

VADER Sentiment Analysis: Measuring the happiness of Twitter users

Introduction

This paper is a summary of a statistics experiment conducted to analyze Twitter data. The purpose of this experiment is to compare the happiness of two groups of Twitter users by analyzing their tweets. Specifically, my question was: are people that are vocal about their physical activity happier than people who are vocal about their media consumption? The same question phrased differently: are people that talk about fitness happier than people who talk about media (tv, movies, internet videos, etc.)?

Hypothesis

Null hypothesis: the happiness of people that talk about fitness is equal to (not different from) the happiness of people that talk about media.

Alternate hypothesis: the happiness of people that talk about fitness is greater than the happiness of people that talk about media.

Defining happiness

To test this hypothesis, the definition of happiness should be clarified. According to the Oxford American dictionary, a person that is happy is “feeling or showing pleasure or contentment”. In the context of Twitter, emotions are conveyed primarily through written language (at least in the scope of this project). Users post photos and videos as well, but we are analyzing the sentiment of language to determine happiness. This begs another question: can a person show pleasure or contentment and yet be unhappy? Yes, this is possible – but common-sense dictates that for most people, this is usually not the case.

For this experiment, happiness is defined as a linear function of sentiment polarity. This is to say, happiness is directly correlated to the positivity, neutrality, or negativity of a person’s expression. A person who expresses positive feelings and thoughts is more likely to be happy than a person who expresses non-sentimental (neutral) or negative feelings. And a person that

expresses negative thoughts and feelings is more likely to be unhappy than a person who expresses neutral or positive sentiment.

Measuring happiness

Given that Twitter is a platform intended primarily for written expression, our focus is on analyzing language. One method of gauging happiness is to measure the sentiment (polarity) of tweets. This is called sentiment analysis.

In this experiment, sentiment analysis is performed using a Python package from the Natural Language Toolkit called VADER (Valence Aware Dictionary for sEntiment Reasoning) ^[1]. VADER is a “lexical approach to sentiment analysis that is sensitive to both polarity (positive/negative) and intensity (strength) of emotion” ^[2]. By “lexical”, we mean that VADER uses a dictionary of words mapped to emotions. To help the reader understand this approach a little more, I quote from DataMeetsMedia.com:

“Lexical approaches look at the sentiment category or score of each word in the sentence and decide what the sentiment category or score of the whole sentence is. The power of lexical approaches lies in the fact that we do not need to train a model using labeled data, since we have everything we need to assess the sentiment of sentences in the dictionary of emotions. The sentiment score of a text can be obtained by summing up the intensity of each word in the text. [Individual words are scored from -4 to +4, and] the sentiment score of a sentence is calculated by summing up the sentiment scores of each VADER-dictionary-listed word in the sentence. Normalization is applied to the total to map it to a value between -1 to 1. [Also note that] lexical features aren’t the only things in the sentence which affect the sentiment. There are other contextual elements, like punctuation, capitalization, and modifiers which also impart emotion. VADER sentiment analysis takes these into account by considering five simple heuristics.” ^[2]

Although this experiment uses only VADER sentiment scores for analysis, we also capture sentiment scores from another (Python) natural language processing package called TextBlob ^[3]. Statistical analysis, however, is performed using only VADER sentiment scores.

Experiment design

- **Step 1:** Identify users that fall into two groups: “fitness vocalizers” and “media consumers”. Each group is determined by the key words and hashtags that they currently use in their tweets. By “currently” – we are implying that we will stream live their tweets.
- **Step 2:** Stream tweets from step 1.

- **Step 3:** Clean streamed tweets from step 1 to eliminate bias.
- **Step 4:** For the first N users of each sample group, fetch (non-filtered) tweets.
- **Step 5:** Generate sentiment analysis for full tweet list from step 4.
- **Step 6:** Conduct statistical analysis of sentiment.

Experiment example:

Sample 1: Fitness	Sample 2: Media
<ol style="list-style-type: none"> 1. Create keywords and hashtags to define both samples. 2. Stream 5000 tweets (Stream API) for each sample. 3. Clean (filter) streamed tweets. 4. Fetch tweets for first 3000 filtered users. Twitter REST API returns a max of 20 tweets per user. Ideally, this step will result in a sample of at least 10,000 tweets. 5. Run sentiment analysis. 6. Conduct statistical analysis of sentiment. 	

API limitations

Twitter's free API has specific limits on the quantity and types of queries that can be performed. For the REST API, these limits are ^[4]:

- Only 20 (of each user's) most recent can be pulled for timeline requests.
- Full tweets are returned for up to 100 users per requests

As a result of these limitations, our sample size is much smaller than possible.

Stages 1 and 2: Defining samples and streaming them

To capture recent and relevant tweets and circumvent the rate limiting of Twitter's REST API, it is optimal to use Twitter's Stream API, which is a live connection to a feed of tweets. Twitter's users generate an average of 6000 tweets per second, so this is an abundant source of data for this experiment.

The two groups of Twitter users whose tweets we want track in this project are people that talk about fitness ("fitness vocalizers") and people that talk about media ("media vocalizers"). To categorize these individuals into samples, we will filter Twitter's Stream API by keywords and hashtags.

In order to avoid creating bias in our samples, it is crucial to select key words and hashtags that are as neutral (unemotional/unsentimental) as possible, while also relating to the target

group. For example, take the media group. If I were to filter for this group's tweets by including the hashtag of '#bestshow', then I risk introducing bias into my sample. The reason is because the word 'best' is very positive, so it skews our sample of tweets/authors towards positive sentiments. The same thing can be said of any positive or negative words. As a result, it is important that our keywords are as neutral as possible so as to prevent (unnecessary) bias in our sampling. The following are Python lists of keywords used to capture the tweets of these groups:

Keywords of “fitness vocalizers”:

- `keywords_m = ['watching show', 'watch season', 'netflix', 'movie', 'new season', 'watching tv', 'binge watching', 'newseries', 'new episode', 'prime video', 'dvr', 'atthemovies', 'film', 'horror', 'comedy', 'thriller', 'shortfilm', 'firstseason', 'secondseason', 'thirdseason', 'fourthseason', 'fifthseason', 'lastseason', '#watchingshow', '#watchseason', '#newseason', '#watchingtv', '#bingewatching', '#newepisode', '#primevideo', '#edgeofmyseat', '#nbc', '#abc', '#disney', '#cnbc', '#cbs', '#primetime', '#waitedsolong', '#comedycentral']`

Keywords of “media vocalizers”:

- `keywords_f = ['health fitness', 'fitness', 'legday', 'workoutwednesday', 'treadmill', 'pilates', 'yoga', 'gym time', 'deadlift', 'squats', 'FitnessFriday', 'gymlife', 'workouts', 'fitness training', 'postgym', 'armday', 'shoulderday', 'fitnessgoals', 'runner', 'workout', 'workout motivation', 'lift hard', 'lift weight', 'go running', 'crossfit', 'morning workout', 'muscle', 'six pack', 'lunges', 'cardio', 'elliptical', 'cycling', '#health', '#gymtime', '#fitnesstraining', '#workoutmotivation', '#lifthard', '#liftweight', '#gorunning', '#sweatforit', '#morningworkout', '#sixpack', '#triathlon']`

Stage 3: Cleaning streamed tweets

After capturing tweets returned from the Stream API, we now have two sample groups: tweets of people that discuss fitness, and tweets of people that discuss media. These sample tweet streams need to be cleaned. The reason for this is that we want to avoid authors of tweets

that tend to be “invalid”. Therefore, we filter out such users from our streamed samples so that when we pass them on to stage 4 (full data fetch), we’re dealing with “credible” authors of tweets.

Below are the criteria for filtering tweets. The goal is to remove tweets that match these criteria, and this is done using multiple layers of filtering logic, including regular expression matching. See file “*regex_readme.md*” for specifics on the regex filter.

Tweets are removed from streamed samples if they:

- Are advertisements or have commercial connotations.
- Discuss sensitive material (determined by Twitter).
- Contain profanity
- Are near duplicate tweets.
- Are retweets
- Are replies
- Or if the user is a bot
- Contain a hashtag count that is greater than the median hashtag count + 1 standard deviation.
- Contain other invalid language such:
 - Begin with a number. Example:
 - “7 great ways to save on protein powder!”
 - Begin with a hashtag. Example:
 - #GREAT#NEW#WORKOUT#CREATINE
 - Begin with quotes. These aren’t genuine user tweets. Example:
 - “Did you know that success is earned, not given? Try Precor’s new elliptical at your local Gold’s gym!”
 - Have .com/.org/.net. These are almost always advertisements
 - The words “free”, “buy”, “for men/for women”, “class”, “connect”, “discount”, “weight loss”, \$, +, -, @, and others. These words, phrases, symbols, questions and quotes almost always constitute an advertisement.
 - Periods separated by new lines. These are also usually advertisements.

Stage 4: Fetch non-filtered tweets for samples

After having collected and cleaned tweets of both sample groups, we now want to collect as much of their other (non-filtered) tweets as possible. Using the example above, we select the users of the first 3000 out of 5000 streamed tweets and pass them to the Twitter API to fetch their other tweets.

Stage 5: Run sentiment analysis

Upon retrieving the full list of tweets from step 4, the next step is to perform NLTK VADER sentiment analysis. We do this using Python. As mentioned previously, TextBlob sentiment analysis was also performed as a supplementary method but is less sophisticated and suitable compared to VADER, which is purpose-built for social media sentiment analysis.

Stage 6: Analyze sentiment

The final stage is to perform a statistical analysis of the sentiments generated in stage 5. This analysis is performed and described in detail in the file called '*Twitter_Sentiment_Analysis.pdf*'. A second file, entitled '*Twitter_Sentiment_Analysis_Exp_2.pdf*' is the same analysis for a larger sample (but start with the first).

Project Results

The result of the statistical analysis performed in stage 6 is that we should reject the null hypothesis. Indeed, it appears that people that talk about fitness are, at the population level, happier than people that talk about media.

Implications

The implications of this project are varied and numerous:

- Businesses that operate in the health (or fitness) industries would be keen to advertise or sell services to individuals who are not only fitness-minded, but also generally more positive people. These companies know that positive people want to sustain their attitude, lifestyle, and the longevity that comes along with it. Companies that could directly benefit by targeting "happier" people would include "health" organizations (clinics, gyms, wellness centers, hospitals) and insurance companies. The idea is to help people be happier (by being healthy) rather than to sell them on specific products (a new treadmill) or services (protein powders or new diets). I don't think it's an entirely new idea, but currently, the business focus by companies that are (even indirectly) related to health is more on the product/service rather than lifestyle changes that lead to a happier life. Gyms certainly

advertise to this degree, but even so, they're usually targeting weight loss goals or performance increases instead of fitness for the sake of happiness.

- Media companies, ironically, would also be interested in accessing a customer segment that is both healthier and happier. Not only is this a customer segment that offers considerable payoff financially, but media companies could also attune to the fact that fitness minded people might prefer different kinds of entertainment compared to non-fitness minded people. Perhaps healthier people like to watch cooking and traveling shows rather than reality tv or action movies. This is conjecture, but definitely worth exploring.
- Lastly, academic institutions could also benefit by studying fitness-minded individuals (who tend to be more positive as a population). The reason for studying this group directly is to understand if they have other traits or habits that help them function in society. I would venture to say that being fitness-minded is directly correlated to the instinct of self-preservation. And the degree of this instinct is positively correlated to how fulfilling life is. So the question is, what makes life fulfilling to people that are (as a whole) positive.

Technical Details of Project

What tools were used to complete this experiment?

- Stages 1 through 4 were performed using Python. A program was created for this experiment. To learn more about the Python program, start with the 'README.md' file in the 'Twitter_Sentiment_Analyzer' folder.
- Stage 5 was completed in R. To read about statistics analysis, see 'Twitter_Sentiment_Analysis.pdf' (or *.Rmd).

Sources

[1] Hutto, C.J. & Gilbert, E.E. (2014). VADER: A Parsimonious Rule-based Model for Sentiment Analysis of Social Media Text. Eighth International Conference on Weblogs and Social Media (ICWSM-14). Ann Arbor, MI, June 2014.

[2] VADER Sentiment Analysis, <http://datameetsmedia.com/vader-sentiment-analysis-explained/>

[3] TextBlob, <https://textblob.readthedocs.io/en/dev/quickstart.html>

[4] Tweepy Twitter API, <http://docs.tweepy.org/en/v3.6.0/api.html>