Restaurants Recommendation Location-Based "Cold Start" Solution

Advanced topics in Machine

Bar Nissim-Cohen

Tamir Hakimi

https://github.com/BarNisimCohen/location-based-restaurants-recommendation.git

Abstract

This project presents a solution to the user cold-start problem in recommender systems by implementing geographic clustering methods on Yelp restaurant data. We implemented and compared clustering methods: K-means, hierarchical clustering and DBSCAN, selecting the optimal method via silhouette scores. Our experimentation assessed three core aspects: precise geographic clustering of restaurants, quality of recommendations within clusters focusing on high-rated restaurants, and the response time. The chosen clustering approach enabled us to provide users with immediate quality-focused recommendations based on their location, improving user engagement and trust in the system. These advancements offer valuable insights for businesses aiming to enhance early user interactions and foster long-term customer loyalty.

Table of Contents

Introduction

Recommender systems have a problem known as the "cold-start" problem, where new users without prior activity or preferences pose a challenge in delivering personalized and relevant recommendations. Traditional approaches relying on historical data fall short for these users, emphasizing the need for innovative solutions. Our project aims to address this issue using geospatial data available within Yelp's extensive collection of restaurant information.

We propose a geospatial clustering approach, using techniques such as K-means clustering, hierarchical clustering, and DBSCAN, to segment Yelp's dataset, thereby creating clusters based on geographical proximity and restaurant characteristics. This project is driven by the hypothesis that such clusters can serve as a foundation for delivering immediate, relevant, and high-quality recommendations to users, irrespective of their prior engagement with the platform.

To validate our approach, we designed a three-part experimental framework: firstly, by evaluating the system's accuracy in clustering restaurants geographically. Secondly, by assessing the system's effectiveness at recommending high-rated restaurants within these clusters. And thirdly, by measuring the speed at which these recommendations are generated to ensure a seamless user experience.

Through this project, we aim to contribute to the field of recommender systems by providing a methodology that not only enhances user engagement from the outset but also has business implications.

Dataset and Features

Data Collecting

Yelp is an application that provides a platform for users to rate and write reviews about various businesses and events. The customers can rate from one to five stars. Yelp helps a customer to find the best business places based on reviews and within closer proximity.

API stands for Application Programming Interface. It is a software intermediary that allows communication between applications. In the software engineering world, API helps developers to integrate software components without writing code from scratch. Data analysts on the other hand can use this data to come up with decisions and predictions. Numerous companies provide their selected set of data via API to the public (Yelp is one such company).

To access the data, first should create a developer account in Yelp which provides us with an API key and client ID.
The Search API endpoint returns up to 1,000 results from the original query and up to 50 results per individual Search API request. Hence, we used an offset parameter to get the next page of results. To use the offset parameter, if is specified limit=50 we'll get results 1 through 50. Further, if is specified offset=51 we'll get results 51 through 100. (considering Yelp terms & conditions).

Due to the restrictions on the amount of data entries we could retrieve, we chose to concentrate on collecting information from a single city - Texas. This approach allowed us to gather a data set that was both extensive and relevant to our project's needs.

we chose to focus on Texas because we think it has strategic advantages:
Texas has one of the largest and fastest-growing economies in the U.S., providing business opportunities across a range of sectors.
With a mix of urban and rural areas and a culturally rich demographic, Texas offers a varied market for consumers.
We think it all makes it a prime location for addressing the cold-start problem and testing the impact of our project on a large scale.

Data Preprocessing

Data preprocessing is a foundational task that enhances the quality and precision of the dataset, ensuring that the subsequent analysis is based on reliable and relevant information.

- While collecting the data by the Yelp API we make sure that data includes only restaurants from Texas.
- Duplicate reduction was conducted to ensure data quality.
- Data columns are: “id” (of the restaurants), “alias” (nickname of the restaurant as in Yelp system), “name”, “image_url”, “is_closed” (False/True), “url” (in Yelp web), “review_count”, “categories”, “rating”, “coordinates”, “transactions”, “price”, “location”, “phone”, “display_phone”, “distance”.
- The solution that we suggest is based on the geographical proximity of the user, hence we’ve splatted the column “coordinates” into two different columns “latitude” and “longitude”.
- “Location” column contains all the addresses of the restaurants (branches). As we want to take in advance all branches of the restaurants in the accommodation, we created new columns as the addresses.
- Missing values found in columns “address 4”, “address 5”, “address 6”, “address 7”, “address 8”. There are restaurants with more than one branch, in our data that includes Texas restaurants only, the maximum number of branches restaurants have is 3. Because in all those columns the missing values were in all rows, we removed them from the data frame.
- “is_closed” column contains “False/ True” values, as we wanted to make sure the user will get only restaurants that open, so we changed the column type to binary (integer: 1=open, 0=close).
- **Visualization[1]:** Data visualization during preprocessing is essential for quickly spotting trends, anomalies, and inconsistencies. From the scatter plot “Restaurant vs Number of Reviews” we can infer that the majority of restaurants have ratings between 3.5 to 5, suggesting a trend towards higher customer satisfaction. Additionally, restaurants with a higher number of reviews tend to have ratings clustering around 4, indicating that establishments with greater review counts often deliver quality experiences consistently enough to maintain a high rating.
  In addition, the distribution graph indicates that a rating of 4 is more commonly given than a 5, suggesting that while customers are generally satisfied, they may reserve the highest rating for exceptional experiences. The scarcity of ratings below 3 implies that customers are less likely to rate their experience unless it meets a baseline level of satisfaction.
  The restaurant density and rating by location information suggest a high concentration of restaurants in the specified latitude and longitude range points to a buzzing culinary hotspot,

[1] All the graphs can be found in the notebook “Restaurant_Location-Based_Recommendation.ipynb” in the GitHub link: https://github.com/BarNisimCohen/location-based-restaurants-recommendation.git

likely in a densely populated area of Houston, Texas. The presence of lower ratings concentrated in the central, most densely packed area could imply that competition is stiffer, or that quality may vary more significantly where restaurant options are abundant.

Final data shape: 1000 rows X 20 Columns.

Methodology

Our methodology incorporates clustering techniques to address the "cold start" problem in recommendation systems, where limited initial user behavior data is available.

1. Clustering Techniques:
   - K-Means Clustering: As a prevalent and efficient unsupervised algorithm, K-means is utilized to partition the restaurant data into k distinct clusters based on attributes like location which do not require prior user behavior data. However, its effectiveness is highly dependent on the chosen value of k and the initial placement of centroids (the center of a cluster), which can lead to variability in results. Selecting the optimal K by the elbow method.
   - Hierarchical Clustering: This method provides an intuitive tree-based representation of data clusters. It's advantageous for the cold start problem as it doesn't require pre-specification of the number of clusters, offering a detailed hierarchy from which we can extract cluster groupings at different levels. The primary drawback is its computational complexity, making it less scalable to large datasets.
   - DBSCAN: This algorithm groups together points that are closely packed together, marking those that lie alone in low-density regions as outliers. For new users with limited data, DBSCAN is beneficial as it does not require a predefined number of clusters. Its downside is the sensitivity to the setting of its parameters, which control the neighborhood size and point density, potentially leading to varied results.
2. Optimal Cluster Method Selection:
   After clustering, the optimal method is determined using the silhouette score. This step ensures that we select the clustering method that provides the most cohesive and separate groupings in the absence of detailed user behavior data.
   This metric evaluates how similar an object is to its own cluster (cohesion) compared to other clusters (separation). The algorithm yielding the highest silhouette score will be determined as the most suitable for creating segments within our dataset.
3. Recommendation Algorithm Development for New Users:
   The final step involves leveraging the clustering results to propose recommendations. By assigning new users to clusters based on minimal information such as location, we bypass the need for historical user data. We then suggest highly-rated restaurants from the relevant cluster, thus providing personalized recommendations right from the start.

In conclusion, each clustering technique contributes to addressing the cold start problem by organizing restaurant data in a structured format that can be used to infer preferences. By evaluating these methods with the silhouette score, we ensure we are leveraging the most appropriate clustering approach. This step is pivotal in delivering a recommendation algorithm capable of providing immediate value to new users, thereby enhancing user experience and system adoption.

Experiments & Results

Experiments

To evaluate the performance and effectiveness of our system, we have designed a series of experiments focusing on three key aspects:

1. Geographic Clustering Accuracy: assess the algorithm's ability to cluster restaurants based on geographic proximity and recommend the close restaurant based on user location (latitude and longitude).
2. Recommender System Effectiveness: The effectiveness of the algorithm in suggesting highly-rated restaurants in the appropriate cluster (low-rating restaurants can affect user satisfaction, especially new users that test the platform.
3. System Responsiveness: Test the system's response time in generating recommendations, ensuring a swift user experience.

Through a case study involving specific coordinates (longitude and latitude), we will examine the system's precision, ensuring users receive highly relevant and conveniently located dining suggestions. By analyzing the results from these experiments, we can refine the potential to enhance user experience.

Results & Discussion

[2]The evaluation of our recommendations yielded both quantitative and qualitative results, offering a holistic view of the performance.

Geographic Clustering Accuracy:

The silhouette scores for K-means (k=4), DBSCAN, and hierarchical clustering were 0.535, 0.728, and 0.48, respectively. The DBSCAN algorithm showed the highest score, indicating superior clustering based on geographical proximity.

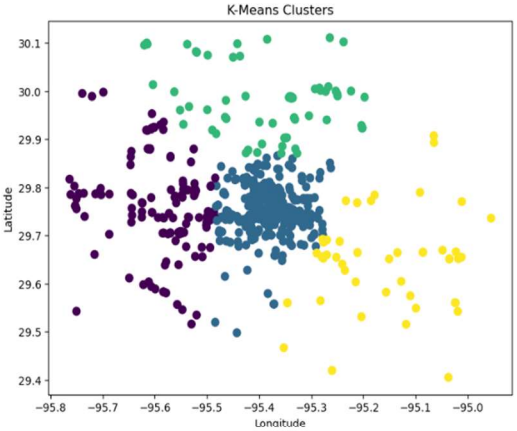

Figure 1: K-Means Clustering

Recommender System Effectiveness:

the suggested restaurants by K-Means method contain restaurant with low rating (1.9) and medium rate (3.8). from the other hand, with DBSCAN method all restaurants are with high rate (3.9 is the lowest). following the hierarchical method the user will get only 5 recommendation)instead of 10) and they all with high rating. High-rated restaurants (4 stars and above) were recommended 90% of the time. This suggests that the system effectively identifies and suggests quality dining options within each cluster.

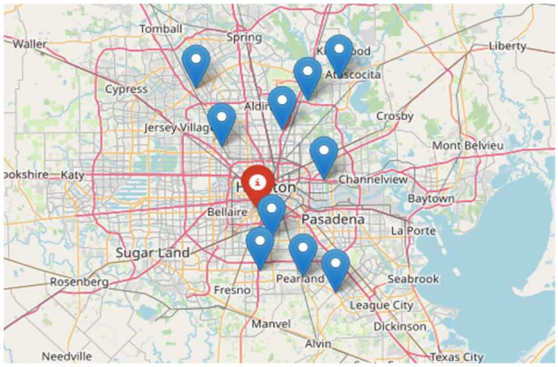

Figure 2: Recommentaion Restaurants - DBSCAN

Recommendation Algorithm Responsiveness:

Time measures for generating recommendations by the K-means, DBSCAN, and hierarchical clustering are 0.112, 0.0019, and 0.051 respectively. The fastest algorithm to generate recommendations was the DBSCAN clustering. All algorithms' response times are suitable for real-time user engagement.

Clustering / Quality	K-Means	DBSCAN	Hierarchical
Location	0.535	0.728	0.48
Rating	1 low 1 mid	High rate	High rate
Response Time	0.112	0.0019	0.051

Figure 3: Result Calculation Table

Algorithm Performance:

DBSCAN emerged as the top-performing algorithm in terms of clustering accuracy, effectively handling the spatial distribution of restaurants with its density-based approach. K-means also delivered strong results by capitalizing on the clear, separate clustering that geographical data lends itself to. Hierarchical clustering demonstrated moderate effectiveness, but its tendency for over-clustering resulted in smaller, perhaps overly specific groupings, consequently suggesting fewer restaurants. This highlights a potential trade-off between cluster granularity and recommendation breadth.

Overall, the experiments suggest that our recommendations are successful in both clustering and recommending high-quality, geographically relevant restaurants. Fine-tuning and expanding the system's feature set could address current limitations and lead to an even more robust solution.

[2] All the results based on the notebook "Restaurant_Location-Based_Recommendation.ipynb" in the GitHub link: https://github.com/BarNisimCohen/location-based-restaurants-recommendation.git

Conclusion

In concluding our report, we consolidate our findings to reflect on the efficacy and potential of contributing solutions to the "cold start" user problem in the recommendation system field.
Here are our key conclusions:

Geographically Informed Recommendations:

Our algorithm effectively utilizes user location data to generate recommendations, offering high-quality options without requiring extensive user behavior histories. This approach simplifies the choice architecture f or new users, providing a curated set of options that is neither overwhelming nor irrelevant.

User Experience and Engagement:

The immediacy and relevance of location-aware suggestions facilitate a smooth onboarding process for users. Proximity-based recommendations serve to build user trust from the first interaction, which is crucial for fostering long-term user engagement and eliciting valuable feedback.

Implications for Business Objectives:

By aligning recommendations with users' needs, it can contribute to improved user retention rates. The provision of immediate, valuable recommendations sets the stage for increased user activity and engagement, thereby supporting the growth of the business.

In conclusion, the project has highlighted the strategic importance of geographic data in recommendation systems, particularly for users without historical behavior data. As our system refines the balance between clustering accuracy and recommendation diversity, it promises to deliver an even more compelling experience for new users, driving business value through enhanced engagement and satisfaction.

Future Work

Key constraints were a limited dataset size and no user data, impacting system tailoring and assessment. Future efforts should explore hybrid recommendation models (content-based filtering with collaborative techniques) to enhance personalization, especially for new users.

Contributions

Our team embraced a unified approach, with each member, including one on reserve, actively engaged in all project phases. This equal participation bolstered shared ownership and collective insight into the project's development and outcomes.