Social Media Mining Exploration: Top-100 Ranked Higher Education Institutions and Twitter Activity

JASON CRISMORE, Indiana University
ASHLEY MILLER, Indiana University
NATHANIEL MOODY, Indiana University

The increased scrutiny on higher education has created questions among the public regarding the quality, financial investment, and ultimately, value, of pursuing a degree from an institution. A declining population of college bound students along with increased competition between institutions as well as arising alternatives to the traditional college experience leaves the future of higher education in a state of flux. Potential students today are faced with a multitude of choices and resources to help inform that choice. One common cited resource is the U.S. News and World Report (USNWR) rankings list which is released annually. This resource highlights where public and private universities fall relative to each other as they take into account reputation, selectivity of students, and resources, among other measures. This one resource provides a partial picture into the competitive landscape and perception of institutions, but social media is also another avenue to explore how one 'talks' about a particular university. It also offers an opportunity to assess whether this is a way to predict whether a tweet from Twitter is from a highly ranked institution (such as 'Top-10') or from another institution that is not as highly ranked. We will seek to explore this question through the use of Twitter data to better understand if a model can be created and produce a high enough accuracy rating to predict whether a tweet is associated with a 'Top-10' institution, as defined by the USNWR rankings, compared to a 'Non Top-10' institution, looking at the top 100 colleges and universities from the 2017 published list.

Additional Key Words and Phrases: social media mining, twitter, rankings, higher education

ACM Reference format:
Jason Crismore, Ashley Miller, and Nathaniel Moody. 2017. Social Media Mining Exploration: Top-100 Ranked Higher Education Institutions and Twitter Activity. *Proc. ACM Hum.-Comput. Interact.* 1, 1, Article 1 (December 2017), 14 pages.
https://doi.org/1

1 INTRODUCTION

While a multitude of factors contribute to college selection among students, national and even international ranking lists can be used as a technique to shorten the initial consideration set [6]. Many list types and methodologies exist; however, the U.S. News and World Report (USNWR) rankings popularized in the 1980s started the process of looking beyond the research reputation of universities. Instead, this ranking system takes into consideration graduation and retention rates, faculty resources, alumni giving rates, as well as undergraduate academic reputation through a survey assessment that currently accounts for 22.5% of the ranking calculation [7]. Academic reputation of a university can be conveyed in a number of ways via traditional media methods as well as through social media. Today, one of out every five high school students utilize Twitter to learn more about a particular college [4]. Colleges and universities have now included social media as an integral part of their university's 'brand' strategy which calls into question: how much of a role does social media play into the rankings game [9]?

2573-0142/2017/12-ART1 $
https://doi.org/1

As of July 2017, there are now nearly “2.4 million fewer college bound students" which means increased competition among higher education institutions [1]. Social media has now been cited as a “phenomenon which can drastically impact a brand’s reputation and in some cases survival" [9]. Prior research reviews show that while social media can impact the performance of a university in regards to branding, affinity, and recruitment, little data is available to showcase the relationship, if any, between social media and national rankings.

Since timing of rankings are released annually, this project methodology could be used as a way to continually track performance over time in between ranking releases so institutions can better gauge where they may fall in the mix of other colleges and universities. Further, this analysis may provide guidance on their overall marketing and branding efforts. With these factors in mind, this project intends to explore whether the use of Twitter data can be used as a method to predict a college or university’s USNWR ranking, which will be used as a proxy for ‘brand strength’.

2 METHODOLOGY

2.1 Literature Review

Several other studies and sources have broached the topic of what relationship, if any, exists between social media as it relates to university branding and college rankings. The use of available Twitter data has been used in exploring the relationships among top-ranked universities to determine whether universities with common features are likely to associate with each other [10]. While this study showed a relationship between institution rankings to other ‘world-class’ universities, it also stated there was a relatively small impact. Further, this study only evaluated accounts of the top-ranked universities themselves and did not explore perception or sentiment among those in the ‘general population’. While the use of data points such as followers, hashtags, and mentions used in this study will likely provide guidance into our work, we hope to fill the void left by this study in examining similar data among a more representative audience. Existing research has also explored whether social media use can be tied to student outcomes and performance. One experimental study showed the positive impact of Twitter use on academic performance and engagement. Specifically, this study did not address the population of students included so it begs the question whether one is truly influenced by the other or whether these students were already ‘high-achievers’, with or without the use of Twitter [2].

Other work has showcased the ability to recreate the ranking methodology and adjust factors within to determine which variable(s) would need to be achieved in order to reach a specific ranking. An issue of ‘noise‘ is addressed frequently as a reason why rankings would fluctuate over time [5]. Evaluating sentiment data on Twitter may provide one opportunity to address this issue of ‘noise‘ in the data that is not addressed in great detail from other research. While Hazelkorn states that the influence of rankings on public consciousness is ‘immeasurable’, one could argue that the use of social media could be used in an effort to fill this gap and provide a consistent and measurable process to utilize over time [6].

Research by Rutter, Rope, and Lettice discuss a similar question to ours as they evaluated the relationship of higher education institution‘s use of social media and inclusion in the Russell Group study [9]. While this study found that “universities that interact more with their followers achieve better student recruitment performance than universities that fail to interact", this merely looked at the number of interactions versus the quality or sentiment of interactions to further validate the relationship [9]. Clark, Fine, and Scheuer sought to understand the relationship current students had with their university through not only social media data but also with primary quantitative research [3]. This approach seeks to explore the sentiment around the university and relationship quality which other work does not address in detail. However, sample size utilized was below the threshold of obtaining a five percent margin-of-error and convenience-based. Further, the authors only targeted those in marketing courses at the university which does not allow the ability to be representative of the student population and therefore results may be skewed. Also, their focus was only on one institution which limits

the ability to determine trends. Our approach intends to explore multiple institutions to better understand the sentiment of social media as it relates to rankings and what information can be gleaned from that relationship.

2.2 Initial Direction and Revision

As an initial direction, this research hypothesized that rankings similar to those compiled by USNWR could be achieved through a process of focused data-collection, NLP, and sentiment analysis. The initial goal was to replicate the rankings of those schools belonging to the Big-10 conference. However, the large number of colleges and universities being ranked by USNWR, their various categories/classes, and the inherent difficulty of determining which school is being referenced in the natural-language data we were drawing from, led to a minor redirection of our research.

Based on feedback from objective peer-reviewers, we revised our approach to instead include data from the top-100 universities according to the USNWR rankings instead of limiting it to those in the Big-10 set. From here, we sought to determine whether we could predict if a school is a 'top-10 ranked' university or 'not a top-10 ranked university'. This dataset was significantly larger, which allowed the opportunity for improved statistical analysis as compared to the original approach that would have produced a much smaller, and perhaps problematic, dataset. We were sure to emphasize that the intent is not to determine which universities students will or will not enroll in, but rather explore whether sentiment analysis among the top-10 ranked universities is different from those that are not ranked in the top-10.

2.3 Data Collection, Transformation, and Feature Selection

The dataset was drawn from Twitter status updates across the United States that pertain to those universities ranked by USNWR, specifically those in the top-100 rankings. This data is readily available, and easy to access. To acquire this data, the Twitter API was utilized. Status updates were collected over the course of several weeks, at different points during the day, and searched based on the names of universities from the USNWR list [8]. Once collected, this dataset formed a corpus of Twitter status updates that reference the ranked universities from a representative sample of Twitter users. This dataset controlled possible confounding variables such as time-of-day, day-of-the-week, or excessive attention to a daily current-event (i.e. a football game) by spreading out the collection process and gathering data from across the country. After collection, the dataset was cleaned and features selected from it, resulting in data ready for analysis. Features selected included date/time, text, and school-name searched. The text from each status, specifically, was evaluated and cleaned to ensure its readiness for natural language processing and analysis. The date/time and location features were preserved alongside the text for possible consideration, but did not prove to be pertinent to our results.

To acquire the desired dataset, we made use of the Tweepy package in Python to access the Twitter API. A cursor object was created and used over the span of several days to search for tweets based on a list of the top-100 schools from the USNWR rankings. The schools were initially searched for based on their full name and several abbreviations or colloquial ways of referring to them, but all searches carried out using anything other than the full school name returned an excessive amount of falsely-matched tweets. From this finding, the searches based on abbreviation and colloquial reference were eliminated. Even with this adjustment, one unavoidable limitation of our data became that several schools employ the same name (i.e. "Indiana University" in Pennsylvania and "Indiana University" in Indiana).

The raw dataset consisted of over 200,000 status updates. At the time of collection, this raw data was filtered by selecting only those features which pertained to the course of the project: Status text, Status location, and Status time. In addition to these features, the name of the school searched by and a binary variable to indicate if the school is one of the top-ten ranked colleges and universities (1) or not (0) was added. The data was then cleaned using regular-expression pattern matching. Tweets were flagged to indicate the status of the tweet as

either an original or a retweet. The Afinn Python package was used to create a polarity score for each cleaned text, and this was appended to each observation of the entire dataset.

2.4 Dataset Filtering and Masking

During the machine learning phase, secondary data filtering and cleaning was required. This filtering included a reduction of dataset size and masking of key data. The reason for dataset reduction was repeated out-of-memory errors due to the large size of the data after use of a Count Vectorizer from the Sklearn library. The reason for the data masking was to prevent the machine learning algorithms from associating the school name, city, or state to either the positive or the negative category. Data to support this decision is presented in the discussion section.

The hardware used for machine learning was an eight core 4.00 GHz AMD FX-8370 with 16.0 GB of RAM installed. The system also utilizes an Nvidia GeForce 1080 GTX Ti video card configured to utilize the platform for additional processing capability. During the running of machine learning scripts, there was no indication that the Sklearn library was utilizing the video card. Instead, the effective memory limit of the machine was measured as 14.4 GB. Beyond this point, the machine would destabilize and throw a memory error in Python. To resolve this issue, we limited the dataset to 100,000 records for processing machine learning algorithms.

2.5 Hypothesis

The causal relationship proposed is that higher ranked institutions are more highly prized by students, and therefore, attract more social media attention. Sometimes, this attention is in the form of tweet rates and other times it is shown in the polarity of the tweet. The figure belows shows polarity scores for each Tweet text, which were then grouped them by school, and summed. Overall school polarity scores varied greatly. Since we were operating off the general notion that Twitter-chatter polarity is somewhat predictive of a school's ranking in USNWR rankings, this shows that attempts to develop a model proved to be problematic.

2.6 Machine Learning Library and Model

The machine learning library used was Sklearn. The model used was a logistic regression. This model was one of the first models used, and it scored well enough to warrant continued investigation throughout the various iterations of machine learning.

A Count Vectorizer was used to break tweets into their individual words and fed into the machine learning algorithm. In all cases, this data overwhelmed the other data fields and became the primary factor in predictive model success.

Figure 1 shows the number of grouped Tweets collected for select schools. Twitter chatter referencing some schools hit the maximum for our collection process, while some schools barely generated any references at all. This discrepancy had to have an impact on the polarity scores aggregated and likely also played a role in making our predictive-model process difficult, as shown in figure 2.

3 DISCUSSION OF RESULTS

Iteration 1 of the machine learning phase analyzed all data available. This model scored 98.3%, but analysis of the good results quickly found that the machine learning algorithm was simply taking the school name and comparing it to the top ten flag. From here, data masking was performed to remove the college names. The alternative to masking was to remove these names. However, there must be some value in the data if a person mentions the college by name. Therefore, each keyword in the college name was replaced with the placeholder "uname" to indicate that the college name was replaced. This drove the model accuracy down below 65%.

New analysis of the data showed that a variety of factors were affecting the model. These factors included city, state, and the name of a prestigious school within the university. Masking of all of these terms was required.

These became ucity, ustate, and uschool to denote the different masking attempts. This quickly failed in many specific cases. For instance, Louisville became ucityville because the word Louis was replaced with ucity. Nearly every state school experienced some sort of conflict with this naming scheme and therefore it was difficult to differentiate between mention of a city in a tweet vs. mention of a school with a city name. It was decided to leave all replacements with a single placeholder called "uname."

Table 1 shows the words with the most weight from Iteration 1, while Table 2 shows the words with the least weight from Iteration 1. These words are replete with names of schools. Names of majors also appear, but no attempt was made to mask majors.

Iteration 2 contains all of the data masking. Its accuracy score is 95.9% after all masking is complete. As previously noted, this accuracy fluctuated from 65% with minimal masking up to the 95.9% score with full masking. Table 3 shows the words with the most weight, while Table 4 shows the words with the least weight. At this point, several differences in the data become obvious.

Separation by major begins to appear in the results. For instance, the term "divinity" is not a common major, but in the data appears to be associated with the top 10 schools.

Other majors might indicate student ambition or drive for success. The term "interdisciplinary" means the student is not committed to a single major, so perhaps they are a little less committed to academics. Hidden below the top 20 is the term "nursing" which might be contraindicative to attendance at medical school.

Social issues, such as "suppression", or fair weather terms, such as "tropical" might also indicate that the student's primary focus is not fully committed to academics.

The term "beinecke" is related to college scholarships and appears to be just as selective as the top 10 schools. The term "uk" might indicate the country abbreviation, but it could also indicate "University of Kentucky" and therefore, might be reviewed for the masking process. The word "kunshan" is probably another school naming error that has slipped through masking. In this case, the suspicion is that it refers to "Duke Kunshan University." This causes several problems in analysis because Duke University itself is a top-10 school while Duke Kunshan University is not. It also means that the data mining script was not 100% accurate in associating tweets to their proper schools.

A final group of keywords is likely famous professors getting mentioned at a prestigious school. This includes several terms that are common last names, such as Wong, Farragut, and Glover in the higher weighted words, or Coach Addazio from Temple University in the lower weighted words. Again, this difference in last name appearances may show a focus on academics vs. a focus on social factors such as athletics.

4 IDEAS FOR FUTURE RESEARCH

Future research in this area should further explore correct data masking. This list of items to mask would include University Name, City, State, Zip Code, School with the University, Scholarship Programs, Coaches Names, Mascot Names, and Famous Professor Names. If the principal of academic tweets vs. social tweets holds true, then tweets mentioning coaches would be grouped into the social category and tweets mentioning a particular school or scholarship program would be identify the top-10. There should also be some sort of discernible set of words that could be classified as academic or social. This also might be augmented to check the user of complete sentences vs. the use of vernacular or abbreviations.

5 TEAM MEMBER CONTRIBUTIONS

The team members for this project provided equal support in a variety of ways by making the following contributions:

- Jason Crismore: Jason's contribution to this project was in the areas of machine learning and natural language processing. This included writing code, secondary data scrubbing, and multiple attempts at finding

a good model for the data. Jason was also responsible for reference amalgamation near the beginning of the project.

- Ashley Miller: Ashley worked on developing and refining the question the team sought to explore. Ashley also contributed to this project by providing the literature collection and review of relevant research articles and topics. Ashley also compiled the final paper submission through the use of ShareLateX.
- Nathaniel Moody: Nathaniel's contribution to this project included some up-front organization of efforts, and assistance in composing the project proposal, dataset check, and peer feedback. Nathaniel also took responsibility for creating the scripts used to collect and clean the data, as well as the actual execution of these scripts over several weeks to create the final dataset. He also contributed to writing the sections of this final report that pertained to data collection and transformation.

6 CONCLUSIONS

This paper provided a clear framework for mining data from Twitter, preparing the data for analysis, and analyzing the data with machine learning. Problems with the data during the machine learning process were recognized and corrected by using data masking as a technique to get to an actual result. Iteration 1 involved no data masking and created an artificially high accuracy score. Iteration 2 involved heavy data masking and does not readily show that the outcome was affected by factors that would lead to incorrect results. Instead, our interpretation of the final model from Iteration 2 shows a clear differentiation in detecting academic related ideas vs. detecting tweets that are not focused on academics. In many cases, we were able to identify logical matching between terms and their relation to academic or social ideas. This was the true differentiation in tweets from the top 10 schools vs. tweets from the next 90 institutions.

ACKNOWLEDGMENTS

The authors would like to thank Vincent Malic, Ashley Dainas, and fellow students in the fall 2017 section of the Social Media Mining Course for their thoughtful feedback and guidance during this project. The authors would also like to thank the Association for Computing Machinery for providing a template for submission of this paper.

REFERENCES

[1] Isaac Carey and Jon Marcus. 2017. There are 2.4 million fewer college students than there were five years ago. (2017). http://hechingerreport.org/2-4-million-fewer-college-students-five-years-ago/

[2] III Charles H.F. Davis, Regina Deil-Amen, Cecilia Rios-Aguilar, and Manuel Sacramento Gonzalez Canche. 2012. Social Media in Higher Education: A Literature Review and Research Directions. *The Center for the Study of Higher Education at The University of Arizona* (2012).

[3] Melissa Clark, Monica B. Fine, and Cara-Lynn Scheuer. 2017. Relationship Quality in Higher Education Marketing: The Role of Social Media Engagement. *Journal of Marketing for Higher Education* 27, 1 (2017), 40–58.

[4] Stephanie Geyer. 2017. Ruffalo Noel Levitz E-Expectations 2017: How Online Expectations Change for TodayâĂŹs Prospective Students and Parents. (2017). http://www.act.org/content/dam/act/unsecured/documents/epc2017/Session-T4.5-E-Expectations-2017_v2.pdf

[5] Shari L. Gnolek, Vincenzo T. Falciano, and Ralph W. Kuncl. 2014. Modeling Change and Variation in U.S. News & World Report College Rankings: What would it really take to be in the Top 20? *Research in Higher Education* 55, 8 (01 Dec 2014), 761–779. https://doi.org/10.1007/s11162-014-9336-9

[6] Ellen Hazelkorn. 2014. Rankings and the Global Reputation Race. *New Directions for Higher Education* 2014, 168 (2014), 13–26. https://doi.org/10.1002/he.20110

[7] Robert Morse, Eric Brooks, and Matt Mason. 2017. How U.S. News Calculated the 2018 Best Colleges Rankings. (2017). https://www.usnews.com/education/best-colleges/articles/how-us-news-calculated-the-rankings

[8] U.S. News and World Report. 2017. 2018 Best National Universities (2017). (2017). https://www.usnews.com/best-colleges/rankings/national-universities

[9] Richard Rutter, Stuart Roper, and Fiona Lettice. 2016. Social media interaction, the university brand and recruitment performance. 69 (02 2016).

[10] Robin Shields. 2016. Following the leader? Network models of âĂIJworld-classâĂİ universities on Twitter. *Higher Education* 71 (2016), 253–268. Issue 2. https://doi.org/10.1007/s10734-015-9900-z

List of Figures

List of Tables

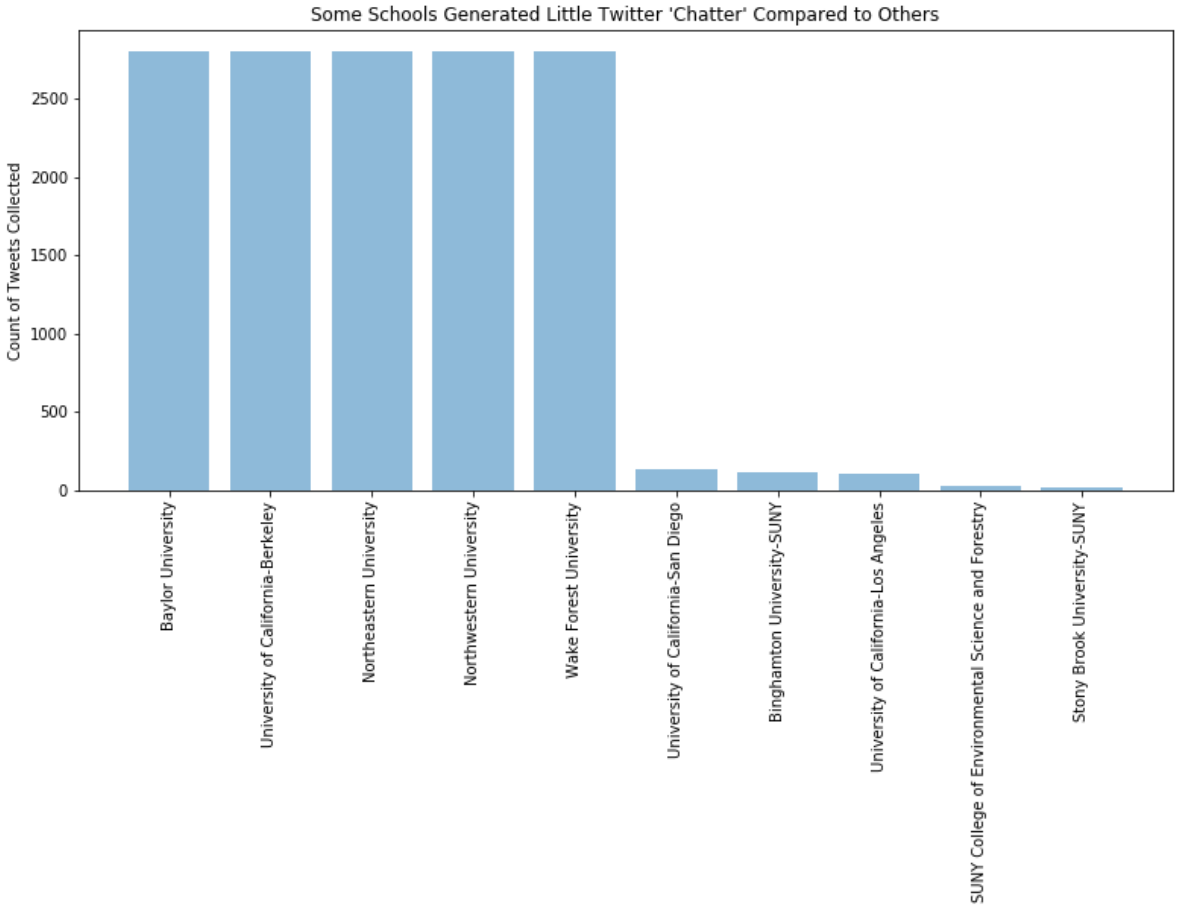

Fig. 1. Head and Tail Tweet Counts

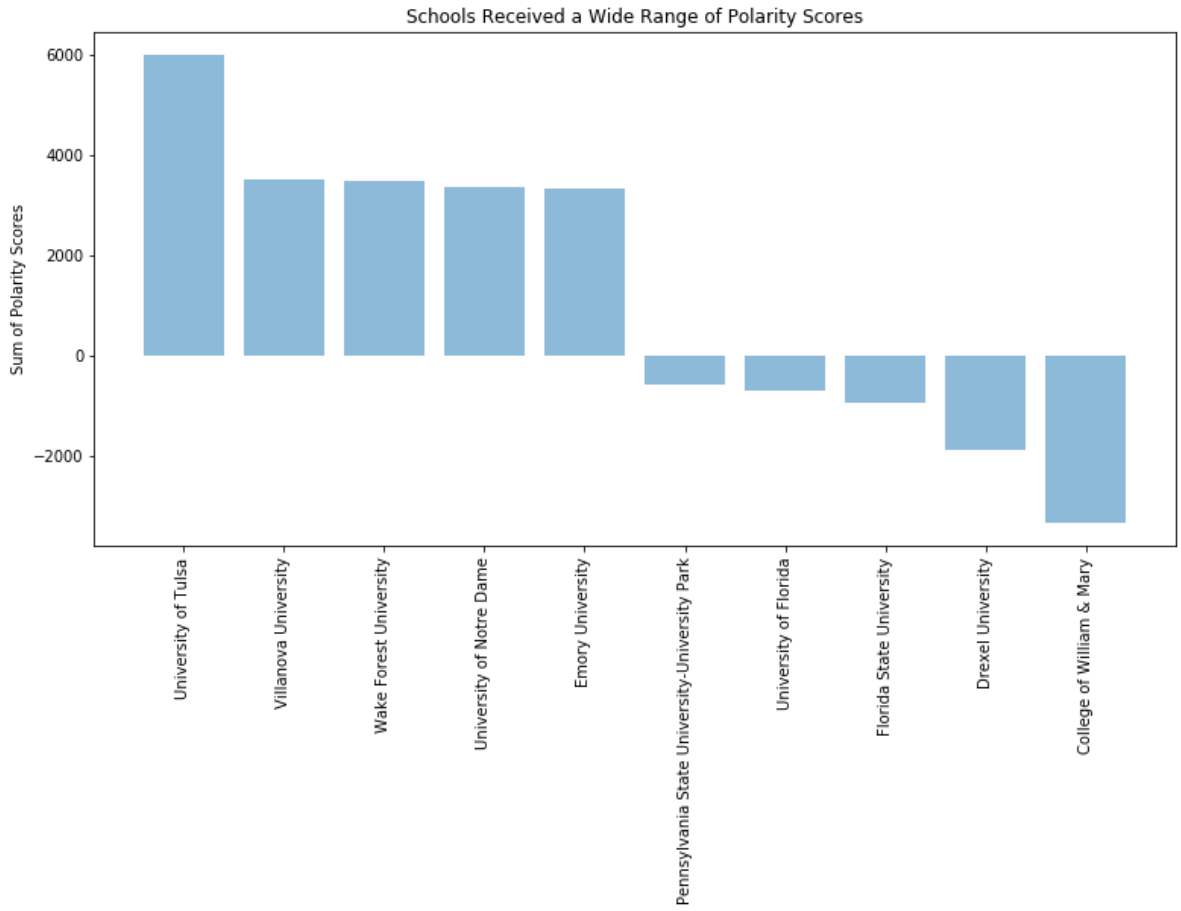

Fig. 2. Best and Worst Polarity Sums

Table 1. Words From Iteration 1 with the Most Weight

Word	Score
duke	17.1386406806
princeton	16.730480173
yale	16.5050114019
pennsylvania	16.2529736832
assistantassociate	14.071619079
exhibition	14.0523699713
chicago	13.4531261345
finance	13.0345810053
andor	12.8678905089
mit	12.3989804067
brian	12.0124035179
anxiety	11.9494559857
shut	11.4186137689
harvard	11.3412002415
idea	11.3198902568
vegas	10.6080296476
patch	10.1126489384
processor	10.0394928399
ct	9.81089676854
contemporary	9.78298630386

Table 2. Words From Iteration 1 with the Least Weight

Word	**Score**
nohwestern	-21.3758367298
intern	-19.0605458353
stevens	-18.9693950718
baylor	-18.4582230278
emory	-18.3479255321
virginia	-17.4813525788
villanova	-16.6814893272
georgia	-16.4718232594
pepperdine	-15.7852744893
wilson	-15.5571762121
amherst	-15.1472869037
fsu	-15.0070656646
podcast	-14.9945212549
profit	-14.8666109304
pittsburgh	-14.8477108087
cloudy	-14.8146980131
note	-14.6324973545
urbanachampaign	-14.4623740049
rochester	-13.9552771453
fordham	-13.7885977659

Table 3. Words From Iteration 2 with the Most Weight

Word	**Score**
colleg	15.7661350656
kunshan	14.659622067
beinecke	13.6055178672
farragut	13.4297606746
dislike	12.9285159906
cous	12.4788589705
gracie	12.4741243308
initiatives	12.0476284264
grows	12.0192913765
glover	11.9020885944
consume	11.8527599367
karsh	11.7605130024
dogs	11.6206693768
birds	11.3584376184
wong	11.2931006387
divinity	11.2144207195
heat	11.1754581711
uk	11.1638010435
difference	10.8816815703
users	10.8201683887

Table 4. Words From Iteration 2 with the Least Weight

Word	**Score**
interdisciplinary	-13.0578241609
tay	-12.7519433988
essence	-12.1263481082
kolin	-12.0414455722
addazio	-11.9232788422
mines	-11.859643081
chairelectrical	-11.8451479981
positive	-11.7344299963
entire	-11.5797263002
transfer	-11.3917010079
suppression	-11.3210149373
registration	-11.2882012878
counselor	-11.1698090493
ssm	-11.0572013615
cprc	-10.8519784768
tropical	-10.7994059315
undergrad	-10.795591441
cybersecurity	-10.6465121942
provide	-10.555736303
tweets	-10.4221199551

Data Acquisition Script - MK2

December 6, 2017

1 Final Project - Data Acquisition Script

1.0.1 Acquiring data in fulfillment of the final project for ILS Z-639, "Social Media Mining"

The following script will be run three times each day, over the course of seven sequential days, gathering 1400 Twitter Statuses each time and appending them to a dataset. To accomplish this, the script will execute the following tasks:

1. Access the Twitter API.
2. Create data storage and execute search for Twitter Statuses.
    - Execute a search on the school's name.
    - Use the cursor object to account for rate-limiting, and accumulate Twitter Statuses in a temporary dictionary.
    - Append the Twitter Statuses gathered this way to the accumulator dataframe.
3. Write/append the accumulator dataframe to file.
4. Check the state of the dataset, and create a new backup copy each time the script is executed.

Once the data-set is accumulated, it can be accessed by other scripts for cleaning/prep and analysis.

1.1 1. Access the Twitter API:

```
In [1]: import tweepy as tp
        import pandas as pd
        import json
        import datetime as dt
        from datetime import timedelta
```

```
In [2]: with open("Twitter_Keys.json", "r") as file:
            keys = json.load(file)

        API_KEY = keys["API_KEY"]
        API_SECRET = keys["API_SECRET"]
```

```
In [3]: auth = tp.AppAuthHandler(API_KEY, API_SECRET)
        api = tp.API(auth, wait_on_rate_limit=True, wait_on_rate_limit_notify=True)
```

1.2 2. Data Storage Creation and Search/Accumulation:

```
In [4]: topTen = [
            'Princeton University',
            'Harvard University',
            'University of Chicago',
            'Yale University',
            'Columbia University',
            'Massachusetts Institute of Technology',
            'Stanford University',
            'University of Pennsylvania',
            'Duke University',
            'California Institute of Technology'
        ]

        topHundred = [
            'Princeton University',
            'Harvard University',
            'University of Chicago',
            'Yale University',
            'Columbia University',
            'Massachusetts Institute of Technology',
            'Stanford University',
            'University of Pennsylvania',
            'Duke University',
            'California Institute of Technology',
            'Dartmouth College',
            'Johns Hopkins University',
            'Northwestern University',
            'Brown University',
            'Cornell University',
            'Rice University',
            'Vanderbilt University',
            'University of Notre Dame',
            'Washington University in St. Louis',
            'Georgetown University',
            'Emory University',
            'University of California-Berkeley',
            'University of California-Los Angeles',
            'University of Southern California',
            'Carnegie Mellon University',
            'University of Virginia',
            'Wake Forest University',
            'University of Michigan-Ann Arbor',
            'Tufts University',
            'New York University',
            'University of North Carolina-Chapel Hill',
            'Boston College',
```

```
'College of William & Mary',
'Brandeis University',
'Georgia Institute of Technology',
'University of Rochester',
'Boston University',
'Case Western Reserve University',
'University of California-Santa Barbara',
'Northeastern University',
'Tulane University',
'Rensselaer Polytechnic Institute',
'University of California-Irvine',
'University of California-San Diego',
'University of Florida',
'Lehigh University',
'Pepperdine University',
'University of California-Davis',
'University of Miami',
'University of Wisconsin-Madison',
'Villanova University',
'Pennsylvania State University-University Park',
'University of Illinois-Urbana-Champaign',
'Ohio State University-Columbus',
'University of Georgia'
'George Washington University',
'Purdue University-West Lafayette',
'University of Connecticut',
'University of Texas-Austin',
'University of Washington',
'Brigham Young University-Provo',
'Fordham University',
'Southern Methodist University',
'Syracuse University',
'University of Maryland-College Park',
'Worcester Polytechnic Institute',
'Clemson University',
'University of Pittsburgh',
'American University'
'Rutgers University-New Brunswick',
'Stevens Institute of Technology',
'Texas A&M University-College Station',
'University of Minnesota-Twin Cities',
'Virginia Tech',
'Baylor University',
'Colorado School of Mines',
'University of Massachusetts-Amherst',
'Miami University-Oxford',
'Texas Christian University',
'University of Iowa',
```

```
            'Clark University',
            'Florida State University',
            'Michigan State University',
            'North Carolina State University-Raleigh',
            'University of California-Santa Cruz',
            'University of Delaware',
            'Binghamton University-SUNY',
            'University of Denver',
            'University of Tulsa',
            'Indiana University-Bloomington',
            'Marquette University',
            'University of Colorado-Boulder',
            'University of San Diego',
            'Drexel University',
            'Saint Louis University',
            'Yeshiva University',
            'Rochester Institute of Technology',
            'Stony Brook University-SUNY',
            'SUNY College of Environmental Science and Forestry',
            'University at Buffalo-SUNY'
        ]

        df = pd.DataFrame(columns=["school", "text", "location", "time", "topten"])
```

In [5]:
```
df.shape
```

Out[5]:
```
(0, 5)
```

In [6]:
```
#startTime = dt.datetime.now().time()
for school in topHundred:

    if school in topTen:
        topten = 1
    else:
        topten = 0

    c = tp.Cursor(api.search, q=school, lang="en")

    for status in c.items(200):
        tempDict = {"school":school, "text": status.text, "location":status.geo, "time
        df = df.append(tempDict, ignore_index=True)
#endTime = dt.datetime.now().time()
```

```
Rate limit reached. Sleeping for: 749
Rate limit reached. Sleeping for: 741
```

In [7]:
```
#timeToRun = startTime - endTime
#print("It took the script", (startTime-endTime), "to run.")
```

```
In [8]: df.shape

Out[8]: (14843, 5)

In [9]: df.tail(2)

Out[9]:                            school  \
        14841  University at Buffalo-SUNY
        14842  University at Buffalo-SUNY

                                                                text location  \
        14841  RT @BenedictLabs: Ready to rock!!!! #acspib #n...     None
        14842  Ready to rock!!!! #acspib #ncw University at B...     None

                              time  topten
        14841 2017-10-24 22:57:51     0.0
        14842 2017-10-24 22:53:02     0.0
```

1.3 3. Write/Append Data to a File:

```
In [10]: import pickle as pkl
         import os

In [11]: # Setup the dataframe for the first time this file is created.
         projectData = pd.DataFrame(columns=["school", "text", "location", "time", "topten"])

         # Check if the file exists. If so, load in the current state of the data.
         if os.path.exists("ProjectData.pkl"):
             with open("ProjectData.pkl", "rb") as fileData:
                 projectData = pkl.load(fileData)
             fileData.close()

         # Append the df created in the cells above to either
         # (1) the empty projectData dataframe, or (2) the current state of the data as it has
         projectData = projectData.append(df)
         with open("ProjectData.pkl", "wb") as fileData:
             pkl.dump(projectData, fileData)
         fileData.close()
```

1.3.1 Check the Curent State of the Dataset, and Create a Backup Copy:

```
In [12]: # Check if the dataset is intact.
         with open("ProjectData.pkl", "rb") as fileData:
             accumulated_dataset = pkl.load(fileData)
         fileData.close()

         print(accumulated_dataset.shape)
         print(accumulated_dataset.head(5))
```

```
        # Overwrite the backup dataset with the full current dataset.
        with open("BackupDataset.pkl", "wb") as backup:
            pkl.dump(accumulated_dataset, backup)
        backup.close()
```

```
(204841, 5)
                school                                                text  \
0  Princeton University  RT @ElizabethOhar16: Ray Lewis just joined Ner...
1  Princeton University  RT @wonderfulboyy: Nerium had a global patent ...
2  Princeton University  RT @Princeton: Karina Aguilar Guerrero 20 tal...
3  Princeton University  RT @imfrancesnegron: Bankruptcy and Citizenshi...
4  Princeton University  D. Vance Smith - Princeton University | RateMy...

  location                 time  topten
0     None 2017-10-14 19:41:06     1.0
1     None 2017-10-14 19:40:56     1.0
2     None 2017-10-14 19:02:27     1.0
3     None 2017-10-14 18:56:21     1.0
4     None 2017-10-14 18:55:55     1.0
```

1.3.2 And... obsessively check the backup dataset to make sure it is alright:

```
In [13]: with open("BackupDataset.pkl", "rb") as checkBackup:
             bup = pkl.load(checkBackup)
         checkBackup.close()
         print(bup.shape)
         print(bup.head(5))
```

```
(204841, 5)
                school                                                text  \
0  Princeton University  RT @ElizabethOhar16: Ray Lewis just joined Ner...
1  Princeton University  RT @wonderfulboyy: Nerium had a global patent ...
2  Princeton University  RT @Princeton: Karina Aguilar Guerrero 20 tal...
3  Princeton University  RT @imfrancesnegron: Bankruptcy and Citizenshi...
4  Princeton University  D. Vance Smith - Princeton University | RateMy...

  location                 time  topten
0     None 2017-10-14 19:41:06     1.0
1     None 2017-10-14 19:40:56     1.0
2     None 2017-10-14 19:02:27     1.0
3     None 2017-10-14 18:56:21     1.0
4     None 2017-10-14 18:55:55     1.0
```

Data Cleaning Script - MK2

December 6, 2017

1 Final Project - Data Cleaning Script

1.0.1 Cleaning data in fulfillment of the final project for ILS Z-639, "Social Media Mining"

The following script will take in a corpus of Twitter Status updates and clean them, in preparation for sentiment analysis. To do this, this script will accomplish the following tasks: 1. Read in the data from the pickle file where it is stored, as a pandas dataframe. 2. Iterate through the dataframe and clean the features. 3. Create a polarity score for each Status. 4. Write out the newly cleaned and scored dataset as a pickle file and a csv file.

```
In [21]: import pandas as pd
         import re
         import pickle as pkl
         from afinn import Afinn
         import winsound
         import time

In [22]: start = time.time()
```

1.1 1. Read in data from pickle file as a pandas dataframe:

```
In [23]: with open("ProjectData.pkl", "rb") as file:
             oldData = pkl.load(file)

In [24]: oldData.index = range(0, len(oldData)) # Need to re-index the dataset because data-co
```

1.2 2. Iterate through the dataframe and clean the features:

1. Fix the topten classification to be an integer instead of a float.
2. Classify the data as a retweet or not.
3. Clean the english of the text.

In [25]:
```
topten = []
for row in oldData.topten:
    if row == 1.0:
        topten.append(1)
    else:
        topten.append(0)

oldData["topten"] = topten
```

In [26]:
```
retweet = []
rt = re.compile("^RT")
for row in oldData.text:
    if re.search(rt, row):
        retweet.append(1)
    else:
        retweet.append(0)

oldData["retweet"] = retweet
```

In [27]:
```
for row in range(len(oldData)):
    text = oldData.loc[row, "text"]
    text = text.lower()

    text = re.sub("[\.\?\!\:\+\;\,\/\-\$\>\|]", "", text) # Remove punctuation/charac
    text = re.sub(u"\u2551", "", text) # Remove whatever this is
    text = re.sub(u"\u00BB", "", text) # and whatever this is
    text = re.sub("&amp", "", text) # Remove ampersands
    text = re.sub("rt", "", text) # Remove retweet tag
    text = re.sub("@(\w*)\s", "", text) # Remove retweet/tweet-at names
    text = re.sub("@(\w*)$", "", text) # Remove final retweet/tweet-at names
    text = re.sub("#(\w*)\s", "", text) # Remove hashtags
    text = re.sub("#(\w*)", "", text) # Remove final hastag
    text = re.sub(u"\u2026", "", text) # Remove ellipses (you have to use the unicode
    text = re.sub("http(\w*)\s", "", text) # Remove hyperlinks
    text = re.sub("http(\w*)$", "", text) # Remove end-of-line hyperlinks
    text = re.sub("\d", "", text) # Remove time/date numerals
    text = re.sub("\s(\s*)\s", " ", text) # Strip extra whitespace
    text = re.sub("^\s", "", text) # Strip beginning whitespace

    oldData.text[row] = text
```

```
C:\Users\natha\Anaconda3\lib\site-packages\ipykernel\__main__.py:21: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/indexing.html
```

1.3 3. Prep data for sentiment analysis:

1. Use afinn to tokenize each tweet and give it a polarity score.

```
In [28]: afinn = Afinn()

In [29]: scores = []

         for row in oldData.text:
             polarity = afinn.score(row)
             scores.append(polarity)

         oldData["polarity"] = scores
```

1.4 4. Write out the cleaned and scored data to a new file:

1. Write data as binary in a pickle file.
2. Write a duplicate set of the data out as a csv file, for "Progress Report" submission.

```
In [30]: with open("WrangledData.pkl", "wb") as file:
             pkl.dump(oldData, file)
         file.close()

In [31]: with open("WrangledData.pkl", "rb") as file:
             testData = pkl.load(file)
         file.close()

         print(testData.head())
         print(testData.tail())
```

```
                school                                               text  \
0  Princeton University  ray lewis just joined nerium because our amazi...
1  Princeton University  nerium had a global patent and rights for eht ...
2  Princeton University  karina aguilar guerrero  talks with on being ...
3  Princeton University  bankruptcy and citizenship the caribbean crisi...
4  Princeton University  d vance smith princeton university ratemyteach...

  location                time  topten  retweet  polarity
0     None 2017-10-14 19:41:06       1        1       4.0
1     None 2017-10-14 19:40:56       1        1       0.0
2     None 2017-10-14 19:02:27       1        1       0.0
3     None 2017-10-14 18:56:21       1        1      -6.0
4     None 2017-10-14 18:55:55       1        0       0.0
                           school  \
204836  University at Buffalo-SUNY
204837  University at Buffalo-SUNY
204838  University at Buffalo-SUNY
```

```
204839  University at Buffalo-SUNY
204840  University at Buffalo-SUNY

                                                    text location  \
204836  ready to rock university at buffalo suny close...     None
204837  hello we did an alumni survey for my universit...     None
204838  ready to rock university at buffalo suny close...     None
204839  ready to rock university at buffalo suny close...     None
204840  ready to rock university at buffalo suny close...     None

                     time  topten  retweet  polarity
204836 2017-10-25 01:24:44       0        1       0.0
204837 2017-10-25 00:10:15       0        0       0.0
204838 2017-10-24 23:31:50       0        1       0.0
204839 2017-10-24 22:57:51       0        1       0.0
204840 2017-10-24 22:53:02       0        0       0.0
```

In [32]:
```
# This just alerts me when the script is done cranking through the whole dataset.
frequency = 2500  # Set Frequency To 2500 Hertz
duration = 500  # Set Duration To 1000 ms == 1 second
winsound.Beep(frequency, duration)

end = time.time()
wall_clock_execution_time = (end - start)

print("Time to execute script:", wall_clock_execution_time)
```

```
Time to execute script: 6302.431977033615
```