Restaurant Recommendation System Using Knn Classifier Based on Content-Based Filtering

Manul Chourasia[1], Sakshi Gautam[2], Ankit Bagri[3] and Satyendra Ahirwar[4]
Department of Computer Science and Engineering, Samrat Ashok Technological Institute , Vidisha ,Madhya Pradesh, India
Email: manul24cs062@satiengg.in[1],

***Abstract* - In the burgeoning domain of recommendation systems, this research endeavours to develop a robust restaurant recommendation system tailored to individual user preferences. Leveraging data preprocessing techniques, the system effectively filters and cleanses input data, ensuring relevance and accuracy. Upon prompting the user to select a location of interest, the system dynamically curates a dataset specific to the chosen city, subsequently refining it by eliminating extraneous columns. The categorical attributes are meticulously extracted and presented to the user for selection, facilitating a personalized filtering mechanism. Further enriching the user experience, the system incorporates pricing considerations, prompting the user to input a preferred price range for dining. Leveraging this input, along with past user preferences and location-based information, the system applies a K-Nearest Neighbors (KNN) classifier to generate refined restaurant recommendations. This model, trained on historical data, effectively discerns user preferences, and offers tailored suggestions, enhancing user satisfaction and choice convenience.**

Keywords: **recommendation, preference, suggestion, restaurant**

I. INTRODUCTION

In the digital age, navigating the plethora of dining options available can be overwhelming for consumers seeking personalized experiences. Recommender systems have emerged as a solution to this challenge, offering tailored suggestions based on individual preferences. Within this landscape, content-based filtering stands out as a powerful technique for delivering personalized restaurant recommendations[1].

This paper introduces a novel content-based filtering approach to restaurant recommendation, leveraging a K-Nearest Neighbors (KNN) model. Unlike traditional collaborative filtering methods, which rely on user similarities, content-based filtering focuses on the attributes of the items themselves. In the context of restaurant recommendations, this means considering factors such as cuisine type, restaurant ratings, location, and price range.

By harnessing the power of content-based filtering, the proposed system aims to provide users with highly relevant dining suggestions that align with their unique preferences. Through the utilization of a KNN model, the system can effectively identify restaurants that closely match the user's desired criteria, resulting in a more personalized and satisfying dining experience.

This paper seeks to showcase the efficacy of content-based filtering in enhancing restaurant recommendations, demonstrating its ability to deliver tailored suggestions based on individual preferences. By harnessing the rich attributes of dining establishments, the proposed approach offers users a seamless and personalized dining discovery process.

II. LITERATURE REVIEW AND METHODOLOGY

A. Literature Review

The Yelp Food Recommendation System employs diverse principles and methodologies to create predictive models based on users' restaurant reviews and ratings. Leveraging available datasets, the system extracts user preference features and utilizes a combination of collaborative and content-based filtering algorithms[2]. Additionally, the system integrates clustering techniques such as K-nearest neighbors and weighted bipartite graph projection, alongside various other learning algorithms to enhance recommendation accuracy.

Likewise, the Preference-based Restaurant Recommendation System caters to both individuals and groups by tailoring recommendations to their specific preferences. For a wide user base, the system employs a ranking Support Vector Machine (SVM) model, integrating features like users' food preferences, dietary restrictions, cuisine type, available services, ambience, noise level, and average rating. Additionally, the system prioritizes maximizing overall satisfaction within user groups, diverging from conventional group recommendation systems that simply select the most commonly recommended restaurant among individual users.

Similarly, Feature Selection Methods for Text Classification employs an unsupervised approach to select features that effectively generalize textual data, thereby transforming it into valuable information by categorizing text into classes. By incorporating these features, the system notably enhances the accuracy of classification tasks compared to when features are not utilized. Key feature selection strategies include subspace sampling, uniform sampling, document frequency, and information gain.

B. Collaborative Filtering

Collaborative filtering is a method used to predict what a user might be interested in by collecting preferences from many users. The idea behind it is that if one user shares similar tastes with another on certain topics, they're likely to have similar preferences on other topics too. This concept comes from the belief that people often get the best recommendations from others who have similar tastes. So, collaborative filtering focuses on finding and matching users with similar interests to make recommendations[1].

Collaborative filtering algorithms typically rely on three key components:

1. Active involvement from users,
2. A user-friendly method for expressing users' preferences to the system, and
3. Advanced algorithms capable of identifying individuals with similar interests.

In the typical workflow of a collaborative filtering system:

1. Users provide their preferences by rating various items like books, movies, or CDs within the system. These ratings serve as an approximation of the user's preferences in that specific domain.
2. The system compares these user ratings with those of other users to pinpoint individuals with the most similar preferences.
3. Using the preferences of similar users as a guide, the system suggests items highly rated by these users but not yet rated by the current user, assuming the absence of a rating indicates unfamiliarity with the item.

A key challenge of collaborative filtering lies in determining how to aggregate and prioritize the preferences of a user's peers. Over time, as users rate recommended items, the system improves its understanding of their preferences. However, collaborative filtering may struggle to precisely match content with a user's preferences, particularly in communities dominated by a single viewpoint. Additionally, the introduction of new users or items can present a "cold start" problem, as insufficient data may hinder accurate recommendations. To mitigate this issue, the system must initially analyze past voting or rating behaviours to learn the preferences of new users. Furthermore, a significant number of user ratings are typically necessary for a new item before it can be recommended effectively by the collaborative filtering system.

C. Content - based Filtering

Content-based filtering (CDF) recommends products based on their similarities, following the principle of "Show me more of what I have liked". It involves creating item profiles by extracting features from items and user profiles from the features of items purchased by users. Similarity scores between user profiles and item profiles are then calculated to recommend items with the highest similarity scores. This method is commonly used to recommend documents like news, websites, movies, and books based on keywords from profiles. It offers personalized recommendations and transparency in its workings. However, it may struggle with recommending items similar to those already purchased by users and could face challenges in generating attributes for items in certain areas[7]. One example of CDF is the Content-Based Citation Recommendation Model, which consists of two stages: a fast, recall-oriented candidate selection and a feature-rich, precision-oriented reranking. This model utilizes a supervised neural model to embed all available documents in candidate selection (NNSelect) and employs a three-layered feed-forward neural network with two ELUs and a sigmoid layer for Reranking Candidates (NNRank). This approach has enabled the development of a citation recommendation system without relying on metadata available in baseline methods.

D. K-nearest neighbours (KNN)

K-nearest neighbors (KNN) is a simple, instance-based learning algorithm used for classification and regression. In the context of classification, KNN is a non-parametric method that classifies a new data point based on the majority class among its K nearest neighbors. Here's how the KNN classifier works[10]:

1. **Training Phase**: In the training phase, the algorithm simply memorizes the feature vectors and corresponding class labels of the training data. No explicit model is built.
2. **Prediction Phase**: When a new data point is to be classified, the algorithm calculates the distance between the new data point and all the training data points. The most common distance metrics used are Euclidean distance, Manhattan distance, or Minkowski distance.
3. **Finding Neighbors**: The algorithm then selects the K nearest neighbors (data points with the smallest distances) to the new data point.
4. **Majority Voting**: For classification, the algorithm assigns the class label that is most common among the K nearest neighbors to the new data point. This is typically done using majority voting.
5. **Regression**: For regression, the algorithm assigns the average of the K nearest neighbors' target values to the new data point.
6. **Choosing K**: The value of K is a hyperparameter that needs to be specified before training the model. The choice of K can significantly impact the performance of the algorithm. A smaller K value leads to more complex decision boundaries, while larger K value leads to smoother decision boundaries.

KNN is a simple and intuitive algorithm that is easy to understand and implement.

E. Euclidian's Distance

Euclidean distance, also known as Euclidean metric or Euclidean norm, is a measure of the straight-line distance between two points in Euclidean space. It is the most common distance metric used in geometry and spatial analysis.
In Cartesian coordinates, if $p = (p1,p2,...,pn)$ $p = (p1,p2,...,pn)$ and $q=(q1,q2,...,qn)q=(q1,q2,...,qn)$ are two points in Euclidean n-space, then the distance (d) from p to q, or from q to p, can be calculated using the Pythagorean formula.

$$\mathrm{d}(\mathbf{p},\mathbf{q}) = \mathrm{d}(\mathbf{q},\mathbf{p}) = \sqrt{(q_1-p_1)^2+(q_2-p_2)^2+\cdots+(q_n-p_n)^2}$$

$$= \sqrt{\sum_{i=1}^{n}(q_i-p_i)^2}.$$

Equation 3:Euclidean's distance formula

In simpler terms, the Euclidean distance formula calculates the straight-line distance between two points by finding the square root of the sum of the squared differences between their corresponding coordinates along each dimension. This distance metric is commonly used in various applications, including machine learning, clustering, and pattern recognition, to measure similarity or dissimilarity between data points.

G. Manhattan Distance

Manhattan distance, also known as city block distance or L1 distance, is a measure of the distance between two points in a grid-based system. It is calculated as the sum of the absolute differences between the coordinates of corresponding dimensions. In simpler terms, it represents the distance a person would travel if they could only move along grid lines to reach their destination, like navigating the streets of a city block by block.
In mathematical notation, the Manhattan distance between two points
$P1=(x1,y1)P1=(x1,y1)$ and $P2=(x2,y2)P2=(x2,y2)$ in a two-dimensional space is given by:

$$\boldsymbol{D\text{Manhattan}\,(P1,P2) = |x1-x2|+|y1-y2|}$$

The Manhattan distance is often used in machine learning algorithms, especially in clustering algorithms like K-means, and in recommendation systems where it helps measure the similarity between items or users based on their features or preferences.

III. ACTUAL WORK

A. Data Collection and Format

The research uses Zomato Dataset available from the website of Kaggle . The dataset contains information on 1965 restaurants in Delhi NCR as listed on Zomato. The list of restaurants covers the entire NCR Region, last updated on August 30th 2021.
This Data is present in the form of csv file.

1. Columns present initially:

['Restaurant_Name', 'Category', 'Pricing_for_2', 'Locality', 'Dining_Rating', 'Dining_Review_Count', 'Delivery_Rating', 'Delivery_Rating_Count', 'Website', 'Address', 'Phone_No', 'Latitude"Longitude', 'Known_For2', 'Known_For22']

2. Sample representation :

Restaurant_Name:	Cafe Lota
Category:	Cafe, South Indian, North Indian, Beverages
Pricing_for_2:	1200
Locality:	Pragati Maidan, New Delhi
Dining_Rating:	4.9
Dining_Review_Count:	3748
Delivery_Rating:	3.9
Delivery_Rating_Count:	37
Website:	https://www.zomato.com/ncr/cafe-lota-pragati-m...
Address :	National Crafts Museum, Gate 2, Bhairon Marg
Phone_No:	9.17839E+11
Latitude:	28.613429
Longitude:	77.242471
Known_For2:	Pondicherry Fish Curry, Coconut Rabdi
Known_For22:	Artistic Decor, The Service, Natural

B. Exploratory Data Analysis (EDA)

Exploratory data analysis was conducted to gain insights into the distribution of key variables, such as dining ratings and final ratings. Visualizations, including histograms and bar plots, were utilized to summarize the characteristics of the dataset and identify any trends or patterns.

C. Data Cleaning and Feature Engineering

To prepare the dataset for analysis, several preprocessing steps were performed. Missing values were handled appropriately, and irrelevant columns, such as delivery-related information and geographic coordinates, were removed. Additionally, a new feature called **Final_rating** was engineered based on the dining rating and review count to capture the overall popularity of each restaurant.
This Final_rating is formed using below Formula:

$$Final\_rating = Dining\_rating * log10(Dining\_Review\_count)$$

Here One more feature called Pricing Index is also added to scale the price from 0-5.

$$Max\_price = max(Pricing\_for\_2)$$
$$Pricing\_Index = (Pricing\_for\_2)*5/MaxPrice$$

After this one more column is added with the name 'Category_Match' which represents the categories which user selected.

The Feature which are used to train aur model:-

['DINING_RATING','FINAL_RATING','CATEGORY_MATCH ', 'PRICING_INDEX']

D. User Input and Category Selection

Users begin by selecting a city of interest from a predefined list, enabling the recommendation system to focus on localized dining options. Subsequently, they choose up to five preferred restaurant categories from a presented list, aligning with their culinary preferences. These selected categories are then utilized to generate personalized restaurant recommendations tailored to the user's tastes. By incorporating these preferences into the recommendation algorithm, the system prioritizes restaurants that closely match the user's culinary preferences, enhancing the accuracy and relevance of the recommendations and increasing user satisfaction.

E. Model Building

A K-Nearest Neighbors (KNN) classifier was employed as part of the recommendation system, leveraging features such as dining rating, final rating, pricing index, and category match. These features were instrumental in determining the similarity between restaurants and identifying potential recommendations based on user preferences.

Specifically, the KNN model was trained using historical data comprising various restaurant attributes, including dining rating, final rating, pricing index, and category match. This training process enabled the model to learn patterns and relationships within the data, effectively capturing the underlying structure of the restaurant landscape.

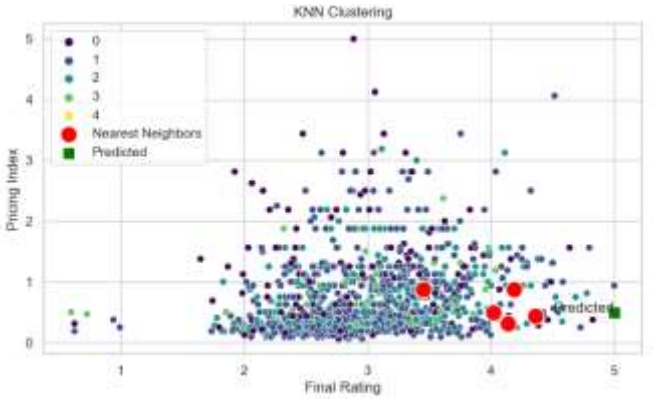
Once trained, the KNN classifier was ready to make predictions for new data points, such as those generated from user input. For instance, when a user provided their preferences for dining, pricing, and restaurant categories, a test data point representing these preferences was created. The KNN model then utilized this test data point to predict the nearest neighbors—i.e., the restaurants most similar to the
user's preferences.

These predicted nearest neighbors formed the basis for personalized restaurant recommendations tailored to the user's specific tastes and preferences. By leveraging the KNN algorithm, the recommendation system was able to provide relevant and targeted suggestions, enhancing the overall user experience and satisfaction with the dining options presented.

IV. RESULTS AND DISCUSSION

A. Recommendation and Classification

Finally we got top 5 recommendations from all the available options based on the nearest neighbors.



This is the visual representation how the neighbors are predicted based on the input provided by the user.

V. CONCLUSION

In conclusion, this paper has presented a novel approach to personalized restaurant recommendations by leveraging content-based filtering and KNN modeling with distance consideration. By integrating key attributes such as cuisine type, restaurant ratings, location, and price range, along with distance from a hypothetical user preference point, the proposed system offers tailored dining suggestions that closely align with individual preferences.

The utilization of content-based filtering enables the system to focus on the intrinsic characteristics of restaurants, ensuring that recommendations are based on relevant attributes rather than solely on user similarities. Additionally, the incorporation of distance consideration enhances the practicality and relevance of recommendations by accounting for geographical proximity to the user's location of interest.

Through the implementation of a KNN model, the system can effectively analyze these attributes and proximity factors to generate personalized restaurant recommendations. By considering both user preferences and geographical proximity, the proposed approach addresses the limitations of traditional recommendation systems and offers a more refined and tailored dining experience for users.

VI. FUTURE WORK

Moving forward, there are several avenues for further exploration and refinement of the personalized restaurant recommendation system. One potential area of focus is enhancing user interaction to gather real-time feedback on recommended restaurants. By implementing interactive features, users can provide valuable insights that can improve the accuracy and relevance of future recommendations.

Another aspect that warrants attention is the development of a more dynamic pricing index. By considering factors such as time of day, day of the week, and seasonal variations, the system can offer more precise pricing recommendations tailored to the user's preferences and budget constraints.

Furthermore, incorporating contextual information such as weather conditions, special events, or user mood could further refine restaurant recommendations. By leveraging contextual data, the system can adapt recommendations to better suit the user's current situation or preferences.

Exploring alternative machine learning algorithms beyond KNN modeling is also worth considering. Neural networks, decision trees, and other techniques may offer advantages in terms of recommendation accuracy and efficiency, warranting further investigation.

REFERENCES

[1] A.Dasgupta and P. Drinea, et al. “Feature Selection Methods for TextClassification”.[ONLINE]Available:www.stat.berkeley.edu/~m mahoney/pubs/kdd07.pdf[

[2] C. Pan and W. Li. “Research paper recommendation with topic analysis,” *In Computer Design and Applications.* IEEE. Vol. 4 (2010), pp. V4-264

[3] E. Gabrielova and C. Lopes, et al. "The Yelp dataset challenge - Multilabel Classification of Yelp reviews into relevant categories", *Ics.uci.edu*, 2017. [Online]. Available: http://www.ics.uci.edu/~vpsaini/.

[4] F. Ricci, L. Rokach and B. Shapira. (2011). “Introduction to Recommender Systems Handbook. Springer

[5] H. Jafarkarimi, A.T.H. Sim and R. Saadatdoost. (2012, June). “A Naïve Recommendation Model for Large Databases.” *International Journal of Information and Education Technology*.

[6] J. A. Konstan and J. Riedl. “Recommender systems: from algorithms to user experience User Model User-Adapt Interact.” Vol.22 (2012), pp. 101–123

[7] R. J. Mooney and L. Roy. “Content-Based book recommendation using learning for text categorization”. In *Proc. Fifth ACM conference on digital libraries*, 2010, pp. 195-204

[8] S. Sawant, and G. Pai. “Yelp Food Recommendation System.” [ONLINE] Available:http://cs229.stanford.edu/proj2013/SawantPaiYelpFood RecommendationSystem.pdf[

[9] Zomato Restaurants in Delhi NCR [ONLINE] Availablhttps://www.kaggle.com/datasets/aestheteaman01/zomato-restaurants-in-delhi-ncr/data

[10] Ramni Harbir Singh, Sargam Maurya, Tanisha Tripathi, Tushar Narula and Gaurav Srivastav “Movie Recommendation System using Cosine Similarity and KNN”. International Journal of Engineering and Advanced Technology (IJEAT) [ONLINE] Available:http://www.edu.dmomeni98.ir/papers/Movie%20Recommendation%20System%20using%20cosine%20similarity%20and%20knn_2020.pdf