Han Liu

Professor Biggs

Intro to Computational Linguistics

14 December 2016

Computational Linguistics Final Paper:

Picking out good housing conditions from Airbnb

ABSTRACT

Computational linguistics contributes a lot to the analysis of the world nowadays, considering the huge amount of texts either in publications, or online. Through linguistic analysis, the big data people gathered could be interpreted and understood as a great number of useful information. One good way to study these data is to extract texts such as reviews of a product or service on the website of a company. In this paper, an analysis of texts on Airbnb.com would reveal several luminous and interesting information through statistical linguistic analysis.

1 INTRODUCTION

The project is inspired by a statistical linguistic analysis on food reviews from Yelp [1]. The idea of an analysis on Airbnb is derived with personal interest. Airbnb is an online service that connects people looking for accommodation with people who can rent or share their houses or apartments. Users can leave ratings and comments about their experience on the page of the listing that accommodates them. These project downloads these reviews, classifies them with features and

analyzes the housing conditions presented or implied in the texts. Although Airbnb's service covers many countries and areas, this project discusses the extracted housing conditions of listings on the west coast and east coast of United States, considering the mainstream of users and the most common language used through the service.

2 DATA SET

Data of listings and reviews are gathered through an Airbnb application program interface (API)[1]. Scripts are written in python; data are stored and maintained in a local MongoDB database. All data collected are from the Airbnb API server in the month of November 2016. Data of several neighborhood are considered; a few are chosen based on listing counts. Listings in the following area are researched and documented (data used in this project are blackened):

- New York City: **Manhattan, 19216 listings; Brooklyn 16697 listings;** Harlem, 3757 listings; Williamsburg, 3805 listings.
- Los Angeles: **Hollywood, 3965 listings; Venice, 2025 listings;** Orange County, 4520 listings; Mid-Wilshire, 3000 listings.
- San Francisco: **Oakland, 2205 listings;** Berkeley, 1464 listings; Mission District, 1108 listings.
- Miami: Miami Beach, 5149 listings; South B, 2855 listings.
- Hawaii: Oahu, 4339 listings; Maui, 3996 listings; Honolulu, 2660 listings.

Data in three cities, Oakland, Los Angeles, and New York, are selected and gathered considering popularity, listing counts, and review counts. Data are classified as five groups according to the neighborhoods they belong: Hollywood, Venice, Oakland, Manhattan, Brooklyn.

[1] For the documentation and usage of the API, please see APPENDIX.

Listings are first requested to the server. Data returned are sorted by unknown algorithm on the server. Due to API limitation, at most three hundred listings could be returned per area. Reviews are requested by existing listings. Every review of the existing listings in the database can be requested and downloaded. The number of listings and reviews are counted. The number of listings with more than fifty reviews and the correlating reviews are also counted.

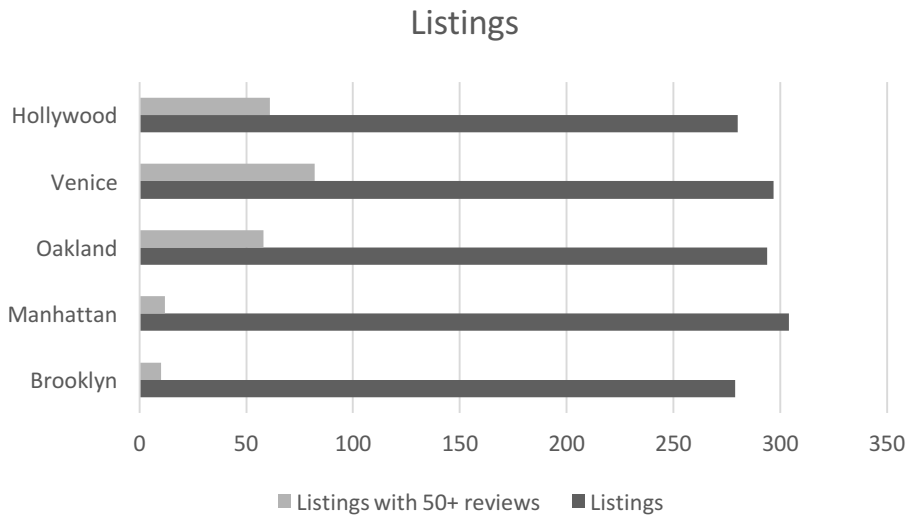

Figure 1: Listings vs. listings with more than fifty reviews per area

As presented in figure 1, about three hundred listings are counted but most listings have less than fifty reviews. Information of these listings are stored in the database. The exact number of listings is as follow:

- Hollywood: 280 listings, 61 with more than fifty reviews.
- Venice: 297 listings, 82 with more than fifty reviews.
- Oakland: 294 listings, 58 with more than fifty reviews.
- Manhattan: 304 listings, 12 with more than fifty reviews.
- Brooklyn: 279 listings, 10 with more than fifty reviews.

These are the mainly studied listings in the project. It is easy to see the listings in west coast are generally more reviewed than those in the east coast. Accordingly, reviews of these listings are also counted and presented in Figure 2:

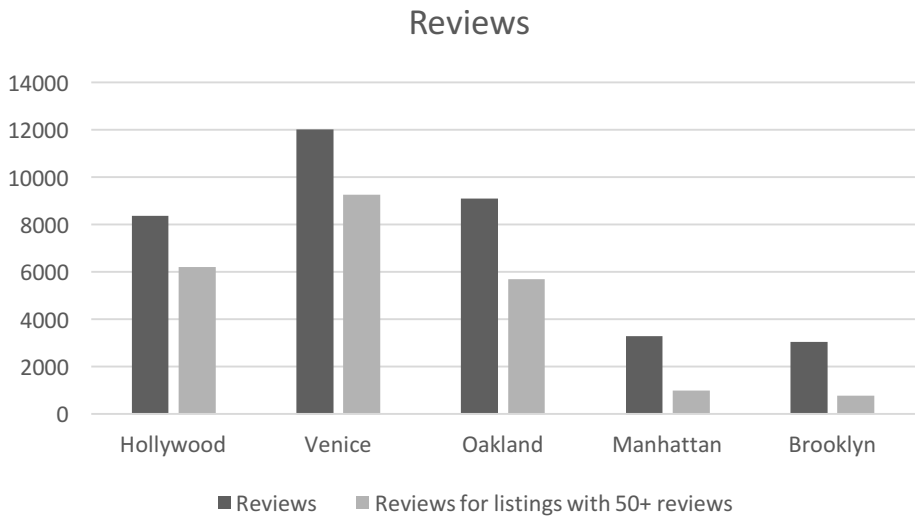

Figure 2: Reviews vs. reviews for listings with more than fifty reviews per area

As presented in Figure 2, about ten thousand reviews each are gathered for listings in Hollywood, Venice, and Oakland. About three thousand reviews each are gathered for listings in Manhattan and Brooklyn. Generally, listings in west coast have more reviews than those in the east coast. The exact number of reviews is as follows:

- Hollywood: 8379 reviews, 6179 of which are for listings with more than fifty reviews.
- Venice: 12031 reviews, 9271 of which are for listings with more than fifty reviews.
- Oakland:9096 reviews, 5700 of which are for listings with more than fifty reviews.
- Manhattan: 3286 reviews, 1001 of which are for listings with more than fifty reviews.
- Brooklyn: 3038 reviews, 782 of which are for listings with more than fifty reviews.

These are the mainly studied reviews in the project. Reviews are maintained in the database, and parsed into raw texts for storage as well. Language or reviews are set to be English when the reviews are requested. However, there are a few reviews that are labeled as in English but actually in another language. It is difficult to avoid this problem when requesting data. Luckily, reviews in foreign language are filtered through sentiment analysis.

Sentiment analysis are implemented on the general reviews. Reviews are classified as three groups judged by their polarity: positive reviews, negative reviews, and neutral reviews. The sentiment distribution is presented in Figure 3:

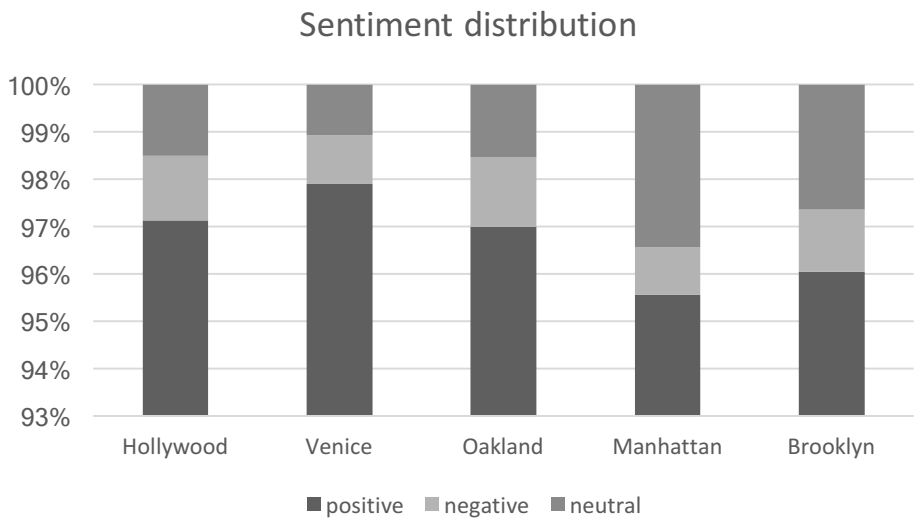

Figure 3: Sentiment distribution of reviews per area.

As presented in figure 3, roughly nighty seven percent of the reviews are positive. The exact number of sentiment distribution is as follow:

- Hollywood: 8139 positive reviews, 114 negative reviews, 126 neutral reviews.
- Venice: 11778 positive reviews, 124 negative reviews, 129 neutral reviews.
- Oakland: 8823 positive reviews, 134 negative reviews, 139 neutral reviews.
- Manhattan: 3140 positive reviews, 33 negative reviews, 113 neutral reviews.

- Brooklyn: 2918 positive reviews, 40 negative reviews, 80 neutral reviews.

In this project, only positive reviews and negative reviews are studied, with a focus on positive reviews, due to the significant amount of positive reviews in all reviews.

3 METHOD

In this section, the technical methods of the implementation will be discussed.

Python is the main programming language used in this project. MongoDB is used for maintaining database. PyMongo is used for connecting the database with operations in python scripts[2]. In the project directory, "init_database.py" is the main python script that manages the sending of requests to the server and initializes the database with data returned. "get_listings_by_location.py" and "get_reviews_by_listings.py" are two main scripts that send the actual request and parse the result into dictionaries to save into the database. These three files need to be run with stable internet connection and local database connection. "get_listings_by_neighborhood.py" and "get_reviews_by_neighborhood.py" connect to local database and classify the corresponding listings and reviews into different neighborhoods.

```
def reviews_to_text(reviews_collection, db):
    reviews_cursor = db['reviews_' + reviews_collection].find()
    reviews = [review for review in reviews_cursor]
    comments = [review["comments"] for review in reviews]
    comments = [comment for comment in comments
            if 'This is an automated posting.' not in comment]
    comments = [re.sub('\s+', ' ', comment) for comment in comments]

    with open('data/reviews_%s' % reviews_collection, 'w') as f:
        for comment in comments:
            f.write('%s\n' % comment)

```

Figure 4: Function that filters the text to be written

[2] Python version 3.5.2; MongoDB version 3.4.0; PyMongo version 3.4.0.

"reviews_to_text.py" connects to the database, read the collections of reviews, and write out the raw texts of those reviews into .txt files in the "data/" directory. As presented in Figure 4, automated reviews are excluded from the texts. Regular expression is also used to eliminate redundant end line signs that may vary in different format or operating system. These three files need to be run with stable connection to local database.

The main script for text analysis is "raw_analysis.py". It could be run without need of connections if the raw texts of data are already preprocessed and prepared in the "data/" directory. If the texts prepared do not include sentiment analysis, it will first generate the classified positive and negative collections of reviews. Then, the analysis will begin.

```
def analysis(reviews_collection_text):
    with open('data/reviews_%s' % reviews_collection_text, 'r') as f:
        raw_data = f.read()
    with open('data/reviews_%s' % reviews_collection_text, 'r') as f:
        comments = f.readlines()
    data = raw_data.replace('\n', ' ')
    data_lower = data.lower()
    tokens_with_punc = word_tokenize(data_lower)
    tokens = RegexpTokenizer(r'\w+').tokenize(data_lower)
    print("--- Most frequent tokens ---\n",
        FreqDist(tokens_with_punc).most_common(15))
    print("--- Tokens without punctuation ---\n",
        FreqDist(tokens).most_common(15))
    stop = set(stopwords.words('english'))
    words = [word for word in tokens if word not in stop]
    print("--- Most frequent words ---\n", FreqDist(words).most_common(15))
    tagged = pos_tag(words)
    nouns = [word for word, pos in tagged if (pos == 'NN')]
    print("--- Most frequent nouns ---\n", FreqDist(nouns).most_common(15))
    adjts = [word for word, pos in tagged if (pos == 'JJ')]
    print("--- Most frequent adjective ---\n", FreqDist(adjts).most_common(15))
    tokns = [RegexpTokenizer(r'\w+').tokenize(comment) for comment in comments]
    lxdst = [lexical_density(token) for token in tokns if len(token) > 0]
    avgld = sum(lxdst) / len(comments)
    print("--- Average lexical density ---\n", avgld)

def lexical_density(tokens):
    return len(set(tokens)) / len(tokens)
```

Figure 5: Function that does the main analysis

The NLTK package is used in the analysis[3]. All modules needed are already included at the beginning of the file. As presented in Figure 5, the analysis follows such procedures:

- Read raw files entirely into "raw_data".
- Replace end line signs so that texts are presented as sentences in a row.
- Change all characters in lower case.
- Tokenize lowered data with NLTK tokenizer.
- Tokenize lowered data with Regex tokenizer for removing punctuation.
- Print out most frequent tokens of the two lists of tokens
- Remove stop words with NLTK default stop words dictionary.
- Print out most frequent words of the text.
- POS-tagging the tokens with NLTK POS-tagger.
- Print out most frequent nouns and adjectives of the text.
- Calculate and print out the lexical density of the text.

The main function will loop the collections of reviews in every area in such order: Hollywood, Venice, Oakland, Manhattan, Brooklyn. Every collection would go through above analysis with result outputted in the console[4].

The sentiment analysis in this project is implemented with VADER sentiment intensity analyzer from the sentiment module of the NLTK package [2]. The "polarity_scores(*text*)" function calculates the polarity scores of a given text. A dictionary of positive scores, negative scores, neutral scores and compound scores are calculated based on its own algorithm. First three scores are on scale from 0.0 to 1.0, while the compound score is scaled from -1.0 to 1.0. Figure 6

[3] Natural Language Toolkit (NLTK) is a leading package for natural language processing in Python. The version used in this project is 3.2.1.
[4] For the sample output, please see APPENDIX.

below shows an example of the performance of Vader sentiment intensity analyzer on one of the review in Hollywood from the database.

```
>>> sia = vader.SentimentIntensityAnalyzer()
>>> sia.polarity_scores("Very nice host! Highly recommended! I hope I can be the
re again!")
{'neu': 0.431, 'compound': 0.8497, 'neg': 0.0, 'pos': 0.569}
```

Figure 6: Vader sentiment intensity analyzer

Based on the polarity scores, the comments are classified as three groups of comments: comments with compound score greater than 0 as positive comments, comments with compound score less than 0 as negative comments, others as neutral comments.

```
41  def sentiment_analyzer(reviews_collection_text):
42      sia = vader.SentimentIntensityAnalyzer()
43      with open('data/reviews_%s' % reviews_collection_text, 'r') as f:
44          comments = f.readlines()
45      pos_comments = [comment for comment in comments
46                      if sia.polarity_scores(comment)['compound'] > 0]
47      neg_comments = [comment for comment in comments
48                      if sia.polarity_scores(comment)['compound'] < 0]
49      neu_comments = [comment for comment in comments if comment not in
50                      pos_comments and comment not in neg_comments]
51      with open('data/reviews_%s_pos' % reviews_collection_text, 'w') as f:
52          for pos_comment in pos_comments:
53              f.write('%s' % pos_comment)
54      with open('data/reviews_%s_neg' % reviews_collection_text, 'w') as f:
55          for neg_comment in neg_comments:
56              f.write('%s' % neg_comment)
57      with open('data/reviews_%s_neu' % reviews_collection_text, 'w') as f:
58          for neu_comment in neu_comments:
59              f.write('%s' % neu_comment)
```

Figure 7: Function that classifies comments based on sentiment

Above in Figure 7, the function that implements the Vader sentiment intensity analyzer and classifies the comments are shown. Analysis are done on the reviews after sentiment analysis.

4 RESULTS

The following groups of graphs show the most frequent nouns and adjectives mentioned in the positive reviews of the five areas. Unique nouns and adjectives for each area are also picked out.

Hollywood: unique noun, night; unique adjective, N/A.

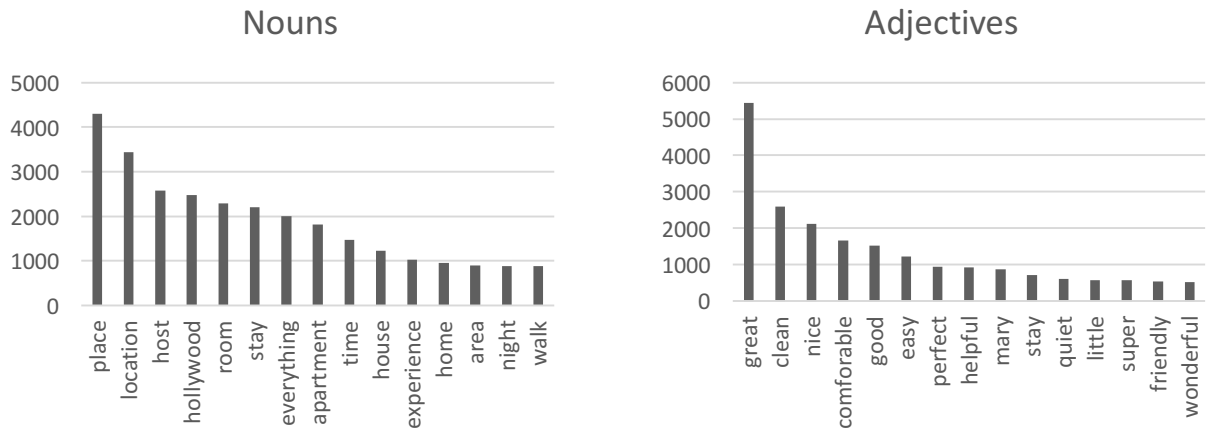

Figure 8: Analysis for Hollywood area

Venice: unique noun, beach; unique adjective, N/A.

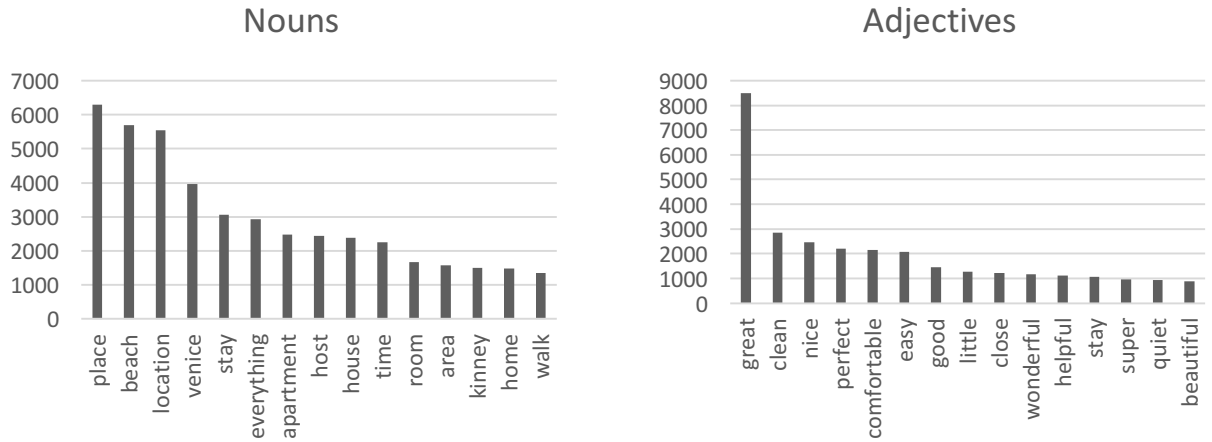

Figure 9: Analysis for Venice area

Oakland: unique noun, space; unique adjective, N/A.

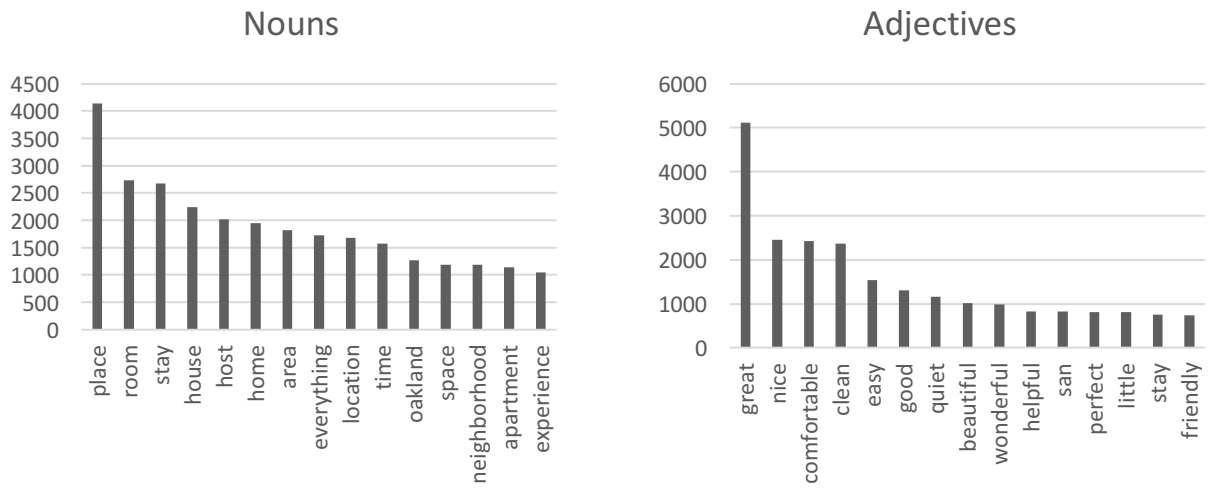

Figure 10: Analysis for Oakland Area

Manhattan: unique noun, park; unique adjective, central.

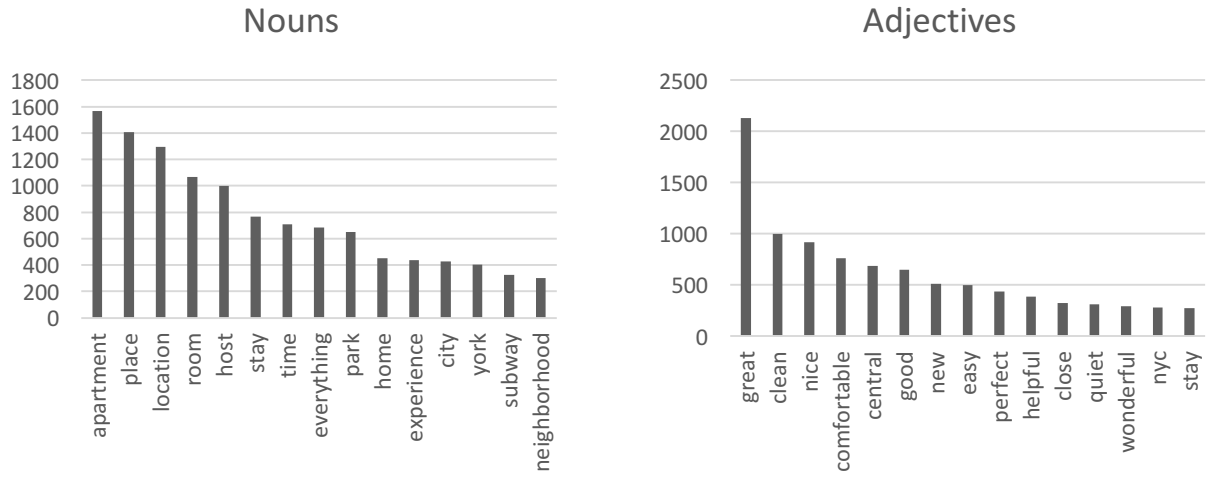

Figure 11: Analysis for Manhattan Area

Brooklyn: unique noun, N/A; unique adjective, safe.

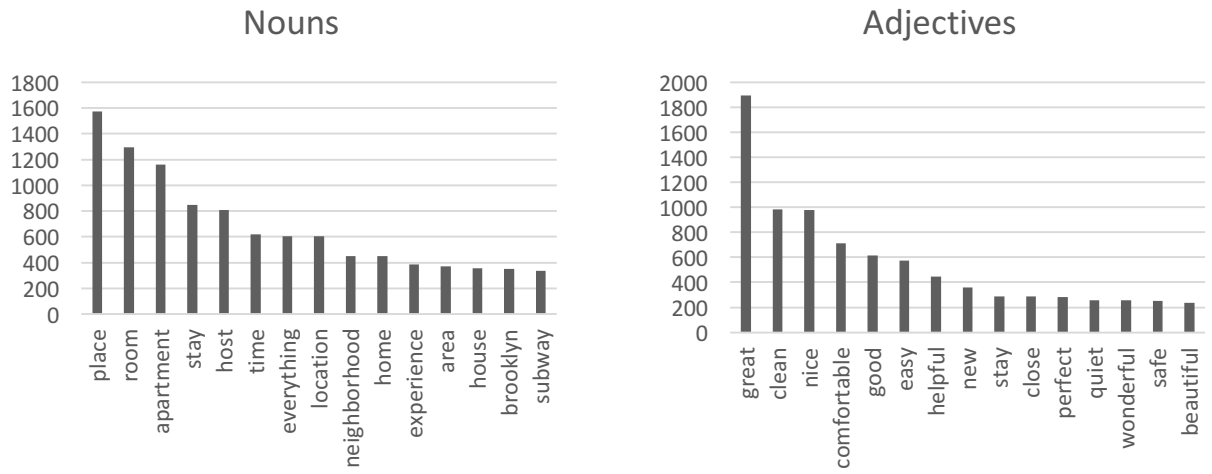

Figure 12: Analysis for Brooklyn Area

The lexical densities of the reviews in the five area are also calculated as follow:

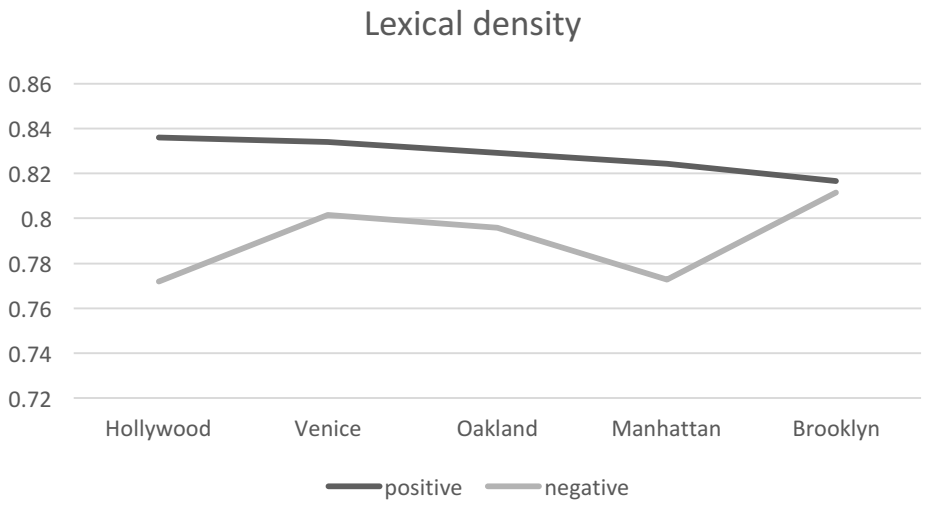

Figure 13: Lexical density per area

Results are all shown as graphs for clarity and readability. For exact numbers and complementary analysis on negative reviews and big reviews (reviews for listings with more than fifty reviews), please see the output of the analysis script.

5 EVALUATION

Through the results, users' preference in different areas can be inferred. Users in Hollywood area pay attention to their experiences at night. Users in Venice area pay attention to their distances with the beach. Users in Oakland area pay attention to the spaces of their accommodations. Users in Manhattan area pay attention to the central park. Users in Brooklyn area pay attention to their safety and transportation.

In addition, the lexical densities of the positive reviews are basically higher than those of negative reviews in each area. Perhaps users like to use more vocabulary on describing their positive experience than complaining about negative ones. Or maybe the lexical density of positive reviews and negative reviews is not comparable, because the amounts of data gathered of these two kinds of reviews have a huge discrepancy.

Another evaluation must be taken on the sentiment analysis. The accuracy of the sentiment analysis is not perfect. The polarity of a review is relative. Several errors could be found. Some negative reviews are mostly positive and some negative reviews turn out to be very positive. For example, one negative review in Hollywood says "Ricky was a perfect host. No complaints!" Apparently, it is difficult to judge the polarity of words such as "perfect" and "complaint". In the other hand, many positive reviews are also labeled as neutral reviews due to ambiguity of vocabulary. Additionally, reviews in foreign language or with special symbols are also filtered in neutral reviews, because the algorithm cannot calculate those polarities. Therefore, the neutral reviews classified by sentiment analysis are not used in the study in this project. In conclusion, we must take it into consideration that the algorithm cannot calculate the polarity with a full hundred percent of accuracy.

6 CONCLUSION

This project has only been carried out for a few months. It still needs further development and improvement. In the future, more techniques of processing and analysis of the texts could be included. For example, key words of every sentence may be filtered. A study with more detailed processing may be more informative and convincing. In addition, more detailed analysis could be done on smaller scales. For example, within one neighborhood, the reviews of different listings may be comparable. The description of the listings may also be useful to consider when analyzing the reviews. With more effort involved, this project could have more insights and be more promising.

7 REFERENCE

[1] Gong, A. & Lu, J. (2013). Picking Out Good Dishes from Yelp. Stanford, CA, June 2013.

[2] 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.

APPENDIX I

The codes of this project can be found at the following link: https://github.com/mrsata/Airbnb-ReviewAnalyzer. The documentation of the API used in this paper can be found at the following link: http://airbnbapi.org.

APPENDIX II

Below is the sample output of the analysis:

```
AirbnbReviewAnalyzer v1.0

------ Analysis for positive comments at hollywood ------
--- Most frequent tokens ---
 [('.', 29305), ('and', 21311), ('the', 21129), (',', 14793), ('was', 14364), ('to',
13573), ('a', 12654), ('is', 8693), ('!', 7990), ('i', 7666), ('in', 7445), ('very', 6800),
('we', 6035), ('great', 5426), ('it', 5241)]
--- Tokens without punctuation ---
 [('and', 21328), ('the', 21178), ('was', 14191), ('to', 13589), ('a', 12856), ('is',
8655), ('i', 7745), ('in', 7655), ('very', 6808), ('we', 6050), ('great', 5438), ('it',
5259), ('for', 5061), ('of', 5041), ('place', 4589)]
--- Most frequent words ---
 [('great', 5438), ('place', 4589), ('stay', 4532), ('location', 3542), ('hollywood',
3296), ('clean', 3166), ('host', 2738), ('us', 2373), ('nice', 2335), ('room', 2294),
('would', 2272), ('apartment', 2150), ('everything', 2006), ('really', 1773), ('recommend',
1700)]
--- Most frequent nouns ---
 [('place', 4300), ('location', 3442), ('host', 2572), ('hollywood', 2471), ('room', 2294),
('stay', 2205), ('everything', 2006), ('apartment', 1815), ('time', 1465), ('house', 1227),
('experience', 1025), ('home', 956), ('area', 897), ('night', 882), ('walk', 876)]
--- Most frequent adjective ---
 [('great', 5438), ('clean', 2601), ('nice', 2116), ('comfortable', 1663), ('good', 1530),
('easy', 1218), ('perfect', 944), ('helpful', 928), ('mary', 863), ('stay', 716), ('quiet',
615), ('little', 577), ('super', 567), ('friendly', 530), ('wonderful', 521)]
--- Average lexical density ---
 0.8360986295924326
------ Analysis for negative comments at hollywood ------
--- Most frequent tokens ---
```

```
 [('the', 718), ('.', 717), (',', 394), ('was', 390), ('to', 382), ('and', 360), ('i',
295), ('a', 228), ('in', 186), ('of', 155), ('we', 146), ('it', 140), ('not', 136),
('that', 133), ('is', 127)]
--- Tokens without punctuation ---
 [('the', 723), ('to', 382), ('was', 375), ('and', 360), ('i', 296), ('a', 232), ('in',
192), ('of', 155), ('we', 147), ('it', 140), ('not', 135), ('that', 133), ('is', 126),
('for', 124), ('had', 104)]
--- Most frequent words ---
 [('room', 66), ('stay', 58), ('place', 57), ('night', 54), ('parking', 48), ('one', 45),
('host', 42), ('us', 41), ('would', 38), ('apartment', 37), ('airbnb', 33), ('good', 32),
('location', 26), ('get', 26), ('also', 26)]
--- Most frequent nouns ---
 [('room', 66), ('night', 54), ('place', 53), ('host', 37), ('stay', 35), ('apartment',
27), ('location', 24), ('day', 23), ('time', 23), ('experience', 22), ('house', 18),
('door', 18), ('airbnb', 18), ('area', 18), ('parking', 17)]
--- Most frequent adjective ---
 [('good', 31), ('bad', 19), ('clean', 17), ('great', 17), ('nice', 15), ('next', 13),
('able', 11), ('comfortable', 11), ('first', 10), ('airbnb', 10), ('old', 10), ('little',
9), ('overall', 9), ('last', 9), ('late', 8)]
--- Average lexical density ---
 0.771787799999943
------ Analysis for big reviews at hollywood ------
--- Most frequent tokens ---
 [('.', 22391), ('and', 16224), ('the', 16107), (',', 11284), ('was', 11115), ('to',
10437), ('a', 9577), ('is', 6678), ('!', 5939), ('i', 5796), ('in', 5709), ('very', 5197),
('we', 4959), ('it', 4029), ('great', 4008)]
--- Tokens without punctuation ---
 [('and', 16238), ('the', 16152), ('was', 10980), ('to', 10448), ('a', 9749), ('is', 6654),
('in', 5865), ('i', 5850), ('very', 5204), ('we', 4974), ('it', 4042), ('great', 4018),
('for', 3856), ('of', 3809), ('stay', 3406)]
--- Most frequent words ---
 [('great', 4018), ('stay', 3406), ('place', 3396), ('location', 2710), ('hollywood',
2517), ('clean', 2496), ('host', 2075), ('us', 1957), ('room', 1941), ('nice', 1806),
('would', 1721), ('everything', 1532), ('apartment', 1528), ('really', 1393), ('recommend',
1305)]
--- Most frequent nouns ---
 [('place', 3183), ('location', 2635), ('host', 1956), ('room', 1941), ('hollywood', 1885),
('stay', 1658), ('everything', 1532), ('apartment', 1282), ('time', 1132), ('house', 1036),
('experience', 777), ('home', 730), ('night', 711), ('walk', 673), ('area', 664)]
--- Most frequent adjective ---
 [('great', 4018), ('clean', 2033), ('nice', 1640), ('comfortable', 1231), ('good', 1215),
('easy', 899), ('mary', 843), ('helpful', 742), ('perfect', 716), ('stay', 533), ('quiet',
455), ('little', 443), ('friendly', 412), ('super', 412), ('wonderful', 390)]
--- Average lexical density ---
 0.8332545348824727
------ Analysis for positive comments at venice ------
--- Most frequent tokens ---
 [('.', 43799), ('the', 37571), ('and', 34105), (',', 23675), ('to', 21199), ('a', 20178),
('was', 18815), ('!', 12307), ('is', 11398), ('in', 11224), ('we', 10943), ('i', 8915),
('great', 8492), ('very', 8379), ('for', 8032)]
--- Tokens without punctuation ---
 [('the', 37628), ('and', 34126), ('to', 21229), ('a', 20365), ('was', 18621), ('in',
11581), ('is', 11315), ('we', 10974), ('i', 8989), ('great', 8506), ('very', 8385), ('for',
8034), ('it', 7860), ('of', 7612), ('beach', 7508)]
--- Most frequent words ---
 [('great', 8506), ('beach', 7508), ('place', 6700), ('stay', 6430), ('venice', 5685),
('location', 5671), ('clean', 3432), ('would', 3429), ('perfect', 3208), ('us', 2999),
('apartment', 2946), ('everything', 2922), ('nice', 2667), ('host', 2619), ('really',
2443)]
--- Most frequent nouns ---
 [('place', 6286), ('beach', 5689), ('location', 5549), ('venice', 3973), ('stay', 3064),
('everything', 2922), ('apartment', 2480), ('host', 2434), ('house', 2389), ('time', 2256),
('room', 1679), ('area', 1583), ('kinney', 1497), ('home', 1487), ('walk', 1349)]
--- Most frequent adjective ---
```

```
 [('great', 8506), ('clean', 2860), ('nice', 2459), ('perfect', 2193), ('comfortable',
2157), ('easy', 2063), ('good', 1457), ('little', 1264), ('close', 1223), ('wonderful',
1170), ('helpful', 1105), ('stay', 1071), ('super', 947), ('quiet', 938), ('beautiful',
883)]
--- Average lexical density ---
 0.8341328290364753
------ Analysis for negative comments at venice ------
--- Most frequent tokens ---
 [('the', 722), ('.', 702), (',', 391), ('was', 313), ('to', 309), ('and', 304), ('a',
241), ('we', 190), ('in', 174), ('is', 170), ('it', 159), ('i', 153), ('of', 147), ('not',
136), ('for', 122)]
--- Tokens without punctuation ---
 [('the', 727), ('to', 309), ('and', 304), ('was', 301), ('a', 243), ('we', 191), ('in',
179), ('is', 166), ('it', 161), ('i', 157), ('of', 148), ('not', 134), ('for', 124),
('that', 112), ('but', 100)]
--- Most frequent words ---
 [('place', 74), ('us', 66), ('beach', 50), ('bed', 46), ('house', 43), ('location', 42),
('stay', 40), ('host', 38), ('one', 35), ('get', 34), ('apartment', 33), ('would', 30),
('night', 28), ('room', 27), ('airbnb', 26)]
--- Most frequent nouns ---
 [('place', 73), ('house', 43), ('location', 41), ('host', 37), ('beach', 35),
('apartment', 30), ('night', 28), ('room', 27), ('bed', 21), ('door', 20), ('day', 18),
('space', 18), ('experience', 18), ('area', 18), ('bathroom', 16)]
--- Most frequent adjective ---
 [('good', 23), ('clean', 19), ('great', 17), ('nice', 15), ('bad', 15), ('small', 14),
('hot', 13), ('uncomfortable', 13), ('little', 12), ('big', 11), ('outside', 11),
('overall', 10), ('loud', 10), ('due', 9), ('airbnb', 9)]
--- Average lexical density ---
 0.8015246211131662
------ Analysis for big reviews at venice ------
--- Most frequent tokens ---
 [('.', 34811), ('the', 30122), ('and', 27223), (',', 19027), ('to', 16952), ('a', 16100),
('was', 14900), ('!', 9767), ('we', 9071), ('in', 9024), ('is', 8986), ('i', 7064),
('very', 6693), ('great', 6597), ('for', 6455)]
--- Tokens without punctuation ---
 [('the', 30174), ('and', 27238), ('to', 16976), ('a', 16243), ('was', 14744), ('in',
9303), ('we', 9097), ('is', 8915), ('i', 7114), ('very', 6697), ('great', 6606), ('for',
6459), ('it', 6357), ('of', 6135), ('beach', 5927)]
--- Most frequent words ---
 [('great', 6606), ('beach', 5927), ('place', 5199), ('stay', 5091), ('venice', 4461),
('location', 4324), ('would', 2679), ('clean', 2638), ('us', 2544), ('perfect', 2503),
('everything', 2356), ('apartment', 2287), ('nice', 2048), ('host', 1945), ('really',
1912)]
--- Most frequent nouns ---
 [('place', 4886), ('beach', 4504), ('location', 4230), ('venice', 3108), ('stay', 2412),
('everything', 2356), ('apartment', 1923), ('house', 1892), ('host', 1810), ('time', 1806),
('room', 1306), ('area', 1249), ('kinney', 1168), ('home', 1166), ('space', 1095)]
--- Most frequent adjective ---
 [('great', 6606), ('clean', 2205), ('nice', 1881), ('perfect', 1706), ('comfortable',
1702), ('easy', 1609), ('good', 1162), ('little', 1062), ('wonderful', 966), ('close',
965), ('stay', 849), ('helpful', 836), ('super', 720), ('quiet', 714), ('fantastic', 705)]
--- Average lexical density ---
 0.8319462921593848
------ Analysis for positive comments at oakland ------
--- Most frequent tokens ---
 [('.', 34084), ('and', 27412), ('the', 25408), (',', 18604), ('was', 15367), ('a', 15216),
('to', 15006), ('is', 8688), ('i', 8338), ('in', 8231), ('we', 8040), ('!', 7703), ('very',
7537), ('for', 5703), ('of', 5644)]
--- Tokens without punctuation ---
 [('and', 27436), ('the', 25449), ('a', 15336), ('was', 15198), ('to', 15035), ('is',
8640), ('in', 8458), ('i', 8384), ('we', 8057), ('very', 7547), ('for', 5706), ('of',
5650), ('it', 5503), ('stay', 5150), ('great', 5122)]
--- Most frequent words ---
```

```
 [('stay', 5150), ('great', 5122), ('place', 4383), ('clean', 2857), ('us', 2803), ('room',
2732), ('nice', 2660), ('comfortable', 2475), ('would', 2434), ('house', 2237), ('oakland',
2200), ('host', 2168), ('home', 2084), ('area', 1819), ('everything', 1725)]
--- Most frequent nouns ---
 [('place', 4135), ('room', 2732), ('stay', 2675), ('house', 2237), ('host', 2022),
('home', 1943), ('area', 1819), ('everything', 1725), ('location', 1682), ('time', 1576),
('oakland', 1268), ('space', 1189), ('neighborhood', 1183), ('apartment', 1140),
('experience', 1042)]
--- Most frequent adjective ---
 [('great', 5122), ('nice', 2452), ('comfortable', 2423), ('clean', 2368), ('easy', 1544),
('good', 1307), ('quiet', 1167), ('beautiful', 1012), ('wonderful', 985), ('helpful', 830),
('san', 827), ('perfect', 816), ('little', 814), ('stay', 757), ('friendly', 745)]
--- Average lexical density ---
 0.829175966549314
------ Analysis for negative comments at oakland ------
--- Most frequent tokens ---
 [('the', 664), ('.', 654), ('and', 367), (',', 332), ('was', 328), ('to', 301), ('a',
283), ('i', 202), ('is', 189), ('in', 189), ('we', 179), ('it', 167), ('of', 148), ('that',
119), ('for', 116)]
--- Tokens without punctuation ---
 [('the', 664), ('and', 367), ('was', 312), ('to', 302), ('a', 284), ('i', 202), ('in',
192), ('is', 186), ('we', 179), ('it', 169), ('of', 148), ('that', 119), ('for', 116),
('not', 109), ('with', 103)]
--- Most frequent words ---
 [('room', 75), ('place', 66), ('house', 51), ('stay', 50), ('us', 42), ('one', 40),
('parking', 37), ('clean', 37), ('host', 36), ('dirty', 36), ('night', 36), ('bathroom',
32), ('apartment', 29), ('bed', 29), ('really', 29)]
--- Most frequent nouns ---
 [('room', 75), ('place', 58), ('house', 51), ('night', 36), ('host', 32), ('bathroom',
32), ('time', 27), ('stay', 26), ('experience', 26), ('area', 26), ('apartment', 25),
('car', 22), ('street', 21), ('problem', 19), ('everything', 19)]
--- Most frequent adjective ---
 [('clean', 33), ('bad', 22), ('nice', 21), ('good', 20), ('great', 16), ('full', 15),
('dirty', 15), ('little', 13), ('key', 13), ('comfortable', 11), ('private', 11), ('stay',
11), ('able', 10), ('late', 10), ('first', 10)]
--- Average lexical density ---
 0.7957405197621659
------ Analysis for big reviews at oakland ------
--- Most frequent tokens ---
 [('.', 22230), ('and', 18382), ('the', 16641), (',', 12194), ('a', 9931), ('was', 9914),
('to', 9822), ('we', 5794), ('is', 5690), ('in', 5440), ('i', 5202), ('!', 5172), ('very',
4988), ('for', 3708), ('of', 3683)]
--- Tokens without punctuation ---
 [('and', 18398), ('the', 16667), ('a', 10034), ('to', 9835), ('was', 9817), ('we', 5807),
('is', 5666), ('in', 5582), ('i', 5231), ('very', 4995), ('for', 3710), ('of', 3687),
('it', 3594), ('stay', 3302), ('with', 3270)]
--- Most frequent words ---
 [('stay', 3302), ('great', 3266), ('place', 2768), ('us', 1991), ('clean', 1792), ('nice',
1767), ('room', 1755), ('comfortable', 1652), ('would', 1567), ('house', 1495), ('oakland',
1424), ('home', 1380), ('host', 1350), ('area', 1209), ('everything', 1146)]
--- Most frequent nouns ---
 [('place', 2621), ('room', 1755), ('stay', 1732), ('house', 1495), ('home', 1289),
('host', 1253), ('area', 1209), ('everything', 1146), ('time', 1098), ('location', 1034),
('space', 813), ('oakland', 800), ('neighborhood', 764), ('experience', 696), ('apartment',
679)]
--- Most frequent adjective ---
 [('great', 3266), ('nice', 1634), ('comfortable', 1621), ('clean', 1486), ('easy', 957),
('good', 812), ('quiet', 773), ('beautiful', 720), ('wonderful', 678), ('san', 555),
('little', 553), ('helpful', 542), ('perfect', 535), ('private', 497), ('stay', 489)]
--- Average lexical density ---
 0.825148880984284
------ Analysis for positive comments at manhattan ------
--- Most frequent tokens ---
```

```
 [('.', 12133), ('and', 9609), ('the', 9349), (',', 6999), ('to', 5750), ('was', 5522),
('a', 5298), ('is', 3937), ('i', 3129), ('in', 3120), ('!', 3105), ('very', 2982), ('we',
2291), ('of', 2132), ('great', 2127)]
--- Tokens without punctuation ---
 [('and', 9618), ('the', 9364), ('to', 5757), ('was', 5471), ('a', 5324), ('is', 3920),
('in', 3217), ('i', 3142), ('very', 2983), ('we', 2296), ('of', 2136), ('great', 2132),
('for', 2053), ('it', 2028), ('apartment', 1874)]
--- Most frequent words ---
 [('great', 2132), ('apartment', 1874), ('stay', 1673), ('place', 1498), ('location',
1328), ('clean', 1189), ('room', 1067), ('host', 1059), ('nice', 1005), ('us', 1002),
('would', 910), ('comfortable', 780), ('really', 759), ('subway', 749), ('time', 708)]
--- Most frequent nouns ---
 [('apartment', 1565), ('place', 1408), ('location', 1293), ('room', 1067), ('host', 1000),
('stay', 766), ('time', 708), ('everything', 685), ('park', 653), ('home', 450),
('experience', 435), ('city', 429), ('york', 405), ('subway', 328), ('neighborhood', 303)]
--- Most frequent adjective ---
 [('great', 2132), ('clean', 999), ('nice', 919), ('comfortable', 759), ('central', 686),
('good', 648), ('new', 508), ('easy', 497), ('perfect', 433), ('helpful', 386), ('close',
323), ('quiet', 314), ('wonderful', 294), ('nyc', 279), ('stay', 274)]
--- Average lexical density ---
 0.8243733504000437
------ Analysis for negative comments at manhattan ------
--- Most frequent tokens ---
 [('the', 195), ('.', 164), (',', 112), ('and', 103), ('to', 88), ('was', 73), ('in', 54),
('i', 53), ('a', 52), ('it', 46), ('we', 46), ('not', 41), ("n't", 38), ('is', 36), ('of',
33)]
--- Tokens without punctuation ---
 [('the', 195), ('and', 104), ('to', 88), ('was', 70), ('in', 58), ('i', 53), ('a', 53),
('it', 46), ('we', 46), ('not', 41), ('t', 38), ('is', 36), ('of', 33), ('room', 27),
('apartment', 26)]
--- Most frequent words ---
 [('room', 27), ('apartment', 26), ('us', 17), ('stay', 16), ('location', 13), ('host',
13), ('one', 12), ('good', 10), ('check', 9), ('bathroom', 9), ('clean', 9), ('time', 9),
('place', 9), ('really', 9), ('floor', 8)]
--- Most frequent nouns ---
 [('room', 27), ('apartment', 18), ('location', 12), ('host', 10), ('bathroom', 9),
('time', 9), ('place', 8), ('stay', 8), ('floor', 8), ('air', 7), ('check', 6), ('night',
5), ('day', 5), ('shower', 5), ('water', 5)]
--- Most frequent adjective ---
 [('good', 10), ('clean', 8), ('apartment', 5), ('bad', 4), ('hard', 4), ('small', 4),
('stay', 4), ('big', 3), ('nice', 3), ('new', 3), ('ok', 3), ('overall', 3), ('key', 3),
('dirty', 3), ('previous', 3)]
--- Average lexical density ---
 0.7729142910688783
------ Analysis for big reviews at manhattan ------
--- Most frequent tokens ---
 [('.', 4405), ('and', 3503), ('the', 3475), (',', 2558), ('to', 2161), ('a', 2048),
('was', 1911), ('is', 1486), ('i', 1180), ('in', 1174), ('very', 1013), ('!', 982), ('of',
853), ('we', 834), ('for', 732)]
--- Tokens without punctuation ---
 [('and', 3510), ('the', 3481), ('to', 2164), ('a', 2063), ('was', 1887), ('is', 1483),
('in', 1203), ('i', 1183), ('very', 1014), ('of', 855), ('we', 835), ('for', 732), ('with',
691), ('great', 686), ('it', 683)]
--- Most frequent words ---
 [('great', 686), ('apartment', 588), ('stay', 580), ('room', 481), ('place', 439),
('location', 432), ('host', 419), ('us', 382), ('clean', 379), ('comfortable', 324),
('nice', 315), ('would', 305), ('really', 276), ('subway', 271), ('time', 261)]
--- Most frequent nouns ---
 [('apartment', 500), ('room', 481), ('location', 421), ('place', 415), ('host', 395),
('stay', 270), ('time', 261), ('park', 207), ('everything', 205), ('home', 188), ('york',
172), ('city', 167), ('experience', 167), ('mark', 164), ('subway', 117)]
--- Most frequent adjective ---
 [('great', 686), ('clean', 325), ('comfortable', 314), ('nice', 294), ('good', 237),
('central', 220), ('new', 217), ('easy', 149), ('helpful', 143), ('perfect', 139),
('quiet', 129), ('friendly', 113), ('wonderful', 106), ('nyc', 104), ('susan', 103)]
```

```
--- Average lexical density ---
 0.8034220612507289
------ Analysis for positive comments at brooklyn ------
--- Most frequent tokens ---
 [('.', 11642), ('and', 9600), ('the', 9346), (',', 6709), ('to', 5578), ('was', 5404),
('a', 5065), ('is', 3553), ('i', 3203), ('in', 3123), ('very', 2958), ('!', 2750), ('we',
2339), ('it', 2040), ('for', 2005)]
--- Tokens without punctuation ---
 [('and', 9613), ('the', 9367), ('to', 5587), ('was', 5323), ('a', 5102), ('is', 3539),
('i', 3225), ('in', 3187), ('very', 2962), ('we', 2347), ('it', 2058), ('for', 2007),
('of', 1904), ('great', 1894), ('with', 1783)]
--- Most frequent words ---
 [('great', 1894), ('stay', 1726), ('place', 1682), ('apartment', 1411), ('room', 1294),
('clean', 1186), ('nice', 1069), ('us', 891), ('host', 851), ('would', 843), ('really',
800), ('subway', 760), ('comfortable', 734), ('recommend', 720), ('brooklyn', 628)]
--- Most frequent nouns ---
 [('place', 1573), ('room', 1294), ('apartment', 1162), ('stay', 850), ('host', 807),
('time', 618), ('everything', 606), ('location', 602), ('neighborhood', 452), ('home',
448), ('experience', 385), ('area', 372), ('house', 358), ('brooklyn', 353), ('subway',
337)]
--- Most frequent adjective ---
 [('great', 1894), ('clean', 985), ('nice', 977), ('comfortable', 714), ('good', 616),
('easy', 572), ('helpful', 446), ('new', 357), ('stay', 288), ('close', 287), ('perfect',
281), ('quiet', 258), ('wonderful', 256), ('safe', 253), ('beautiful', 234)]
--- Average lexical density ---
 0.8166837133873659
------ Analysis for negative comments at brooklyn ------
--- Most frequent tokens ---
 [('.', 163), ('the', 161), ('and', 97), (',', 84), ('was', 81), ('to', 76), ('i', 68),
('a', 56), ('we', 47), ('in', 45), ('it', 39), ("n't", 37), ('of', 31), ('is', 31),
('that', 30)]
--- Tokens without punctuation ---
 [('the', 161), ('and', 98), ('to', 77), ('was', 73), ('i', 70), ('a', 56), ('we', 47),
('in', 46), ('t', 40), ('it', 39), ('of', 31), ('that', 30), ('but', 30), ('is', 30),
('not', 30)]
--- Most frequent words ---
 [('room', 25), ('place', 15), ('apartment', 13), ('great', 12), ('house', 11), ('nadine',
11), ('host', 10), ('us', 10), ('left', 9), ('good', 9), ('one', 8), ('location', 8),
('recommend', 8), ('would', 7), ('well', 7)]
--- Most frequent nouns ---
 [('room', 25), ('place', 14), ('apartment', 12), ('house', 11), ('host', 9), ('location',
8), ('day', 7), ('night', 6), ('nadine', 6), ('experience', 6), ('door', 5), ('meet', 5),
('get', 5), ('minute', 4), ('person', 4)]
--- Most frequent adjective ---
 [('great', 12), ('good', 8), ('nice', 6), ('sure', 6), ('clean', 6), ('nadine', 5),
('last', 5), ('big', 4), ('fine', 4), ('new', 3), ('much', 3), ('comfortable', 3),
('wrong', 3), ('private', 3), ('small', 3)]
--- Average lexical density ---
 0.8113920663209117
------ Analysis for big reviews at brooklyn ------
--- Most frequent tokens ---
 [('.', 3279), ('the', 2635), ('and', 2583), (',', 1738), ('to', 1619), ('was', 1556),
('a', 1339), ('i', 956), ('is', 880), ('in', 826), ('very', 825), ('we', 705), ('!', 653),
('it', 616), ('for', 567)]
--- Tokens without punctuation ---
 [('the', 2639), ('and', 2585), ('to', 1621), ('was', 1525), ('a', 1348), ('i', 965),
('is', 871), ('in', 844), ('very', 825), ('we', 708), ('it', 619), ('for', 568), ('with',
474), ('of', 464), ('place', 460)]
--- Most frequent words ---
 [('place', 460), ('stay', 445), ('great', 410), ('apartment', 403), ('room', 381),
('nadine', 358), ('clean', 355), ('nice', 263), ('us', 240), ('would', 213), ('host', 208),
('really', 188), ('subway', 188), ('comfortable', 184), ('neighborhood', 182)]
--- Most frequent nouns ---
```

```
 [('place', 422), ('room', 381), ('apartment', 320), ('stay', 222), ('host', 196),
('everything', 162), ('time', 147), ('home', 133), ('night', 129), ('neighborhood', 125),
('experience', 122), ('nadine', 112), ('location', 108), ('house', 95), ('zach', 94)]
--- Most frequent adjective ---
 [('great', 410), ('clean', 298), ('nice', 240), ('nadine', 225), ('comfortable', 180),
('easy', 168), ('good', 166), ('helpful', 131), ('new', 89), ('safe', 84), ('close', 78),
('stay', 74), ('apartment', 68), ('quiet', 66), ('little', 64)]
--- Average lexical density ---
 0.8119263862952177
```