

Sentiment-Driven Restaurant Recommendations via Chatbot and API Integration

Josh Irwin, Ben Nally, Thao Nguyen, Brett Witzke
May 1, 2024

ABSTRACT

The restaurant industry has seen significant transformations influenced by technological advancements in today's digital age. Previously, restaurants reached their target customers through word-of-mouth, digital advertising, or their online website. However, the introduction of online review platforms, such as Yelp and TripAdvisor, has redefined the process of finding a place to eat for customers. By hosting billions of reviews and ratings cumulatively from millions of monthly users, customers are now equipped with more reliable resources for restaurant recommendations other than relying solely on friends and family recommendations and manual research. The successes of those platforms are well-reflected statistically since as of 2023, Yelp recorded a cumulative 287 million reviews ([Yelp](#)) while TripAdvisor reached over one billion reviews and ratings ([TripAdvisor](#)). However, these positive notions also pose a significant challenge for customers since navigating through all those reviews to reach the final dining decision is not an easy task. Understanding this pain point, our group's solution strives to enhance the restaurant searching process by utilizing API integration and a generative AI model. By performing API calls on those websites to collect relevant information about restaurants that meet customer demands, we feed this data into an AI chatbot for sentiment analysis to generate finalized real-time responses to customers' prompts. After applying our proposed solution, we are successfully streamlining the whole process of manually filtering through thousands of restaurant results on Yelp and TripAdvisor based on users' initial prompts to give a solid recommendation of the best restaurant with desirable ambiance, preferable cuisine, and positive reviews and ratings that highly matches with customer's demand via a human-liked virtual assistant AI chatbot.

INTRODUCTION

Fueled by consumer demand, the restaurant industry saw rapid growth over the years, expecting to reach \$1 trillion in sales by 2024 ([National Restaurant Association](#)) and becoming one of the driving forces behind global consumer spending. Since the success of this industry lies in the ability of the restaurants to provide personalized customer services to satisfy their preferences and satisfaction the industry has to constantly adapt to customer's changing natures due to society and technological advancements.

As a result, the introduction of virtual platforms like Yelp and TripAdvisor, which host millions of restaurant reviews and recommendations from online users, has reshaped how people discover and pick their dining options, serving as the first portals into our dining-out experiences. By making hundreds of millions of customer reviews publicly available on their sites, these websites have transformed online reviews into virtual word-of-mouth, influencing our dining decisions and shaping restaurant reputations. However, filtering through those reviews to find an appropriate eating option can be a significant challenge for customers since nearly half of us

adopt a wait-and-see approach when picking a restaurant. Understanding this issue, we aim to revolutionize the process of restaurant recommendations by employing data analytics and generative artificial intelligence.

The challenge our team is trying to tackle is how to streamline the entire time-consuming and ineffective process of filtering through each restaurant on the recommendation lists of Yelp and TripAdvisor that match customers' preferences to pick the best option into a more analytical process of collecting information about the restaurants that fit with customers' demands and transforming this data into actionable insights that improve the process of restaurant searching. The significance of our project is its ability to address the pain points in the restaurant industry. We understand that with the current traditional approach of restaurant discovery, customers are generally based on stationary platforms or subjective reviews, which potentially expose them to limited or outdated information and biased recommendations. According to TripAdvisor, 94% of American consumers read online reviews before deciding on their restaurants, emphasizing the importance of reliable information for customers in making rational food choices ([TripAdvisor](#)). Another study from Harvard Business School also further defined a direct positive correlation between positive reviews and a business's profit, as it showed that an increase in a restaurant's ratings from Yelp leads to a 5-9% increase in revenue ([HBR](#)).

By understanding the impacts of online reviews, our team decided to adopt a generative AI model solution using API scraping techniques to collect real-time data from the internet to serve as input to an AI-driven recommendation system to solve the current problem. Our final product is the combination of API integration and AI generation models. Beginning with API scraping, we collect a list of restaurants, with relevant information, including types of food, ratings, price, location, and customer reviews that match customers' requests to our initial prompt. Then, we feed this data to our AI chatbot, powered by ChatGPT, to perform sentiment analysis on the reviews extracted from the list above to pick the best restaurant that satisfies customers' needs, such as affordable pricing, high-quality food, and positive reviews and ratings, and deliver a personable human-like recommendation to customers. With the development of the solution, our finalized model was able to achieve the desirable results. From the AI chatbot's prompt "Ask me a food-related question?" and receiving the request from a user, we were able to successfully perform both Yelp and TripAdvisor API calls to find a restaurant that not only satisfied the customer's desired food choice, but also received positive feedback from both Yelp or TripAdvisor with affordable pricing, great ambiance, and high-quality food based on the sentiment analysis that our chatbot performed and give our final recommendation to the customer via a friendly and personable chatbot's response.

DATA COLLECTION & PREPARATION

In order to tackle the restaurant finding and recommendation problem in the Cincinnati area, our team used two data sources (both APIs), ChatGPT, and a smidge of natural language processing. By aligning our collection and preparation goals with phases two and three of the CRISP-DM process (data understanding and data preparation), our data collection & preparation was successful in preparing our data to be used in analysis.

Data Source #1: Yelp API

Our first data source was the Yelp API. Specifically, we used this API in order to retrieve restaurant locations & review information. By passing a search term into the API call (ex: steak), we can search the Yelp database for that keyword to return restaurant locations related to that keyword (ex: steak restaurants), and their corresponding reviews. Strengths of this data source include ease of access and quantity, while weaknesses include quality, recentness, and price. Given that the Yelp API is well documented, it is easy to call and get the data our team was looking for. Additionally, it generates code in multiple languages to make the API call, there are support forums as well as a lot of customizability with the calls in terms of number of items returned and the ability to input multi-word search terms. Yelp has a ton of data available for our team to use. According to Gitnux, “With over 224 million reviews, 5.5 million local businesses claimed on their website, and 28 million monthly average users in the United States alone, it’s clear that Yelp has become a major player in the online review space” ([Gitnux](#)). This conclusion by Gitnux is confirmed by Figure 1 as you can see the number of cumulative reviews has grown steadily (from 2009 - 2022), showing that the Yelp community is not only large, but is still active and growing today.

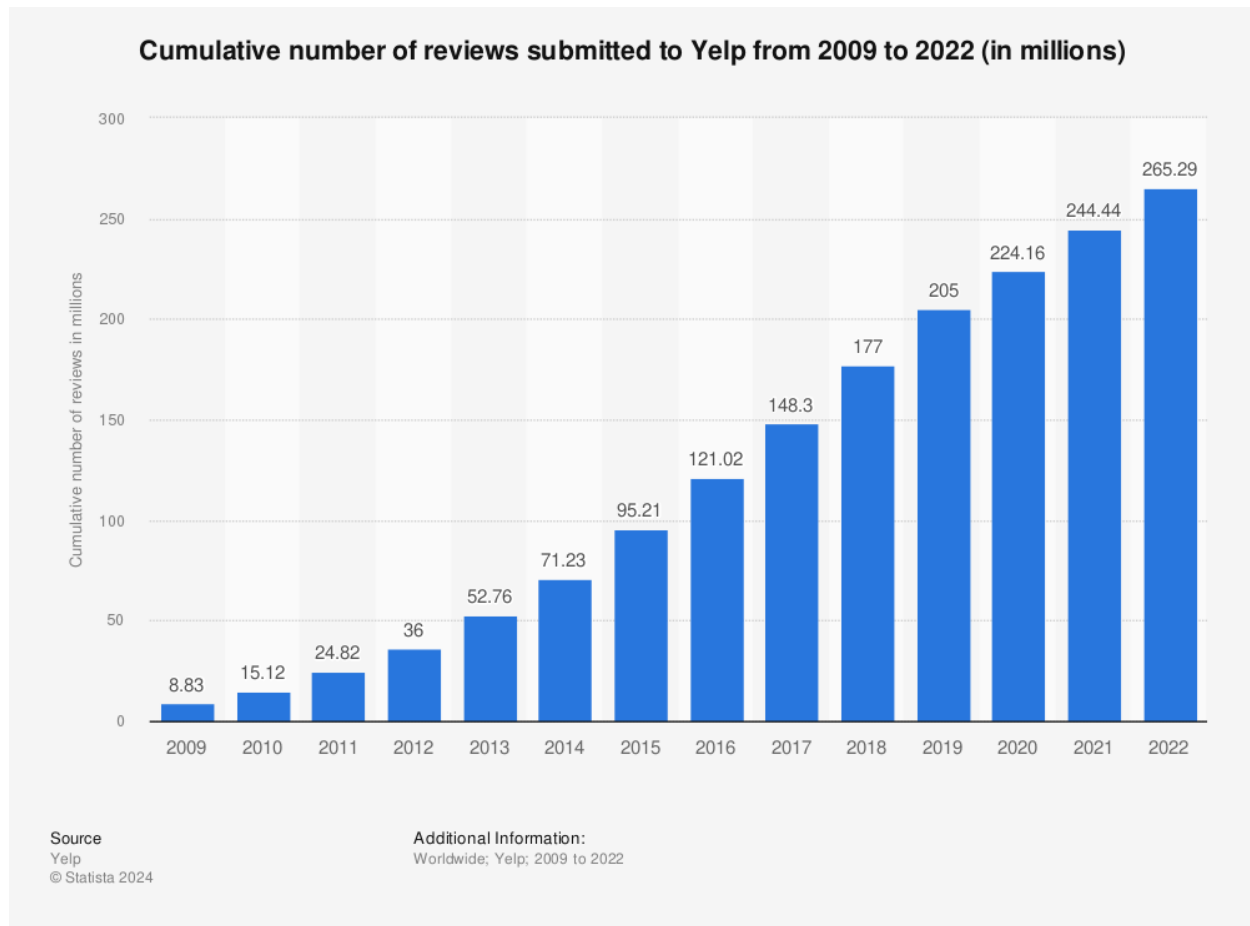


Figure 1 ([Statista](#))

Given the restaurant/user data and consistent growth of reviews, the sheer quantity of available data for use is evident. However, this does not necessarily mean that all 265.29 million reviews on the platform in 2022 ([Statista](#)) are “good” reviews. Some of these reviews may be poor quality, wrong, or fabricated by bots. However, Yelp has moderators that review the platform's reviews, so our team decided to rely on Yelp themselves to provide solid data that we will use as inputs to our model. Another concern with the quality of Yelp reviews is how many five-star ratings have been given. According to Yelp, of the platform's 287 million cumulative reviews, 53% of those (152.1 million) are five-star ratings ([Yelp](#)). This means that over half of the reviews are five stars, and there are many reasons this could be the case. A potential cause could be the willingness by a consumer to report positive experiences over negative experiences, but there is research to support this explanation as well as its opposite. It is also evident from Figure 1 that the platform has reviews that date back to at least 2009. This means, the API has the potential to pull reviews written long ago that may not reflect the current status or quality of a restaurant. However, according to the API documentation, Yelp will pull the newest reviews first, so as long as there are recent reviews, those will be primarily pulled ([Yelp Developer](#)). Finally, the Yelp API is only free to an extent. Our team was able to use the Yelp API for free by

only making a few calls per day. If our solution was to be deployed, the costs of using the API could skyrocket.

Data Source #2: TripAdvisor API

Our second data source was the TripAdvisor API. As with the Yelp API, we pulled restaurant location and review data from TripAdvisor to pass to ChatGPT for analysis. The same search term is used to search for restaurants in the Cincinnati area related to that term, and review data is pulled for top search results. Using two review data APIs has the potential to pull the same restaurants multiple times, but this is not necessarily a bad thing; It can provide more review data on one location helping ChatGPT make a more informed recommendation. However, our team with a handful of manual searches (ex: steak) tested API calls to see if the same restaurants would appear in the returned data but this did not happen. This allows for a wide variety of restaurants and reviews to be returned to ChatGPT so it can make a recommendation to the user. Strengths of this data source include ease of access and quantity, while the weaknesses remain recentness, pricing, and additionally API call customizability. The TripAdvisor API is well documented and makes constructing an API call URL easy. As seen below in Figure 2, TripAdvisor has over a billion reviews, for 5.2 million restaurants, or almost 200 reviews per restaurant ([Statista](#)). The same problem with the recentness of reviews occurs with TripAdvisor, as evidenced by Figure 2, meaning the reviews pulled by the API have the potential to be from 2014, or even earlier, but the TripAdvisor API also pulls the most recent reviews first so that was not a primary concern for our team.

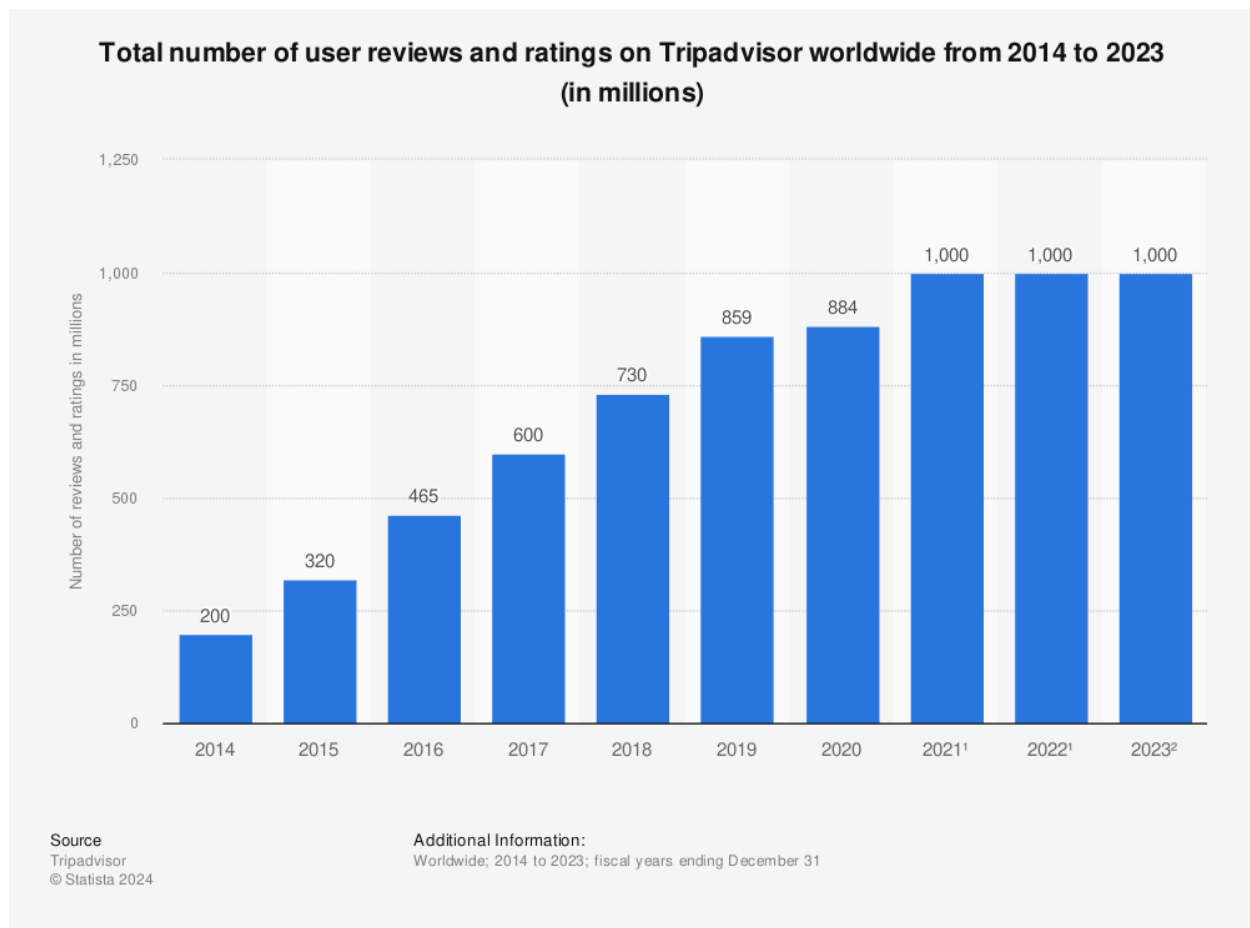


Figure 2 ([Statista](#))

The same pricing problem with the Yelp API appears with the TripAdvisor API, and scalability in terms of price to call the API remains the issue, not scalability with the amount of calls being made. The TripAdvisor API is incredibly easy to use and provides code in multiple languages for performing the requests similar to Yelp. However, it lacks customizability. TripAdvisor is very limited when customizing the API call (number of responses returned, search term mechanics, etc.), which could lead to deployment issues.

CRISP-DM Process

As mentioned earlier, our data collection and preparation process goals were derived from stages two and three of the CRISP-DM process. There was not much our team had to do for the data collection phase. Given that we are getting our raw data in real-time via API requests, there is not much data collection done by us before running our code. The data has been structured and cleaned for us by Yelp and TripAdvisor and this is the data we are pulling from when using their respective APIs. However, after the APIs have been called and returned data the data is cleaned to only return the necessary data that will be sent to ChatGPT, and this will be discussed further

in the analysis section. In essence, we simply pull out relevant location IDs (for the TripAdvisor API) to search for reviews on that specific location. Then, the raw text of the returned reviews are pulled out and sent to ChatGPT for sentiment analysis and final recommendation. Given that the APIs will handle the issue of missing values, mixed data types, and help avoid any copyright issues, our team made the data collection and preparation processes much simpler since we are not using any data not from an API. This “cleaned” data is passed to ChatGPT for analysis, which will be discussed later on.

Technical Requirements

Technical requirements one, two, three, and four are satisfied by this part of the project. Our team uses two different data sets, at least one of which is an API. Neither the Yelp or TripAdvisor were demonstrated in class (although I believe the Yelp API was referenced). Finally, there were no artificial data sets created by an LLM used in our project, given that we pull from our data sets via API.

DATA ANALYSIS & EVALUATION

Our program first defines several functions; `reduce_to_search_term`, `tripAdvisorAPISearch`, `tripAdvisorAPIReviews`, `yelp_request`, `yelp_search`, and `ask_question_openai_api`. The program combines and uses all of these functions during the operation of the main function to answer any question a user may have regarding restaurants in a given area. For our proof of concept, we have limited the location to just Cincinnati.

The `reduce_to_search_term`` function takes a question as input and aims to reduce it to a single search term, which is meant to capture the main topic or intent of the question. Before the function begins, the program imports necessary modules from the Natural Language Toolkit (NLTK) library: ``nlk``, ``stopwords``, and ``word_tokenize``. These modules are used for natural language processing tasks such as tokenization and removing stopwords. It also downloads NLTK resources related to tokenization (``punkt``).

Once the `'reduce_to_search_term'` function is called—taking a question as an argument—the question is tokenized into individual words using ``word_tokenize``. Stopwords (common words like "the", "is", "and", etc., which typically don't carry much meaning) are removed from the tokenized question. The program also defines a list of common food and cuisine. This serves as a reference for identifying relevant keywords in the question. The program calculates the term frequencies for the remaining words in the question. It assigns a higher weight to words that are found in the list of food and cuisine words. The most frequent term (word) is then selected as the search term. If there are multiple words with the same frequency, the last one encountered will

be chosen as the search term. This is due to the last word of question generally being the object of a sentence (Ellis). The search term is then returned.

The program interacts with both APIs using two functions each. For TripAdvisor, `tripAdvisorAPISearch` and `tripAdvisorAPIReviews` are used. Before these functions are called, we store our API key in the variable `tripAPI_JOSH`. When this software is pushed to the public, this key will be removed and stored as a 'secret'. Our interaction with the TripAdvisor API begins when `tripAdvisorAPISearch` is called. This function performs a search for locations using the TripAdvisor API based on a given search term. It takes an API key and a search term as input parameters. It constructs a URL for the API call using the provided API key and search term, along with other parameters like category, latitude and longitude, radius, and language. The function then makes a GET request to the constructed URL using the `requests.get` function. After which it extracts the relevant location IDs from the JSON response and returns them as a list. Our program then calls the `tripAdvisorAPIReviews` function. This function retrieves reviews for a specific location based on its location ID. It takes an API key and a location ID as input parameters. It constructs a URL for the API call using our API key and the location ID that was extracted when `tripAdvisorAPISearch` was called. It makes a GET request to the constructed URL using the `requests.get` function and extracts the raw text of the reviews from the JSON response, returning them as a list.

Our program interacts with the Yelp API using the `yelp_search` and `yelp_request` function. The interaction with the Yelp API begins when `yelp_search` is called. The `yelp_search` function takes three parameters: Our Yelp API key for authentication, The search term or keyword the program derived earlier, and the location where the user would want to perform the search (we have this set to Cincinnati by default for testing purposes). Inside the function, it initializes a dictionary named `url_params` and populates this dictionary with three key-value pairs using the arguments it took for the API key, search term, and location. The function then calls `yelp_request` with the following arguments: The base URL of the Yelp API, The specific endpoint or route for searching on the Yelp API, our Yelp API key and the `url_params` dictionary it defines. The `yelp_request` function then returns the JSON response containing the search results from the Yelp API, which the `yelp_search` function then takes and returns to where it was called.

After collecting the data from both TripAdvisor and Yelp, the program then interacts with ChatGPT using the `ask_question_openai_api` function. This function takes two arguments: question (the question to be answered) and context (the JSON data). The function initializes an OpenAI API client with the provided API key. It makes a request to the OpenAI API's `chat.completions.create` endpoint and specifies the model as `gpt-3.5-turbo`. The message to be sent is then compiled using the questions and JSON data taken as arguments. The function then

returns the generated response content, which is the AI's answer to the question using the given JSON context. This response is then printed to the user.

Here are some sample interactions:

1.

Ask me a restaurant-related question: What are some good pizza places?

Found restaurants on TripAdvisor

Found restaurants on Yelp

Answer: To my knowledge, some good pizza places are Fireside Pizza, St. Francis Apizza, Your Mom's Pizzeria, Adriatico's, and Il Gusto Pizza & Pasta. Fireside Pizza in Cincinnati offers a variety of pizza options along with a beer bar and cocktail bars. St. Francis Apizza also in Cincinnati is highly rated and serves delicious pizzas and salads. Your Mom's Pizzeria is another great spot in Cincinnati known for its exceptional pizza. Adriatico's is a popular choice serving pizza and hosting sports bars in Cincinnati. Il Gusto Pizza & Pasta is another top choice that offers a mix of pizza and Italian dishes like pasta. Each of these places has unique offerings and great reviews, making them good choices for pizza lovers.

2.

Ask me a restaurant-related question: are there any good steak places

Found restaurants on TripAdvisor

Error: 403 Client Error: Forbidden for url:

<https://api.yelp.com/v3/businesses/search?term=steak&location=Cincinnati&limit=3>

No restaurants found on Yelp.

Answer: To my knowledge, based on the reviews provided for restaurants on TripAdvisor, there are several highly recommended steak places listed in the context. Some notable ones include Steak Aoyama in Kobe, Bern's Steak House in Tampa, Peter Luger Steak House in Brooklyn, TDQ Steaks in London, Steak44 in Phoenix, The Steak House at Circus Circus in Las Vegas, Joe's Stone Crab in Las Vegas, and a beach food shack with steaks in an unnamed location. These restaurants have received positive feedback for their food quality, service, ambiance, and overall dining experience.

Steak Aoyama in Kobe stands out for its excellent steak offerings, friendly atmosphere, and welcoming service. Guests have praised the chef's skills, delicious meals, and engaging interactions.

Bern's Steak House in Tampa has received mixed reviews, with some diners having a fantastic dining experience, while others were disappointed. The restaurant's ambiance, service, and food quality were highlighted by some customers, although others found the steaks to be bland and overpriced.

Peter Luger Steak House in Brooklyn has been consistently praised for its outstanding porterhouse steak, flavorful sides, and commitment to quality. Guests have appreciated the attentive service and overall dining experience at this historic establishment.

TDQ Steaks in London impressed diners with its unique offerings of Canadian bison and Scottish beef, along with exceptional service from a knowledgeable hostess/waitress.

Overall, based on the reviews provided, there are several good steak places recommended for different reasons, such as food quality, service, and overall experience. It would be worth considering these restaurants if you are looking for a memorable steak dining experience.
Ask me a restaurant-related question: exit
Exiting...

The program reliably provides accurate restaurant information about the restaurants it suggests, however it doesn't seem to narrow its search to just the Cincinnati test case. Its error handling also effectively manages the frequent disruptions caused by the Yelp API. Though interestingly, after testing with a get request in cURL, the YelpAPI is confirmed to be currently working but the program is not able to get a response. After peer reviewing the code, utilizing ChatGPT 3.5, ChatGPT 4, and Gemini 1.5 Pro, no one is able to identify why the program is unable to get a response from YelpAPI. Additionally, the program currently only focuses on straightforward question-answer exchanges, lacking in-depth conversational capabilities. In the future, developing the program to facilitate longer, more varied dialogues could broaden its utility and appeal to users.

CONCLUSION

In conclusion, we believe our research presents both comprehensive and tailored solutions to the traditional challenges of navigating the landscape of restaurants, a process that has only been complicated in the digital age. By integrating API calls from both Yelp and TripAdvisor with a chatbot powered by ChatGPT, we believe we've developed a sentiment-driven restaurant recommendation system that can streamline the process of finding the perfect place to eat while taking all your personal preferences into consideration.

With the utilization of our different API scraping techniques and natural language processing, we were able to develop a model that efficiently collects real-time data on restaurants including cuisine types, ratings, pricing, location, and customer reviews, and performing sentiment analysis to understand the public opinion and begin to quantify these reviews. This data is then synthesized by our chatbot which in turn provides users with a personalized recommendation based on their individual query. We believe our model is able to effectively address the issue of the overwhelming amount of information users find on the internet when looking through reviews, and is also able to ensure the delivery of up-to-date, and relevant recommendations through the use of API calls and ChatGPT.

We understand that our model is not at all perfect. If we had more time to work on our model, or make changes in the future, we would have a few ideas on how we can further improve our model. One of the ways we envision making improvements could be furthering the information provided by the chatbot, to include things like including links or pictures of menus or other items. We could also work to incorporate more data from sources such as social media pages, or

using map data to show precise locations of restaurants. We can also include things like dietary restrictions, and additional information to make our model even more personalized. We understand incorporating these things will take time and a lot of attention specifically to the data preparation and modeling states. Incorporating additional data sources will take time to properly clean and process the data for effective use. In addition to this, implementing these additional sources into the model takes time and expertise as well. Overall, we think our model is a good starting point, but by incorporating these strategies, we can create a more robust and user-centric restaurant recommendation model that caters to a wider range of preferences and delivers even more accurate and relevant suggestions to users.

Looking forward, as we think about potential deployments of this model, we think it has the power to forever enhance customers' experience when looking for a new restaurant, while in turn benefitting restaurant owners who are consistently receiving positive reviews. As we've discussed earlier in the report, there is a direct correlation between positive reviews and business revenue for restaurants, and our model could help lead to increased foot traffic and revenue for restaurants with consistently positive reviews. One potential deployment use we see could involve integrating this chatbot into popular restaurant review apps, such as Yelp or TripAdvisor to provide users with seamless access to the personalized recommendations made by our model regardless of where they are. Additionally, further refinement and optimization of our model could enhance the accuracy and effectiveness of the recommendations provided, ensuring continued satisfaction and usability. Overall, our research and model we believe lays the groundwork to revolutionize the restaurant search process, offering a glimpse into the future of personalized dining experiences.

References

Lindner, Jannik. "The Most Surprising Yelp Statistics and Trends in 2024 • Gitnux." *GITNUX*, 24 Apr. 2024, gitnux.org/yelp-statistics/#:~:text=Yelp%20is%20an%20incredibly%20popular,in%20the%20online%20review%20space.

(Figure 1)

Dixon, Stacy Jo. "Yelp: Cumulative Number of Reviews 2022." *Statista*, 15 Dec. 2023, www.statista.com/statistics/278032/cumulative-number-of-reviews-submitted-to-yelp/.

(Figure 2)

Published by Statista Research Department, and Feb 20. "Tripadvisor: Number of Reviews 2023." *Statista*, 20 Feb. 2024, www.statista.com/statistics/684862/tripadvisor-number-of-reviews/.

Ellis, Matt. "Sentence Structure: Learn the Rules for Every Sentence Type." *Sentence Structure: Definition and Examples | Grammarly Blog*, 29 June 2023, www.grammarly.com/blog/sentence-structure/.

Influences on Diner Decision-Making | Tripadvisor for Restaurants, 4 June 2018, www.tripadvisor.com/ForRestaurants/r3227.

Luca, Michael. "Reviews, reputation, and revenue: The case of yelp.com." *SSRN Electronic Journal*, 16 Sept. 2011, <https://doi.org/10.2139/ssrn.1928601>.

"2024 State of the Restaurant Industry ." *National Restaurant Association*, 5 Feb. 2024, restaurant.org/research-and-media/research/research-reports/state-of-the-industry/.

"Yelp - Investments in Product-Led Strategy Drove Yelp's Strong 2023 Results." *Yelp Newsroom*, 15 Feb. 2024, www.yelp-press.com/press-releases/press-release-details/2024/Investments-in-Product-Led-Strategy-Drove-Yelps-Strong-2023-Results/.

Tripadvisor. "Total Number of User Reviews and Ratings on Tripadvisor Worldwide from 2014 to 2023 (in Millions)." *Statista, Statista Inc.*, 16 Feb 2024, <https://www.statista.com/statistics/684862/tripadvisor-number-of-reviews/>

[Yelp](#)

[Yelp Developer](#)

[Statista](#)