Restaurant Sentiment Mapping from Yelp Reviews

TeamBD (#110) - Yu-Ying Liao, Matthew Matsuo, Shinnosuke Jay Nonaka, Akira Taniguchi

Introduction

Our team's goal is to visualize the sentiment patterns of restaurant reviews within New York City. The final deliverable is an interactive dashboard enabling users to identify, compare, and understand the sentiment patterns and characteristics of different groupings of restaurants. The major components of our methodology consist of data collection and preparation, the application of natural language processing algorithms (NLP) for sentiment analysis, and the use of clustering methods utilizing both the results of the sentiment analysis and geospatial attributes as inputs. Our efforts provide both local residents and restaurants with insights enabling them to find their next favorite restaurant or identify competitors.

Problem Definition

Despite the wealth of restaurant review data that is available on various social media, no tool or resource exists today to let users understand review sentiment patterns towards restaurants. Currently, users can read individual reviews and identify simple metrics about restaurants such as ratings and review count to try to gauge overall sentiment. However, it is impossible to fully understand sentiment patterns across all restaurants in a major city with this limited approach. A tool with the ability to fully leverage restaurant review data to analyze and visualize sentiment patterns would address this problem. In addition, sentiment score is used over star rating as input variable for clustering because it provides a more nuanced understanding of customer reviews. This tool would benefit residents seeking insights into the sentiments of restaurants across different neighborhoods in their city and local restaurants who could utilize this analysis as an input for their strategic planning.

At a higher level, well-established implementations do not exist for our team's high-level approach to similar problems. While sentiment analysis is often performed at a much larger scale (e.g., regional, nationwide), and localized sentiment mapping within cities is less common. Furthermore, today's practices do not leverage the spatial dimension of sentiment review data limiting the ability to identify sentiment patterns and trends within a major city.

Literature Survey

User activity on social media is a valuable data source which can be analyzed to provide insight into residents' opinions leading to a better understanding of said residents' sentiment towards local services and events [1]. For our project, we leveraged Yelp's API to understand how local economies within cities are changing and the resulting impact on neighborhoods.

We must first preprocess the Yelp data. This may result in an increase in accuracy of our models and will reduce the number of classified neutrals. Removing stop words and certain special characters, clarifying abbreviations and slang, and applying lemmatization and spell check are all necessary steps to take before conducting sentiment analysis and implementing clustering methods [2].

To improve our natural language processing model, we need to move from a rudimentary bag of words approach which tracks word counts and ignores word order to a recursive neural tensor network which accounts for sentence structure and grammatical complexity [3]. We will also consider exploring a hybrid system framework that incorporates NLP models along with linguistic-based approaches. By implementing lexical databases and dictionary-based methods such as SentiWordNet Classifier to identify domain-relevant aspects, the models may increase classification accuracies and reduce the number of classified-neutrals [4]. After identifying aspects and categorizing the sentiments of each sentence based on said aspects, we may incorporate an Ordered Weighted Averaging (OWA) operator that will allow flexibility when aggregating different sentiments and scores within the overall Yelp review into a final sentiment rating [5].

We will perform spatial clustering on restaurants using geographical variables and other numeric attributes. One technique we will use is the density-based spatial clustering algorithm (DBSC) which considers both the spatial proximity and attribute similarity when clustering [6]. Some other techniques used in this paper for comparison purposes include k-means, CURE, GDBSCAN, Geo-SOM, and ASCDT. For our project, we will also leverage different models with several tuning parameters to identify the best model for clustering.

Finally, researchers typically validate results from unsupervised learning via both internal and external validation [7]. External validation will be difficult as we could not find similar research to validate our results against. As a result, our team must rely on internal validation by analyzing the structure of our results and following the appropriate, well-established internal validation methods [7]. One such example of internal validation is to visualize the output using a scatter plot to spot trends and clustering results.

Proposed Methodology

As a proof of concept, our team limited data collection to New York City and its surrounding area. Our team's original intent was to compare sentiment patterns across multiple major cities. However, we ultimately decided to prioritize building a quality dashboard with visualizations for one city which could be re-purposed for other major cities through future work.

Restaurant data was collected using Yelp's API which provided basic attributes for each restaurant. Only restaurants with ten or more reviews were collected because meaningful sentiment analysis cannot be conducted on restaurants with minimal reviews. Additionally, franchise fast food restaurants (e.g., McDonald's, Taco Bell, KFC, etc.) were also excluded from data collection. In total, restaurant data was collected for 18,189 restaurants. In addition to the restaurant data, user review text data on Yelp was collected for sentiment analysis. The initial plan was to also collect review text data using Yelp's API, but the API would only return three review excerpts per restaurant which would not have yielded a sufficient dataset. Therefore, our team relied on web scraping to collect the data. In total, over 300,000 reviews were scraped for sentiment analysis.

Next, we conducted text preprocessing to prepare the raw text reviews before running our NLP models. This was done in Python using the three packages re, NLTK, and contractions. Regex was used to remove HTML character codes, numbers, and the phrase "happy hour" from the text as we decided that numbers and time-related words were not necessary for sentiment analysis on food reviews. We then proceed to use the contractions package to further normalize the text. Contractions are a combination of words shortened by an apostrophe, which we replaced with the original words such as "I'd" to "I would." Furthermore, we used NLTK, or Natural Language Toolkit, to remove punctuations and English stop words that were given in the package, as well as additional stop words such as weekdays and numbers written out alphabetically. This was done after converting every letter in the text to lowercase because the list of stop words only includes lowercase versions of the stop words. We decided against correcting the spelling of words because we came across many restaurants serving different cuisines from around the world. Incorrectly spell-checking ethnic food names that are not part of the English dictionary would lead to problems down the road.

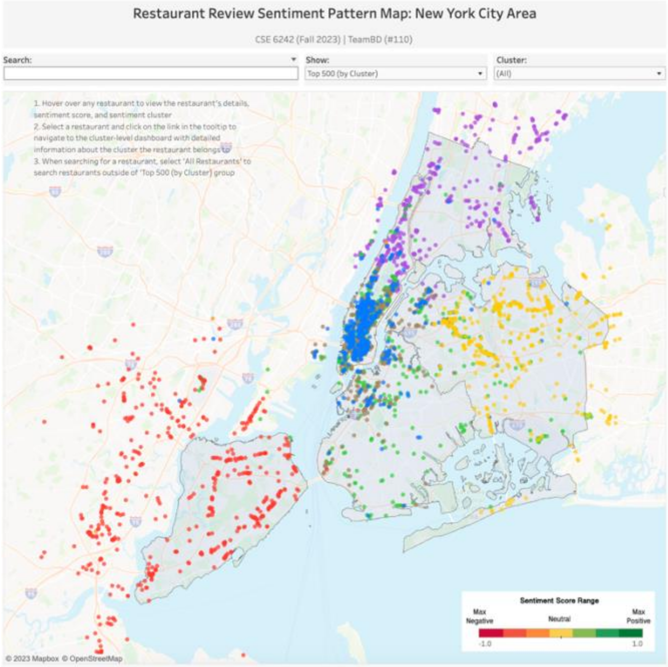
Taking the prepared and cleaned dataset, we tested several NLP models. The pre-trained models we used were taken from Hugging Face, an open-source collaborative model hosting site. After testing seven different models, we picked a model that met our criteria as the picked model had a low relative training time, positive and negative sentiment labels, and a non-heavily skewed distribution of output ranging from –1 to 1. Intuitively, this model was also perfect to use because, according to its documentation, it was trained on product review data so it makes sense that it would pick up correct sentiment on food review data. One last test we performed to confirm the picked model would work well was comparing its sentiment output label against the user's star rating. Indeed, the picked model performed great as it labeled 82% of one-star reviews as negative and labeled 98% of five-star reviews as positive. After running our chosen model on batches of reviews at a time to checkpoint the scoring process, we were left with a positive or negative sentiment label and a sentiment score ranging from –1 to 1 for each review. These reviews could now be used individually or summarized up to the restaurant level for further analysis.

Finally, we performed clustering on our final input dataset using variables including latitude, longitude, review count, rating, and average sentimental score to group restaurants into clusters. The three types of clustering algorithms we used include K-means, DBSCAN, and hierarchical clustering. These methods are chosen based on literature reviews and are suitable for identifying clusters based on spatial and feature similarities.

Since clustering algorithms work by measuring the similarity or distance between data points, we scaled our input variables prior to clustering. We applied scaling methods such as Min-Max and Standard Scaler on our non-geospatial variables. When using DBSCAN, we applied Min-Max scaling to preserve the original distances between data points so DBSCAN can function effectively. In addition, when using K-means cluster we applied Standard Scaler and the reason is that K-Means is a centroid-based clustering algorithm that uses Euclidean distance to determine the similarity or dissimilarity between data points. Euclidean distance is sensitive to the scale and variance of the data so scaling using Standard Scaler helps mitigate these issues. For geospatial variables, such as longitude and latitude, we didn't scale these variables because scaling these values can alter their geographic significance and make it difficult to interpret results in the context of real-world locations. Another reason is many clustering algorithms, including DBSCAN and K-Means, can handle geospatial data without requiring scaling.

Finally, our team developed an interactive dashboard using Tableau. The main dashboard displays a map of New York City area. Restaurants are plotted on the map using latitude and longitude data and are color-coded based on their cluster, giving users a clear visual of the groupings and an understanding of local sentiment patterns and trends. The dashboard also features tools to let users interact with the data. This includes a search bar, allowing users to quickly find specific restaurants and see how they compare to the overall populations. Moreover, a cluster filter allows users a quick way to find the restaurants in selected clusters. Lastly, due to the large size of our dataset, the dashboard experienced slow load times. To fix this, only the top 500 rated restaurants in each cluster are shown by default, though, users can opt to show all restaurants if they want to get the complete picture of a cluster or location.

In addition to the main dashboard, users are able to hover over each restaurant to display basic information about the restaurant and the restaurant's cluster via the tooltip. Furthermore, users will be able to access a detailed cluster-level dashboard via the tooltip if they would like to learn more about a cluster. The detailed cluster-level dashboard displays additional information including a restaurant sentiment score histogram, the most positive and negative sentiment categories, the category mix and their respective sentiment scores, and a word cloud representing the most mentioned review words in the cluster. This feature enables users to “double-click” into each cluster and gain a deeper understanding of the sentiment trends within a local area.



Restaurant Review Sentiment Pattern Map: New York City Area

CSE 6242 (Fall 2023) | TeamBD (#110)

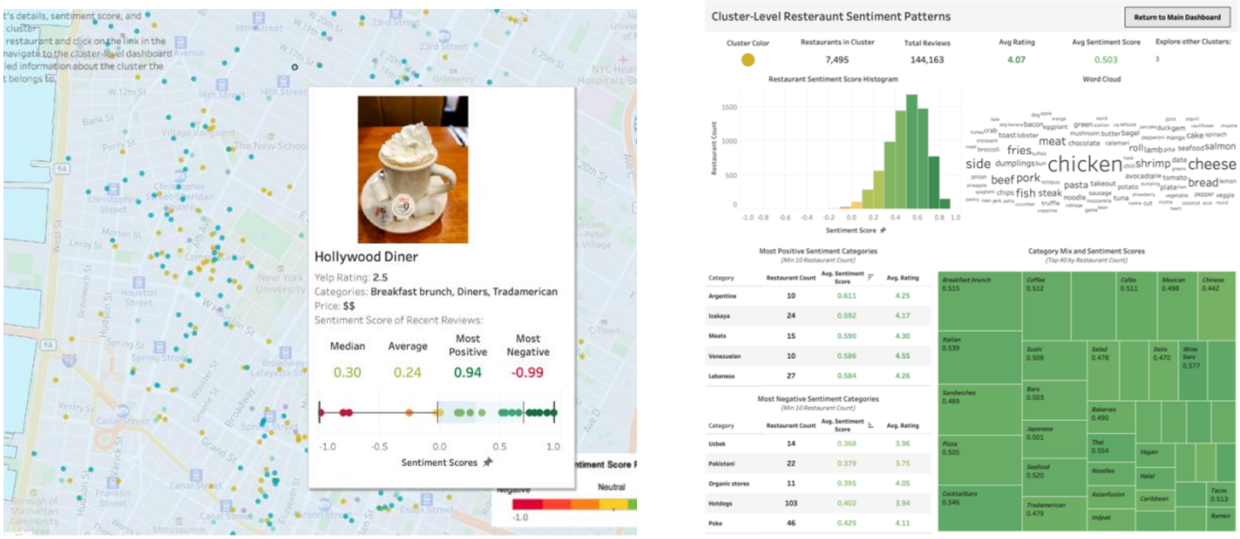
Search:

Show: Top 500 (by Cluster)

Cluster: (All)

1. Hover over any restaurant to view the restaurant's details, sentiment score, and sentiment cluster
2. Select a restaurant and click on the link in the tooltip to navigate to the cluster-level dashboard with detailed information about the cluster the restaurant belongs to
3. When searching for a restaurant, select 'All Restaurants' to search restaurants outside of 'Top 500 (by Cluster) group

Figure 1: *The dashboard design shows clustered restaurants in and around New York City.*



Hollywood Diner

Yelp Rating: 2.5

Categories: Breakfast brunch, Diners, Tradamerican

Price: $$

Sentiment Score of Recent Reviews:

Median	Average	Most Positive	Most Negative
0.30	0.24	0.94	-0.99

Cluster-Level Resteraunt Sentiment Patterns

Return to Main Dashboard

Cluster Color	Restaurants in Cluster	Total Reviews	Avg Rating	Avg Sentiment Score	Explore other Clusters:
●	7,495	144,163	4.07	0.503	3

Most Positive Sentiment Categories
(Min 10 Restaurant Count)

Category	Restaurant Count	Avg. Sentiment Score	Avg. Rating
Argentine	10	0.611	4.25
Izakaya	24	0.592	4.17
Meats	15	0.590	4.30
Venezuelan	10	0.586	4.55
Lebanese	27	0.584	4.26

Most Negative Sentiment Categories
(Min 10 Restaurant Count)

Category	Restaurant Count	Avg. Sentiment Score	Avg. Rating
Uzbek	14	0.368	3.96
Pakistani	22	0.379	3.75
Organic stores	11	0.395	4.05
Hotdogs	103	0.402	3.94
Poke	46	0.429	4.11

Figure 2: *The tooltip (left) for each restaurant shows an option to navigate to the detailed dashboard (right) for the restaurant's cluster.*

Experiments / Evaluation

Our team ran many tests to choose the best NLP model. First, we tested out seven different pre-trained models by training the same random sample of 1,000 reviews on each model. We focused on a few important metrics, the first of which was training time. Because we are dealing with hundreds of thousands of reviews, it was important to get an estimate of how much computing power would be needed to train the entire dataset. Besides that, it was crucial to understand the output of each model and how they differed from each other. For example, some models only produced positive and negative labels while others included a neutral label. Others even produced more distinct sentimental attributes such as excitement or gratitude. In the end, we had to weigh the pros and cons of the additional information and ultimately decided that the neutral label would only confuse the end user as it did not add any value and was a rare occurrence. Additionally, the detailed sentimental attributes could be an opportunity for further study, but for our project, we opted against it due to the increased training time. Finally, we had to look at the distribution of probability scores for each review. Along with each label, each review was given a confidence score in said label. Ideally, the output would be normally or uniformly distributed and not heavily skewed so that the score had more meaning and could be used as a feature in our clustering algorithm.

As for clustering, we performed parameter tuning for all clustering models. For K-means and DBSCAN, we used the elbow method and grid search to identify the optimal parameters for the K-mean cluster size and the DBSCAN's epsilon and minimum sample sizes. For hierarchical clustering, we used dendrogram to visualize the output and identify the optimal cluster size. In addition, we performed principal component analysis for feature dimension reduction. We evaluated clustering models for the different scaling method and PCA dataset using methods including silhouette score and visualization. Dendrograms and scatterplots were some of the visualization tools used to assess the quality of the clustering results. Our final model is K-means with k = 7 and a silhouette score of 0.3. We leveraged PCA on non-spatial features with n components = 2 and then normalized all columns using Standard Scaler after applying PCA for the input data set. In addition, we tried out several other approaches and had some models with relatively a high silhouette score of 0.5. However, these models resulted in only 2 clusters which would be too simple and generic for our purpose.

For the main dashboard, experiments primarily focused on improving dashboard design. Our team worked on refining the design for the main dashboard, the tooltip for each restaurant, as well as responsiveness. One main issue we experienced was the slow responsiveness of Tableau as it proved challenging to show the thousands of restaurants in our data set all at once. To combat this, we first thought to only display one cluster at a time since that would reduce the number of restaurants displayed at a time. However, this would remove the interaction between clusters, something we feel strongly about emphasizing. Instead, we opted to set the default view to only show the top 500 restaurants of each cluster. Users can still include all restaurants if they desire, however, it will greatly slow down the dashboard.

For the cluster-level dashboard, our team researched and experimented on a variety of different visualizations to convey important information on the sentiment cluster. Some visualizations we tested out include summary statistics, sentiment distribution graphs, and word clouds. One important test we ran was how to summarize specific cluster traits. For example, individual restaurants can have up to three different categories, all in different columns. At first, we attempted to compile the data in python and import an additional file in Tableau. This proved unnecessary, though, and after further research, we cut out the step entirely and ended up doing all our data manipulation straight in Tableau. While experimenting on our word cloud, we came across an issue where common words ended up with the most significance. We initially thought to cut the word cloud altogether, but later found a pre-built list of food-related words. Joining this list with the list of words in our reviews lets us filter out the filler and create a useful word cloud.

Our team iteratively developed and evaluated the design as a team to ensure that the resulting dashboard clearly captured the local sentiment trend of the clustered area.

Conclusions and Discussion

Our project aimed to visualize sentiment patterns by clusters in New York City restaurant reviews through a comprehensive methodology. It involved data collection and web scrapping from Yelp, preprocessing and cleaning of raw reviews, sentiment analysis using natural language processing, and spatial clustering via K-means of restaurants. The final deliverable was an interactive Tableau dashboard enabling users to explore sentiment patterns, compare restaurants, and gain insights into local sentiments across different neighborhoods.

The sentiment analysis utilized a carefully selected NLP model and the clustering algorithm (K-means) identified seven distinct sentiment-based clusters. The interactive dashboard provides an intuitive map of New York City, color-coded by sentiment clusters, and allows users the ability to explore sentiments geospatially. The tooltip feature allows for detailed insights into individual restaurants and their sentimental characteristics. The impact of our project was twofold, benefiting both residents seeking restaurant insights and local businesses for strategic planning. The project's significance lies in its novel approach to localized sentiment analysis, providing a unique tool for understanding sentiment patterns within a major city.

Limitations include potential biases in Yelp data and reliance on web scraping for review text. Future work could involve expanding the tool to other major cities, addressing biases in data sources, collecting more data from other sources such as Google reviews, and incorporating more sophisticated sentiment models that include more detailed emotions. In addition, this methodology and dashboard can be extended to look at sentiment patterns in other topics such as hotel reviews, tourist attractions or retail stores. Additionally, ethical considerations, such as user privacy and the impact of online reviews on business success, would require further exploration. Finally, future efforts to refine the dashboard design and explore additional visualizations could enhance the user experience and the tool's overall effectiveness.

As a team, we believe all members have contributed a similar amount of effort.

References

[1] Villena-Román, J., Cobos, A. L., & Cristóbal, J. C. G. (2014, July). TweetAlert: Semantic Analytics in Social Networks for Citizen Opinion Mining in the City of the Future. In UMAP Workshops. https://ceur-ws.org/Vol-1181/pegov2014_paper_01.pdf

[2] Khan, F. H., Bashir, S., & Qamar, U. (2014). TOM: Twitter opinion mining framework using hybrid classification scheme. Decision Support Systems, 57, 245–257. https://doi.org/10.1016/j.dss.2013.09.004

[3] Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C., Ng, A., & Potts, C. (2013). Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank. Stanford. https://nlp.stanford.edu/~socherr/EMNLP2013_RNTN.pdf

[4] Spatial and Temporal Sentiment Analysis of Twitter data from European Handbook of Crowdsourced Geographic Information on JSTOR. (n.d.). https://www.jstor.org/stable/j.ctv3t5r09.20

[5] Serrano-Guerrero, J., Romero, F. P., & Olivas, J. Á. (2020). An OWA and Aspect-based approach applied to Rating Prediction. 2020 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE). https://doi.org/10.1109/fuzz48607.2020.9177614

[6] Liu, Q., Deng, M., Shi, Y., & Wang, J. (2012). A density-based spatial clustering algorithm considering both spatial proximity and attribute similarity. Computers & Geosciences, 46, 296–309. https://doi.org/10.1016/j.cageo.2011.12.017

[7] Ganser, E., Hu, Y., Kobourov, S. (2010). GMap: Visualizing graphs and clusters as maps. University of Arizona. https://www.researchgate.net/publication/224124014_GMap_Visualizing_graphs_and_clusters_as_maps