Social media and brand marketing in the hotel industry

Holly Waisanen-Hatipoglu, 20 May 2014

Brands are increasingly looking to build a deeper engagement with their customers and view social media an opportunity to create ongoing conversations about the brand, its products, and the customers' needs and wants. Some brand conversations are initiated by customers, usually revolving around customer service or reviews. Other conversations are initiated by brand marketing organizations and try to drum up buzz around existing marketing programs or new campaigns developed specifically for social channels. This study will examine this second kind of conversation and attempt to understand the effectiveness of brand led campaigns at generating this buzz and fostering genuine customer response.

In this study, we examine of how the top brands in the hotel industry are using Twitter and attempt to measure the customer response. A Python script running on an Amazon EC2 instance collected Twitter data for all references to a few selected hotel brands over a three week period between April 20 and May 11, 2014. Our initial analysis of this data set attempts to quantify who is driving the conversations (brands, hotels, or customers), which conversations gain traction, and derive overall sentiment.

The primary contribution of this work is tutorial in nature as we document the steps undertaken to collect and manipulate the Twitter data. All original code is available on my github at: https://github.com/dblosqrl/hoteltweets. We also present a few initial business results and several directions for next steps.

Technical process and tutorials

The primary technical exercise of this study is in obtaining and shaping Twitter data, including some preliminary sentiment analysis. I do love to write detailed tutorials for fellow Linux newbies and hope to get a few more up on my blog, but for the purposes of meeting deadlines, please excuse external links and glossed over details below.

Getting started with the Twitter API

At the time of this writing, Twitter has two publicly available APIs: REST and Streaming. A conceptual overview of the differences is provided here: https://dev.twitter.com/docs/api/streaming.

REST API calls are driven by user requests (e.g. one-time request for search results) or interaction with an app (e.g. Twitter widget on a blog). The focus is on simplicity and relevance, not exhaustiveness. For example, using GET search/tweet to search for recent tweets according to some filter returns a maximum of 100 tweets from the last 7 days – searching on Twitter.com directly usually returns more and older tweets. These 100 tweets may not be the most recent either – Twitter performs some scrubbing to determine relevancy of tweets. If you really need

everything (as in a large scale data capture), you really need Streaming. The Streaming API requires a persistent internet connection, and then returns all tweets as they occur.

Both APIs require OAuth and application setup, have similar basic filtering interfaces (REST can get more complicated), and return similar Twitter objects. Both are also subject to unknown rate limits – larger scale applications may require a paid Twitter feed.

Setting up the proper security to access Twitter via OAuth tokens and keys is the first step in accessing either Twitter API. To do this, go to https://dev.twitter.com/ and sign in with your usual Twitter account in the upper right hand side. Create a new application (with some dummy names and websites as needed), click on your app and go to the API keys tab. You'll see an API key and secret already set up. Scroll down and click on Generate Token. Now you have an Access Token and Access Token Secret. Capture these four very important numbers (and don't share them with anybody) to use in the script you'll write to access the APIs. You're ready to rock!

There are various libraries in both R and Python that can be used to simplify the engagement with Twitter. TwitteR and StreamR are often cited libraries in R for REST and Streaming respectively. We implemented the data capture for this project in Python using the TwitterAPI after experimentation with other packages that had apparently become obsolete with the latestTwitter API update. The joys of open source software – you must stay cutting edge or you lose everything.

TwitterAPI uses the four keys/tokens to set up the connection to Twitter and converted the returned JSON string to a Python dict which may be then converted back to a string and written out to a file. When reading back in from the file, eval(line) converts the string back into a Python dict. Thank you to Aaron Schumacher for suggesting TwitterAPI and providing the starter Python code (available on Github).

Setting up on Amazon EC2

Since the Twitter Streaming API requires a persistent connection, I opted to run the data collection on a free AWS EC2 Micro instance. At the time of this writing, the free tier allows unlimited monthly hours on asingle Linux Micro instance, plus 5 GB storage. Setting up a micro instance is as simple as signing in on the AWS management console here: https://console.aws.amazon.com/ with your usual Amazon login, entering some billing and launching an instance.

In liu of an original explanation, I will cite Aaron's blog post on setting up and EC2 instance. http://planspace.org/2014/01/25/easy-aws-ec2-ubuntu-quick-start/. Once you get to the ssh step to login from the terminal on your personal machine from within whatever folder has your ssh key, you should be all good. The scp command can be used to copy files between machines (initiated from whatever computer has the source file to be copied).

When executing code on the AWS instance, use the nohup command to ensure that the code will keep running after you logout. For example, use

$ nohup python myfile.py &

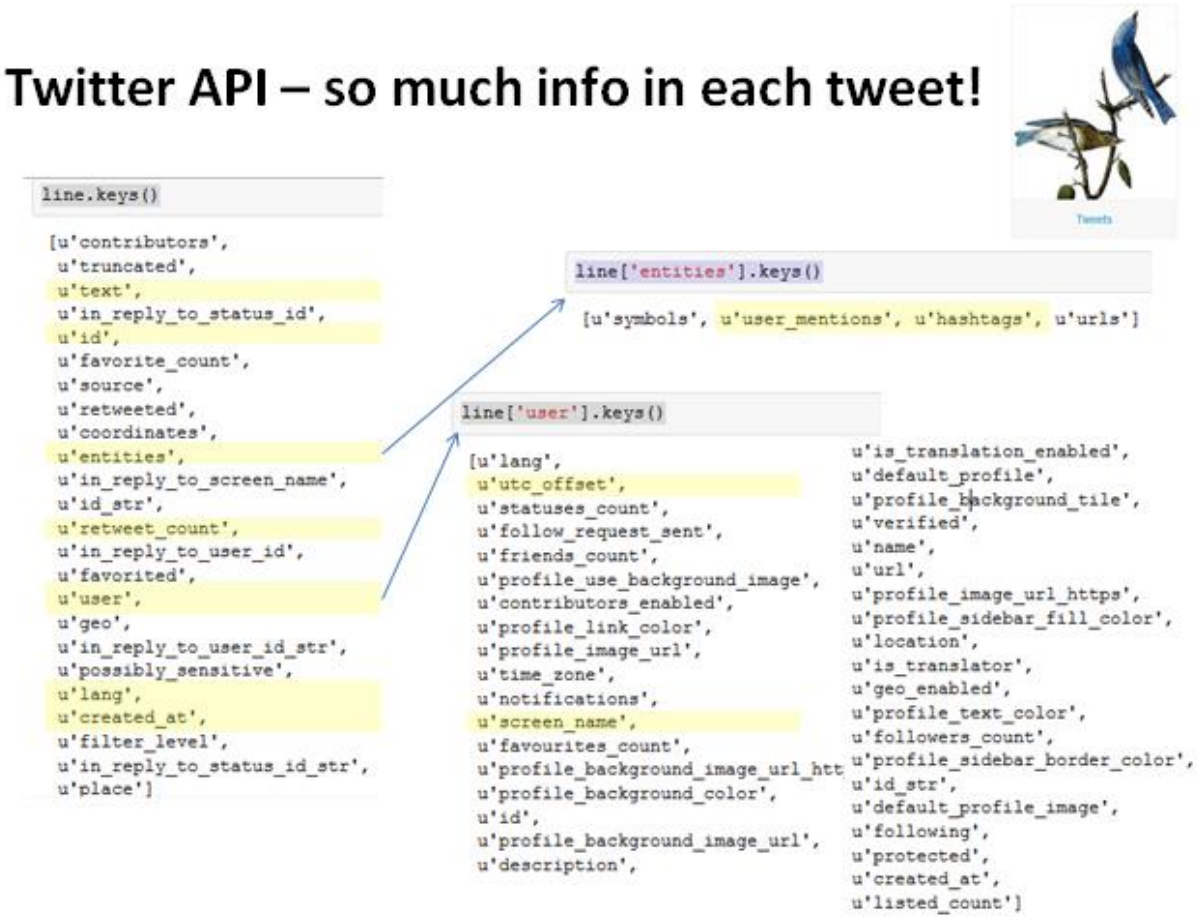
and then logoff at will. If/when you are ready to kill that job and stop capturing data, log back in and use

$ ps aux | grep py

to get a list of python jobs running and find the number (XXXX) you want to kill, then $ kill XXXX.

Cleaning and processing Twitter data

Once tweets were captured and the file scp'ed to my local machine, a Python script extracted a subset of fields from the captured tweets according to the documentation provided by Twitter at https://dev.twitter.com/docs. Detailed code is available on git, and the fields captured are illustrated below. The highlighted fields were selected for further analysis.



A few details on field usage are listed below.

- Only tweets tagged as English were processed.

- created_at captures date and time in UTC. I'd intended to use the user utc_offset to correct time zones and capture local user week part and day part, but in practice, utc_offset was often blank – more work to do there to scrub and complete.
- Hashtags and user mentions are parsed out by Twitter as lists in the entities field. The list of hashtags for each tweet was converted to a space separated string of words.
- User screen_name was set to lower case and the original text had commas replaced by spaces (for eventual writing to csv).
- A second processed text field was created to capture text in lower case with all punctuation removed.
- Finally, to compute text sentiment on the processed test, a corpus of positive and negative words (Hu and Liu - http://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html) was downloaded and a simplified "Bayesian" method was used to determine sentiment of each processed tweet text, e.g. sentiment = count of positive words minus count of negative words. Ideally, the processed text would have also removed stop words (like a and the) and stemmed word endings (like –ing and –ed) using standard NLTK stemmers. Having fewer unique words to classify could speed processing significantly.

Export to csv

Once all of the tweet variables were processed into lists, the lists were combined into a single pandas dataframe and then written to csv using the to.csv method. This csv was loaded into both R and Tableau for further analysis. In practice, additional analysis could have been conducted in Python, but further investigation of R and Tableau better supported my learning goals.

Initial business results

This analysis focuses on three competitors - Marriott, Hilton, and Starwood. Marriott and Hilton are the most similar in terms of number and breadth of properties, but Starwood is also recognized as a key player in upper upscale, luxury, lifestyle, and loyalty. Tweet handles followed in this test correspond to selected full service and luxury hotels as well as chain loyalty programs. Specifically the following terms (and related handles) were followed:

- Marriott, MarriottIntl, RenHotels, #rdiscover, #travelbrilliantly
- Hilton, HiltonWorldwide, HiltonHotels, HiltonHHonors
- Starwood, Sheraton, Westin, W, SPG,starwoodbuzz, #BeAWeekender
- Other brands: IHGRewardsClub, HolidayInn, HotelIndigo, CrownePlaza, #discoverIHG, HyattTweets

There are two sets of questions to answer in a full analysis:

1) What kinds of engagement is each brand driving? Are there detectable classes of tweets - giveaways, thought starters, pure branding messages, responses to consumer tweets about the brand, etc? Which tweets are being retweeted and which hashtags are being reused by consumers?

2) Do brands have meaningfully differentiable brand characters? Are they using certain words or types of tweets to engage? Is there a detectable difference in tone between the brands either in the tweets generated by the corporate entities or in the consumer tweets related to the brand?

A storyline of slides is captured below on initial findings, but in summary, there's some additional work to do in cleaning out non-hotel-relevant tweets (e.g. Hilton Hater for entertainment, SPG as Indian security agency). There are clear classes of both tweets and users that should be identifiable through more advanced clustering methods. For example, by visual inspection, users tend to be corporate entities, properties, paid affiliates, possibly unpaid affiliates / wholesalers, news agencies, or consumers. Once detected, what can we say about the relative quality and tenor of content from these different groups?

Fundamentally, are customers deriving some value from brand engagement in social media? It's too soon to tell.

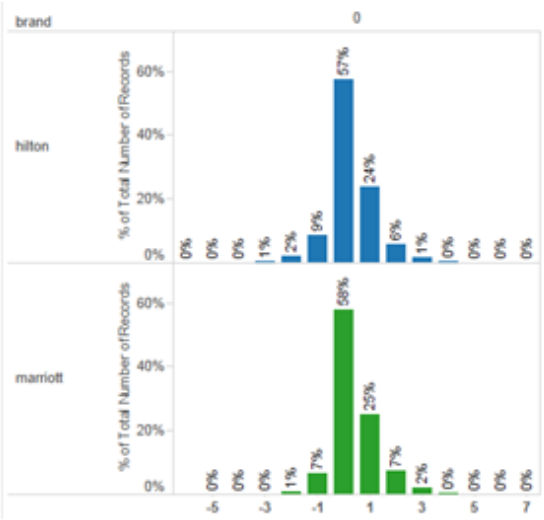
Top level engagement - which brands have the most volume and most positive sentiment?

Hilton has the most tweets, retweets and unique users in current tweet set

brand (group)	# of tweets	# of unique users	Retweets
hilton	82,075	52,869	18,412
marriott	54,768	31,992	10,881
starwood	33,932	21,203	8,832
hyatt, ihg, mixed and 2 more	18,362	11,576	2,625
Grand Total	189,137	111,323	40,750

Question: Is this representative of hotel related tweets or is there more scrubbing to be done?

Marriott has subtly more positive sentiment



Sentiment = # pos words - # neg words

Users - Who is actually doing all this tweeting?

Travel aggregators tweet the most – but entertainment related tweeters suggest more cleaning to do!

user	brand hilton	marriott	starwood
hotelscenes	987	448	10
vegas_visits	794	570	
hotelsyes		1,268	
travelpointers	482	303	117
dreamytravels		205	229
washingtonpist		382	
warshingtonpost		351	
hiltonhotelfans	271		
ehadshorosos	248		
ehadsmusic	248		
ehadstech	248		
alyson9082		234	
cgjase	85	93	43
propertiesreal	36	121	19
topratedhotels	38	125	7
travelair283	87	58	22
ehadsfood	163		
ehadsbeauty	162		
ehadsgossip	162		

Hotel brand tweeters show mix of corporate and property-level tweeters (and more cleanup needed)

user	brand marriott
marriott	123
marriottintl	66
marriottuk	59
atlmarriottmarq	44
keywestmarriott	29
detroitmarriott	25
marriottpov	23
lbmarriott	21
napamarriott	21
pmarriott	21
marriottcareers	19
marriottcolasc	15
jwmarriottlv	14
marriottresorts	14
marriottmanila	13
riyadhmarriott	13

user	brand hilton
hiltonhotelfans	271
hiltonhotels	88
hiltonworldwide	86
hiltonhelp	70
hiltonhhonors	53
bevhiltonjobs	40
hiltoncadets	40
laura_v_hilton	33
daphneyhilton	31
hiltonharwell_	29
hiltonmea	28
hiltonpattaya	28
hiltongrandvac	26
hilton_college	25
hiltonnairobi	25
hiltonsmythe	25
hilton_clwbeach	23
hiltonanatole	23
hiltonheadsc	23

Content –What's being retweeted? Which hashtags are catching on?

Retweets driven by beautiful pictures, inspirational quotes, twitter sweepstakes, and news items (some unrelated)

text	# of retweets
RT @AllyBrooke: Nothin' better than a great night of jazz! Thanks @WHotels ..	916
RT @foxtramedia: DoubleTree by Hilton Hotel Tarrytown http://t.co/aZcyYGp..	568
RT @foxtramedia: The Westin Resort http://t.co/JVCMUtZ3SV #Hotel #Travel	494
RT @Thegooglefactz: Best places to go for free WIFI; Mcdonalds Apple Stor..	394
RT @Inspire_Us: Success seems to be connected to action. Successful peop..	392
RT @Thegooglefactz: Need some free WiFi? The best places to go are Panera..	355
RT @OkeyBakassi_: Pounded yam -N8000 Soup- N6000 Wait! This Transcop ..	343
RT @OK_Magazine WIN @GodzillaMovieUK premiere tickets and overnight M..	278
Success seems to be connected to action. Successful people keep moving. T..	256
RT @DMVFollowers: The largest hotel in Washington D.C. opened up last we..	239
RT @Colts: T.Y. Hilton on Reggie Wayne's return in 2014: "He's going to sho..	227
RT @weRengland: COMPETITION: Win two VIP tickets to the Marriott London..	211
RT @obyezeks: Will YOU come join US today @3pm at the Unity Fountain op..	205
Null	186
RT @anilkapurk: #DamaadGate Why Robber Vadra given SPG security while t..	162
RT @washingtonpost: Marriott just opened a $520M hotel with 1 175 rooms in..	156
RT @obyezeks: Men who wish to join women's #BringBackOurGirls march to..	152
RT @MarkLeibovich: Media ethics panel at WHCD at Washington Hilton. 30 ..	125

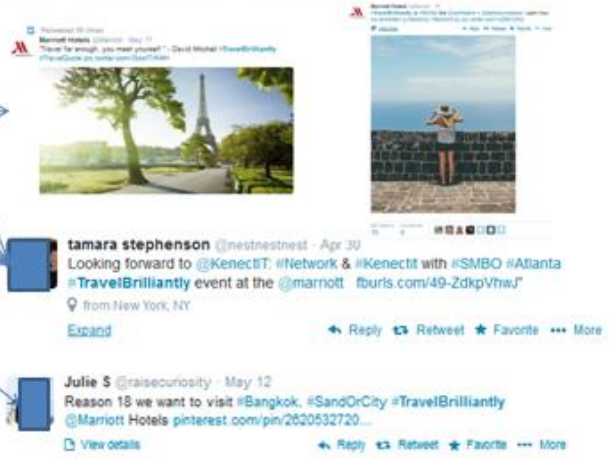
Top hashtags are brand names, "job(s)", travel words, and some promo tags

Hashtag	hilton	marriott	starwood
hilton	4,690	1	
marriott		3,900	
job	985	2,002	159
travel	1,168	702	733
hotel	1,170	526	678
jobs	943	1,187	136
gossip	1,532		
spg			1,086
westin			1,108
asicsrugbyrewards			
travelbrilliantly		1,057	
holidayinn			
hyattplace		1	
bringbackourgirls	611		
vegas	413	278	35
hotels	214	161	111
traffic	2	2	518
vacation	240	199	62

Deep dive on #TravelBrilliantly

- Travel Brilliantly is Marriott's current rebranding campaign
- Biggest volume of tweets using #TravelBrilliantly come from Marriott and paid affiliates – but how can we dig into the tail to identify meaningful value for individual customers?

user (group)	Number of Records	Retweets
nestnestnest	42	17
marriott	37	142
atlmarriottmarq	34	21
kenectit	34	0
marriottuk	20	44
mtgsimagined	17	9
hotelsyes	16	0
worldjourneys75	15	0
bookingcatch	14	0
01582242424, 05singingidiot..	853	356
Grand Total	1,082	589



Learnings and next steps

Social media processing is no joke – whole papers can and have been written on scrubbing Twitter data. A more complete analysis requires additional scrubbing to ensure that only hotel-relevant tweets remain. I also needed to think more carefully about streaming vs batch processing and data structures for scalable text analysis. More sophisticated cluster analysis and streamlined sentiment analysis becomes possible once the set of tweets is in a Term Document Matrix, but R's tm package was too slow to handle the given data set and I had insufficient time to work with Python's NLTK.

That script is still capturing tweets on my AWS EC2 instance. Immediately, I would like to add search terms and develop a cadence around pulling files down off the EC2 instance so that I do not hit storage limits. Much of the batch processing (processing text, etc) could also be implemented directly in the data capture script to save file sizes and batch processing time.

From a technical perspective, there's significant room for more sophisticated analysis around sentiment, clustering (detect types of tweets and users), and network analysis (who is influential). Some of this is advanced visualization of the existing data (Gephi?), and some requires additional packages (such as LDA for clustering and topic analysis). An interesting further application would be to generate monitoring to detect new promotional hashtags or changes in tweet strategy over time. The current three week test is insufficient, but a repeatable analysis pattern could be implemented.

I hope that someone has derived some value from this documented mishmash of learnings on data, software, Twitter, marketing, and hotels. I see this as an intermediate step in an ongoing project – follow my github if interested to see where it goes.