Text Mining Yelp Restaurant Reviews

DANIEL BAE
THOMAS COCHRAN
PATRICK CONLEY

Description: Goal

Goal:
Identify text patterns in restaurant reviews that can facilitate improvements in food, service, and review quality.

Two focal points of our inquiries:

1. Sentiment in frequent text patterns and similarities amongst Yelp users.
2. How review text patterns may vary based on business category, location, and seasonality.

Description: Questions

Identify text patterns for different restaurant categories:

- What are frequent features found in 1-star and 5-star restaurant reviews?
  - What are frequently used words?
  - How does review length vary?
- Are there nuanced text patterns among major cities in the US?
- Can seasonal changes affect text sentiment?

Identify text patterns among Yelp users:

- What makes a user's review tagged as Useful, Funny, or Cool?
- Are there clusters of Yelp users who more frequently give positive or negative reviews?
  - What is common among these users?
  - Are there associations between total review count, average star rating, or friend network?

Prior Work

Since this dataset was released by Yelp for academic purposes, there is a plethora of prior work.

Some relevant works include:

- Kaggle Notebooks
- Category Predictor, Review Autocomplete
- Inferring Future Business Attention
- Sentiment Analysis for Yelp Review Classification
- Identifying Social Sub-Groups Through Clustering of Yelp User Data
- Text Mining and Sentiment Analysis for Yelp Reviews of A Burger Chain

Dataset

The Yelp Open Dataset:

- 5 million user reviews
- 170 thousand businesses
- 11 metropolitan areas.
- 5 json files

Download Link: Yelp Dataset

Documentation: Yelp dataset attributes and types

Files:

yelp_academic_dataset_business.json

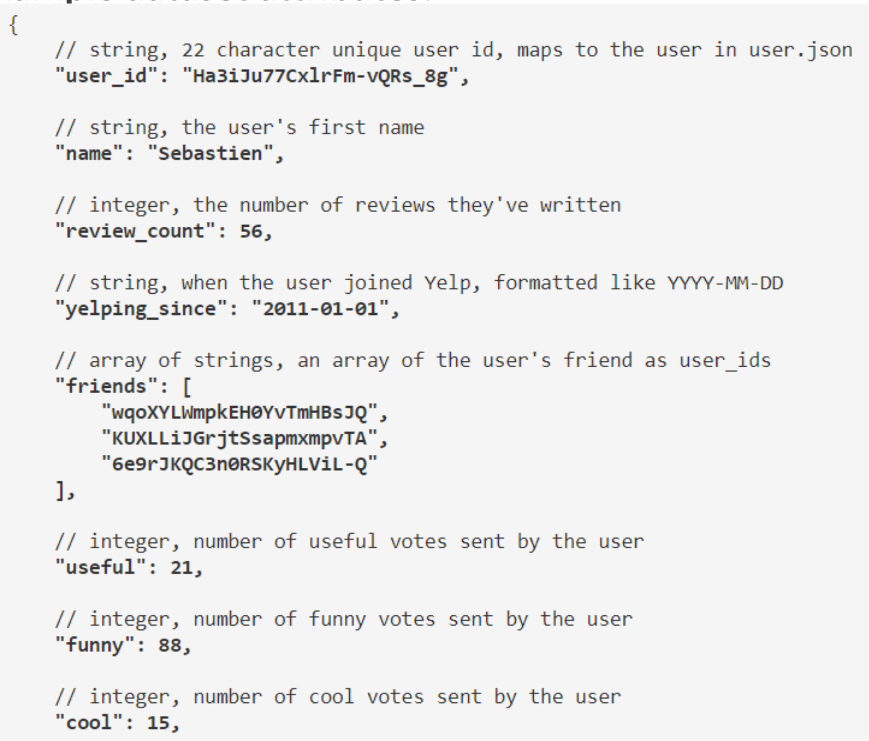
yelp_academic_dataset_checkin.json

yelp_academic_dataset_review.json

yelp_academic_dataset_tip.json

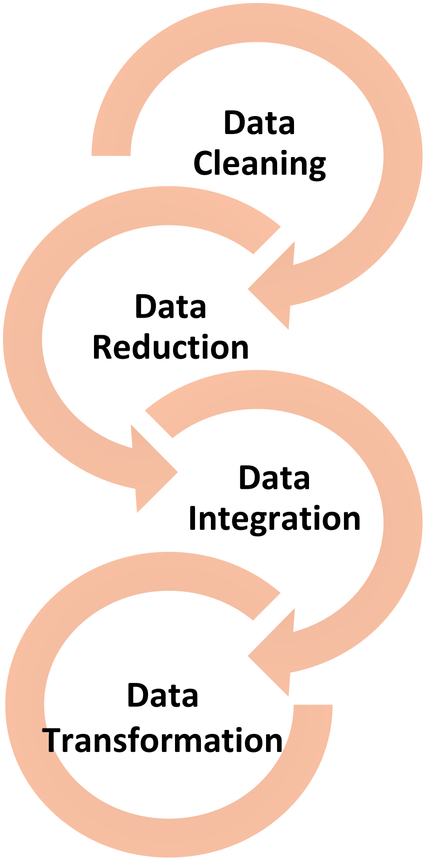
yelp_academic_dataset_user.json

Example dataset attributes:



```
{
    // string, 22 character unique user id, maps to the user in user.json
    "user_id": "Ha3iJu77CxlrFm-vQRs_8g",

    // string, the user's first name
    "name": "Sebastien",

    // integer, the number of reviews they've written
    "review_count": 56,

    // string, when the user joined Yelp, formatted like YYYY-MM-DD
    "yelping_since": "2011-01-01",

    // array of strings, an array of the user's friend as user_ids
    "friends": [
        "wqoXYLWmpkEH0YvTmHBsJQ",
        "KUXLLiJGrjtSsapmxmpvTA",
        "6e9rJKQC3n0RSKyHLViL-Q"
    ],

    // integer, number of useful votes sent by the user
    "useful": 21,

    // integer, number of funny votes sent by the user
    "funny": 88,

    // integer, number of cool votes sent by the user
    "cool": 15,
```

Proposed Work



Data Cleaning

- Filter non-English reviews and special characters.
- Verify all user data points are unique.
- Bin restaurants with similar types of cuisine.
- Detect users giving the highest or lowest possible rating.

Data Reduction

- Remove irrelevant attribute columns
- Remove business categories unrelated to food.
- Remove users with no text reviews.
- Remove redundant data using correlation analysis

Data Integration

- Compile all .json files into a relational database.
- Import relational database to the cloud.

Data Transformation

- Normalize user attributes using Z-Scores or Min-Max.
- Use the bag-of-words model to simplify review text.
- Model review text sentiment.
- Classify reviews as either negative or positive.

Tools

Development Environment

- Python (Link)
- Pycharm (Link)
- Jupyter Notebook (Link)

Data analysis and statistics

- Pandas (Link)
- Numpy (Link)
- NetworkX (Link)
- matplotlib (Link)

Data storage and integration

- SQLite (Link)
- Google Cloud (Link)
- Amazon Cloud (Link)

Text processing and classification

- NLTK (Link)
- TextBlob (Link)

Evaluation Metrics

Clustering and similarity measures:

- **Cluster** users grouped by star review.
- Cluster reviews grouped by:
  - location
  - word frequency and count
  - month posted
- Similarity measures
  - **Minkowski distance**
  - **Euclidean distance**

Association rules and pattern evaluation:

- Identify frequent words across ratings
- Identify **associations**, for example:
  - 1-star review → funny review tag
  - Positive sentiment → useful review tag
- Examine interestingness of associations:
  - **Kulczynski**
  - **Jaccard**

Community Detection:

- Identify friend list communities:
  - **Louvain method**
  - **Label Propagation**

Text Classification:

- Evaluate text sentiment using:
  - **Natural language processing**
  - **Lexicon-based sentiment analysis**