a clean data set

# Data Cleaning

- **Download** of html files to local disk via wget:
  - wget --convert-links --domains www.tripadvisor.com --recursive --no-parent https://www.tripadvisor.com/Attraction_Review-g147365-d2198364

# Data Cleaning

- Parse with **BeautifulSoup**:
  - ratingDateResults = nextTag.find(‘span','relativeDate')
  - nextReviewTag = nextTag.find(‘p’,'partial_entry')

| | place_code | destination_code | date | review |
|---|---|---|---|---|
| **0** | g1006573 | d1102010 | February 27 2016 | |
| **1** | g1006573 | d1102010 | February 27 2016 | |
| **2** | g1006573 | d1102010 | February 13 2016 | This is an experience that should not be misse... |
| **3** | g1006573 | d1102010 | February 13 2016 | |
| **4** | g1006573 | d1102010 | February 13 2016 | The horses are well cared for and well behaved... |
| **5** | g1006573 | d1102010 | February 13 2016 | I had the most wonderful time riding with Pamp... |
| **6** | g1006573 | d1102010 | February 13 2016 | We have horses at home and wanted to ride on a... |

# Data Cleaning

- The resulting data set is made up of:
- **A place code**: g1006573
- **A destination code**: d1102010
- **A date**: February 27 2016
- **A review text**: "The guide and ride was great. Communication was not the best. They use public transportation and didn't make then clear to use (how are we supposed to know what bus to get on). Make sure that they make it very clear to you."