Yelp Elitism

Predicting Annual Yelp Elite User Selections

Technical Report

Michael Peng, Rachel Lee, Pratibha Rathore

Applied Natural Language Processing, Fall 2015

I. ABSTRACT

In this report, we propose a model that attempts to predict whether a Yelp user will be awarded Elite status in the following year, given the user's set of reviews for a year. Elite status is awarded to active, helpful contributors in the Yelp community.[11] Elite Yelpers have the opportunity to be invited to private community events in their local region, many of which provide food free of charge. Elite members' accounts are also designated as such, and their reviews are featured more prominently on the site. We attempt to achieve this by creating a multidimensional model that integrates natural language processing techniques, user metadata,
review count, number of votes their reviews received, social network structure, etc. This is a binary classification problem, so we use various supervised machine learning algorithms.

II. INTRODUCTION

Yelp is a social-networking platform used by people to discover various businesses, especially restaurants, and glean an understanding of the businesses via reviews. The platform encourages users to contribute and establish their presence on Yelp. Our team identified a potential in examining 'Elite' user aspect of Yelp, which works to this end of identifying "star" contributors. Every year, Yelp hand-picks an exclusive community of users who are very active and are ostensibly among its best contributors for that year. We were curious with regard to the features of 'Elite' users, and attempted to create both qualitative and quantitative measures. A model that can predict whether user can become an Elite of not would be very valuable not only for Yelp, but also for users interested in becoming 'Elite'.

III. PROJECT GOAL

From a set of reviews posted by a given user in a given year, predict whether or not this user will be awarded Elite status in the following year, given the reviews for a year.

IV. DATASET DESCRIPTION

The Academic dataset consists of 1.6M reviews written by 366K users for 61K businesses, provided in JSON format and is accessible online for the Yelp Dataset Challenge. Specifically, each data subset (reviews, users, businesses, tips, etc) is comprised of a list of the respective JSON objects. This presented us with some difficulty initially in parsing the data, as there is no access to the objects directly from the list--instead, a Python dict was used to map relevant data. More discussion follows in the Data Processing section.

Table 1 below contains some pertinent statistics regarding the dataset.

Table 1

Statistics	**Elite Users**	**Non-Elite Users**
Review Count (per user)	245	16
Review Length (# words)	98.8	64.9
Vocabulary (size across all reviewers of type)	125,137	95,428
Votes (# on user's reviews)	1336	32
Friends	55	3
Compliments	8	0.0000
Fans	16	0.0000

V. DATA PROCESSING

To begin with, we examine the percentage of elite users and non-elite users. Unsurprisingly, over 95% of users are non-elite, and most of whom have not posted very many reviews. Upon a preliminary examination of the user objects, we ascertain that only .083% of elite users have fewer than 20 reviews. Were we to take the entire dataset into account, without filtering out those users with very few reviews, our data would likely be skewed; review quality would matter little for users who simply did not post often at all. We therefore only consider annual sets of reviews for users who have posted 20+ reviews in their lifetime (an attribute accessible from each user's JSON object).

We also find, upon examining the reviews of the dataset, that reviews in many businesses located in Europe receive reviews that are not in English. In order to have a meaningful NLP-based conceptualization of our model, and achieve a standardized feature set, we err on the side of caution by only examining reviews for businesses in cities located in the US. These cities are as follows: Pittsburgh, PA; Charlotte, NC; Urbana-Champaign, IL; Phoenix, AZ; Las Vegas, NV; and Madison, WI. This way, we put a stronger guarantee on text standardization and lack of foreign-language reviews.

Elite status is provided as an attribute of user objects--it exists as a list of years in which a user was considered elite. Status determination is also decided at the end of each calendar year. We therefore may also conclude that reviews written in 2015 should be excluded from the set we attempt to examine, as they have not been labeled yet.

In order to format the raw data to present to the learning machine phase, we structure the data into lists of review objects posted by a given user in a given year. In order to do this, we use two Python dicts: one mapping each relevant user_id to a list of the years in which we can consider the set of reviews contributory toward elite status, and another mapping a tuple (user_id, year) to a list of the reviews that correspond to the user and year indicated by the key.

At this point, we find that it may be more interesting to examine the performance of a model based on the reviews posted in 2014--the most recent set of reviews, with Yelp labeling users 'Elite' for the 2015 calendar year. To this end, we decided to split the data into two sets: one containing all reviews prior to 2014, and one containing all and only those reviews posted in 2014. The former we may further divide into our training and validation sets later, and the latter comprises our test set. The proportion of the training/development set to test set separated in this way appears to be 77.46%, which is reasonably close to the classic 80/20 data split between test set and train/dev set.

VI. DATA EXPLORATION

In deciding on our approach for building a predictive model, we formulated some questions to help guide our feature selection and model-building, based on other research papers we read. These questions are enumerated below as subsections, with a brief discussion of findings.

How does average number of votes per review for users change over time?

This analysis was not pertinent, as the current Yelp Academic Dataset provides no method of ascertaining votes on a review over time. The dataset only provides a snapshot of the data.

Are Elite users first to review a new business?
It has been found that elite users do not review new business proportionally more often than non-elite users, compared to the baseline rate of business reviews.[5]

Does a user's metadata indicate his/her status?
According to several groups[2,3,5], metadata was often a very strong indicator for a user's elite status. Metadata features include number of friends, review count, votes (on the user's reviews), compliments, and fans. We therefore hypothesized that review metadata could be similarly good indicators, as a user's aggregate vote attribute is derived from his or her individual reviews.

Does the social network structure suggest whether a user is elite user?
According to Lee and Massung[3], using the social network structure produced an algorithm with decent user identification accuracy (.796). However, in refining our goal for this task to be more unique, we determined that a social network structure analysis for individual years would not be possible given the snapshot nature of the data.

How does text in elite reviews differ from text in normal reviews?
It appears that there is much more potential here to differentiate elite and normal reviewers, with language-related features and NLP ideas. As follows, we will examine the performance of orthographic features, tokens, vocabulary size, etc.

VII. FEATURE SELECTION

1. User Metadata

For our predictive task, we begin by looking at user objects, which have the following format:

```
{
    'type': 'user',
    'user_id': (encrypted user id),
    'name': (first name),
    'review_count': (review count),
    'average_stars': (floating point average, like 4.31),
    'votes': {(vote type): (count)},
    'friends': [(friend user_ids)],
    'elite': [(years_elite)],
    'yelping_since': (date, formatted like '2012-03'),
    'compliments': {
        (compliment_type): (num_compliments_of_this_type),
        ...
    },
    'fans': (num_fans)
}
```

It is clear from Table 1 that Elite users have substantially more friends, votes, reviews, and fans. This provides a strong indication that these features will be very important in in determining if a user can be Elite or not. However, because the data we examine is time-sensitive, and related to the review objects for each given user in a year, the user object metadata is discarded here. There is no relation between a user's current metadata and the specific years we are examining, and there is otherwise no way to step into metadata during specific years.

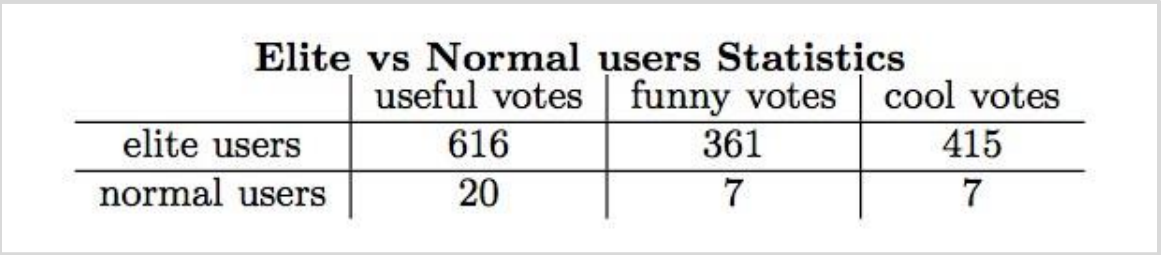
2. Review Metadata

Each review is a JSON object as shown below:

```
{
    'type': 'review',
    'business_id': (encrypted business id),
    'user_id': (encrypted user id),
    'stars': (star rating, rounded to half-stars),
    'text': (review text),
    'date': (date, formatted like '2012-03-14'),
    'votes': {(vote type): (count)},
}
```

No correlation exists between rating given and elite status, which follows, as determining status based on the degree to which a user is favorable toward businesses would be unethical in both directions. Review text, as mentioned, is worth more detailed examination described in following sections. Votes were an important metadata feature as found in prior research[2], so we decide to try useful, funny, and cool votes in our features. Table 2 below shows the distribution of normal vs elite users' votes received. Date is used as described to segment reviews, but has no bearing on the model itself.

Table 2



Elite vs Normal users Statistics

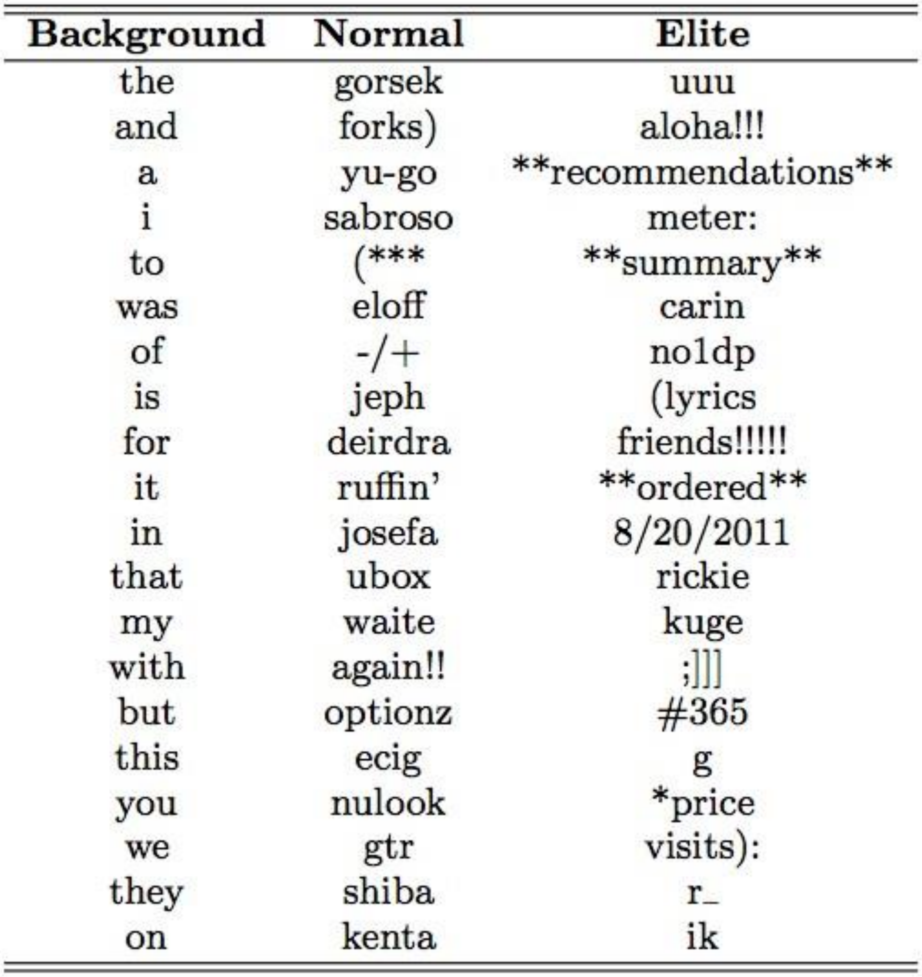
	useful votes	funny votes	cool votes
elite users	616	361	415
normal users	20	7	7

3. Language Model

To begin using more NLP-related techniques to analyze the dataset, we began with a unigram-based language model to analyze how does the review tokens utilized by elite users in reviews differ from those used by non-elite users. Table 3 below compares background frequency (stopwords), normal, and elite users' most common tokens.

The table is constructed using a frequency distribution of all words in the corpus, with the top twenty most frequent words extracted and displayed. Below is a table that shows the top twenty tokens for all the three user category. The intention behind utilizing this unigram language model is to find out words that are indicative of 'Elite' users.

Table 3 : Top 20 tokens for each category



Background	Normal	Elite
the	gorsek	uuu
and	forks)	aloha!!!
a	yu-go	**recommendations**
i	sabroso	meter:
to	(***	**summary**
was	eloff	carin
of	-/+	no1dp
is	jeph	(lyrics
for	deirdra	friends!!!!!
it	ruffin'	**ordered**
in	josefa	8/20/2011
that	ubox	rickie
my	waite	kuge
with	again!!	;]]]
but	optionz	#365
this	ecig	g
you	nulook	*price
we	gtr	visits):
they	shiba	r_
on	kenta	ik

It appears plainly from Table 3 that individual tokens may not be sufficiently indicative of elite versus non-elite status--many of the common tokens in the diagram are written by users with over-representative usage of the words. For instance, the "aloha!!!" token is written by a specific user before each review. Other tokens (e.g. -/+) are bugs in the dataset. It seems that, among both Normal and Elite users, somewhat meaningless tokens may appear very often, and be disproportionately representative. We decide from this model that the Naive Bayes bag-of-words classic spam filtering model may not be particularly useful, and that individual tokens may not be useful for our classification purpose, as a result of the variation of formality and speech (and, therefore, tokens) among Yelp users as a whole.

However, one significant finding gleaned from the above table is that the set of Elite users very often tend to segment reviews into different sections. This paragraph-based segmentation stretches horizontally across all reviews (rather than specific tokens being used consistently by specific users), and in each segment discusses different aspect of the reviews business, e.g. price, food he/she ordered, recommendations, summary, etc. This implies that reviews written by elite users have more structure and style.

4. Textual Features

The brunt of our feature analysis arose in determining which language-based features of the dataset we might use in order to classify users' review years. From orthographic features to paragraph segmentation of the review text, we delved into the review text itself here to understand attempt to understand differences between writing styles of elite users compared to normal users. Below are the features that we considered:

Average review length: This is average number of tokens across all the reviews for a user. We consider total characters as well as total words.

Paragraph rate: This feature was developed from the unigram language model described earlier in the report in which we inferred that elite users have more structure to their reviews. These structures likely include dedicated sections for the experience, the pros, the cons, and a recommendation. Paragraph rate takes in to consideration this aspect of review segmentation, which is simply a count of the rate of double newline characters per review per user, which is the predominant method for paragraphing among users (newline plus another space between paragraphs).

All capital letters : This is a count of number of capital letters per review. We tried out this orthographic feature as it is likely that a high rate might indicate spam or emotional/reactionary reviews with less value to other users.

Bad punctuation: In addition, we also used this feature to detect less sincere reviews or spam reviews. For example: a new sentence starting with lowercase letter or the sentence ending without a period.

Average review sentiment: This is a very complicated feature to understand and implement correctly, hence we did a lot of research to know if sentiment analysis has been done on Yelp user reviews and we discovered a paper called 'Opinion mining and sentiment analysis' by Pang Bo and Lillian Lee[6] in which they came up with a overall sentiment valence score for each user review. If the score is less than zero then the review is considered overall negative and vice a versa. Though this was a very neat approach to incorporating value of sentiments, we did not incorporate it into our predictive model, as it was computationally intensive while yielding marginal benefits, with a predictivity that amounted roughly to the same as guessing a baseline "non-elite". However, we may definitely explore this feature for our future work: there may well be a trend of Elite users expressing more negative sentiment on the whole, as they may be picky in their analysis; at the same time, Elite users may be more likely to espouse the virtues of a store or restaurant they particularly.

5. Readability

Readability scores are simple, linear-regression-based formulas that provide a rough idea of a text's readability. Having taken note of suggestions with regard to automated essay scoring algorithms and similar readability measures, we also tried to use readability features in our analysis. These formulas use surface characteristics, such as word, syllable, and sentence counts, largely ignoring syntactic or semantic detail. We decided to explore how various readability scores, used as features, will affect our model. Sadly, we have not found any conclusive results as of yet. We hypothesize that readability can also be confounded by the tuning of the respective scores' weights. We found a great readability script developed by user Andreas van Cranenburgh on GitHub[11], and attempted to integrate it into the model. Unfortunately, it appears that the weighting of readability features is very difficult to get right, given 30 widely varying features.

In addition to readability, we hypothesized that vocabulary size and variability, as well as or parts-of-speech composition, differentiated Elite user reviews. While vocabulary size was, on average, higher for Elite users (Table 1), it failed to contribute significantly to accuracy (see results for more discussion). Likewise, parts-of-speech composition (i.e. pronoun counts, interrogative counts) did not appear to be significant, and did not make it into the final model.

6. Final Selected Features

```
feature_dict = {
    1: "total reviews",
    2: "total characters",
    3: "total paragraphs",
    4: "total cool votes",
    5: "total funny votes",
    6: "total useful votes",
    7: "total sentences",
    8: "total words",
    9: "total size of vocabulary (unique words)",
    10: "chars per review",
    11: "paragraphs per review",
    12: "cool votes per review",
    13: "funny votes per review",
    14: "useful votes per review",
    15: "sentences per review",
    16: "words per review",
    17: "size of vocabulary per review"
}
```

VIII. BRIEF DISCUSSION ON FEATURES

This may be a good place to take a brief aside on our featurization before we begin discussion on our learning model. We suspect that many of our intuitively chosen features failed to improve our model’s accuracy as a result of a few factors. First, this may be a matter of adjusting weights and feature combinations in order to reach superior performance on our learning models (as discussed previously). Second, one of the few concrete qualifications for Elite status is the user’s age. Only users 21 or older may be considered, but we did not have access to this data, as all users are anonymized in the dataset, and users have the option not to display age on their default profiles as well.

However, while there is large variability in review content and format within the non-Elite users, Elite users must have some quantifiable, common aspect in their reviews; perhaps an algorithmic filtering followed by hand-picking. Should we adjust our training data such that a larger percentage is Elite, we may achieve better results; the current results may be obfuscated by the relative abundance of non-elite users’ representation in the featurized data. Another alternative to structuring our data could be to stratify reviews by stars, perhaps eliminating all 1 or 2 star reviews, then train on each stratum separately. We found a discussion with Professor Marti Hearst regarding such data stratification, and going forward, this may be a better approach to analyzing the data--sans some significant confounding variables.

IX. LEARNING MODEL SELECTION

Naive Bayes: Often used as a classic bag-of-words model . This is usually an incredibly efficient model considering its simplicity, but did not provide satisfactory results for our goals, as we found that the tokenized words in each review were often non-sensical and very obfuscating[2]. We did try the model using the final feature set of 17, without spectacular results.

```
T Positive: 2296, F Positive: 5496
F Negative: 3317, T Negative: 12784

The following metrics are on a scale of 0 to 1:
Model accuracy: 0.6311471979240781
Model precision: 0.2946611909650924
Model recall: 0.40905041867094244
Model F1 Score: 0.34255874673629244
Wall time: 2.33 s
```

SVM: With our large set of features and data points, this model was not very efficient for our process of testing features and moving forward towards our goals. Each featurizing round would take hours, and the SVM would also accordingly take hours given any significant number of features, and the machine generally requires 100+ features for accurate classification[8]. We also found that it was not a particularly strong model in general.

Logistic Regression: Unfortunately, this model yielded results close to guessing non-Elite for all users, which exists as the baseline. We found that logistic regression was an especially poor model to use, as when *running the model on the data on which it was trained*, it still yielded roughly guessing-level accuracy. This suggests that the model as a whole did not work well for our dataset and the specific features we selected. However, it was much quicker than other methods, so we initially attempted logistic regression. The following result occurs when fitting to the training data:

```
T Positive: 1206, F Positive: 456
F Negative: 21260, T Negative: 72653

The following metrics are on a scale of 0 to 1:
Model accuracy: 0.7727857703374313
Model precision: 0.7256317689530686
Model recall: 0.05368111813406926
Model F1 Score: 0.09996684350132626
Wall time: 4.63 s
```

Random Forests: in the end, we settled on random forests for its combination of accuracy and (comparatively) quick training times. We also knew the importance of not overfitting to our training set, and random forests is a relatively good algorithm for that purpose[8]. Using a non-normalized feature set and hyperparameters of n_estimators=40 and max_depth=5 for our scikit-learn model, we received results as displayed in the following results section.

X. RESULTS

Shown below is the confusion matrix for our final predictive model which is Random Forest on non-normalized data.

	Elite User	**Non-Elite User**
Classified Elite	T Positive: 1,143	F Positive: 574
Classified Non-Elite	F Negative: 4,680	T Negative: 28,373

The key model output measures are:

Model accuracy: 0.8489
Model precision: 0.6657
Model recall: 0.1963
Model F_1 Score: 0.3032

The accuracy of 85 % for the prediction model is good but not great as the baseline was 76%, that is 76% accuracy can be by always guessing ‘non-elite’. The precision is 66.5% which is relatively better: out of every 3 guesses that a user was Elite, our model is correct on 2 of those guesses. However the recall is 19%, which is on the lower side: out of 5 elite users, the model only identified 1. The F_1 score was “decent” by some metrics, but is difficult to generalize (e.g. in some situations, precision or recall may matter much more, so the mixing effect of using this score would be undesirable) to all datasets.

```
Top 5 features in order from most to least important:
Rank: 1   |  Feature: total paragraphs       |   Importance score: 0.233239
Rank: 2   |  Feature: total characters       |   Importance score: 0.136758
Rank: 3   |  Feature: paragraphs per review  |   Importance score: 0.127070
Rank: 4   |  Feature: total cool votes       |   Importance score: 0.120385
Rank: 5   |  Feature: chars per review       |   Importance score: 0.093581
```

Importance score is essentially the out-of-bag error of the feature (how much worse the model performed without the feature). We can see here that paragraphing and raw output were the most significant features, with votes also making an appearance. This is in line with what we hypothesized regarding paragraphing: Elite users often utilize segmenting and structuring in their reviews to provide an organized conveyance of their thoughts. Surprisingly, the total word count feature was essentially useless, ranking very low in importance score. This supports our earlier hypothesis that after a certain threshold, review count itself may not matter as much. While the model did not perform as well as we wanted, ideally, we can see that on the test set of 2014 reviews, there was decent overall accuracy at roughly 85%.

XI. FUTURE WORK

We have touched on some of the ideas going forward earlier in the report, but as a post-mortem to this project, we have identified several of the most promising ideas going forward. First, we may attempt to use the readability features again, as they are the most thorough application of NLP, incorporating a wide variety of language-related metrics. By tuning the weights, we may have more success. In addition, we found that a very large barrier to model development for this project was the large size of data. Perhaps a random or stratified sample of the data would serve better to construct basic features with which to train the model until a satisfactory set of features were determined. While we did not find a large number of NLP-relevant features, being able to train on a subset of the data and then classify on a larger set may be more helpful.

Additionally, there exists other publically available, high-performance NLP libraries and modules, such as spaCy. Another attempt at the project could take advantage of highly performant tools available in such toolkits in order to process an adequate number of data points, with a greater abundance of NLP-related features, which may yield more interesting results.

Given our results and available research, metadata or network structures may be the primary determiners of Elite status, and it is possible that complex NLP features may ultimately be relatively fruitless as determiners (especially pertinent given the likelihood of becoming an Elite member given merely a baseline participation rate, demonstrated competency in reviews, and a minimum membership time). However, our analysis appears still to be ultimately inconclusive, and NLP features are not utterly ruled out as possible factors in Elitehood.

XIII. APPENDIX

1. "Yelp." Dataset Challenge. Yelp, n.d. Web. 5 Dec. 2015.
2. Costa, Gian, Arturo Aguilar, and Eric Jiang. "Evaluating The Yelp Elite Squad." (2015): n. pag. University of California, San Diego. Web.
3. Lee, Cheng Han, and Sean Massung. "Multidimensional Characterization of Expert Users in the Yelp Review Network ∗." (n.d.): n. pag. Web.
4. "What Is Yelp's Elite Squad?" What Is Yelp's Elite Squad? Yelp, n.d. Web. 11 Dec. 2015.
5. Crane, Heh, Johnny Winston. "An Analysis of the 'Elite' Users on Yelp.com." Stanford CS 224. Web. 10 Dec. 2015.
6. Pang, Bo, and Lillian Lee. "Opinion Mining and Sentiment Analysis." Foundations and Trends in Information Retrieval (n.d.): n. pag. 2008. Web.
7. http://snap.stanford.edu/class/cs224w-2014/projects2014/cs224w-32-final.pdf
8. Rohrer, Brandon. "Machine Learning Algorithm Cheat Sheet for Microsoft Azure Machine Learning Studio." Microsoft Azure. Microsoft, 13 Oct. 2015. Web.
9. Baharudin, Baharum, Lam Hong Lee, and Khairullah Khan. "A Review of Machine Learning Algorithms for Text-Documents Classification." Journal of Advances in Information Technology JAIT 1.1 (2010): n. pag. Web.
10. Brownlee, Jason. "Classification Accuracy Is Not Enough: More Performance Measures You Can Use - Machine Learning Mastery." Machine Learning Mastery. N.p., 21 Mar. 2014. Web. 10 Dec. 2015.
11. https://github.com/andreasvc/readability/
12. http://moin.delph-in.net/WeSearch/DocumentParsing
13. Yelp Elite FAQ. http://www.yelp.com/elite. Web. 5 Dec. 2015.