Similarity-Based Service Recommendation System

Project Report

Submitted by
Harsha R Chandran
CSE Department
National Institute of Technology Goa

Submitted to
Tailyo Technologies

Academic Year: 2024–2025

Abstract

With the increasing number of digital services available today, selecting the right service based on specific user needs has become a challenging task. This project presents an ML-powered service recommendation system designed to assist users in identifying the most relevant services based on factors such as business type, budget, language preference, and location. The system aims to simplify decision-making by providing accurate, transparent, and user-friendly recommendations.

The solution follows a structured machine learning pipeline that includes data preprocessing, feature encoding, and similarity-based ranking. Categorical user preferences are transformed using one-hot encoding, while service descriptions and optional user keywords are represented using TF-IDF vectorization. Cosine similarity is used to measure relevance between user preferences and available services, enabling the system to rank and recommend the most suitable options. Each recommendation is accompanied by a match score and a qualitative label, along with a brief explanation highlighting the key factors contributing to the recommendation.

The system is deployed using a Streamlit-based web interface, allowing users to interactively provide inputs and view results in real time. Through its modular design and explainable outputs, the project demonstrates how similarity-based machine learning techniques can be effectively applied to build a practical and interpretable service recommendation system.

Contents

1 Introduction

In today's digital ecosystem, users are often required to choose from a wide range of services, each offering different features, pricing models, and availability conditions. Making an informed choice becomes difficult when multiple factors such as business requirements, budget limitations, language support, and location need to be considered simultaneously. This project focuses on building an ML-powered service recommendation system that addresses this challenge by intelligently matching user preferences with relevant services. By combining structured data preprocessing, similarity-based machine learning techniques, and an interactive web interface, the system aims to deliver accurate, transparent, and user-friendly recommendations that simplify the service selection process and enhance the overall user experience.

2 Problem Statement

With the growing number of digital services available, selecting the most suitable service based on user-specific requirements has become increasingly challenging. Users must consider multiple factors such as business type, budget, language preference, and location, making manual comparison inefficient and time-consuming. Moreover, many existing systems lack personalization and fail to provide clear explanations for their recommendations.

This project aims to develop an ML-powered service recommendation system that processes user preferences, ranks relevant services using similarity-based techniques, and generates transparent, explainable recommendations through an interactive interface, thereby simplifying the service selection process.

3 System Architecture

The proposed system follows a five-stage modular pipeline architecture designed to ensure clarity, scalability, and ease of interpretation. The first stage focuses on user input and data preprocessing, where user preferences such as business type, budget, language, location, and optional keywords are collected through the interface. In parallel, the service dataset is cleaned by handling missing values, standardizing categorical fields, and removing inconsistencies to ensure reliable downstream processing.

In the second stage, feature encoding is performed to convert both user inputs and service attributes into machine-learning-ready numerical representations. Cate-

gorical features are transformed using one-hot encoding, while textual information such as service descriptions and keywords is encoded using TF-IDF vectorization. This unified feature representation enables meaningful similarity computation.

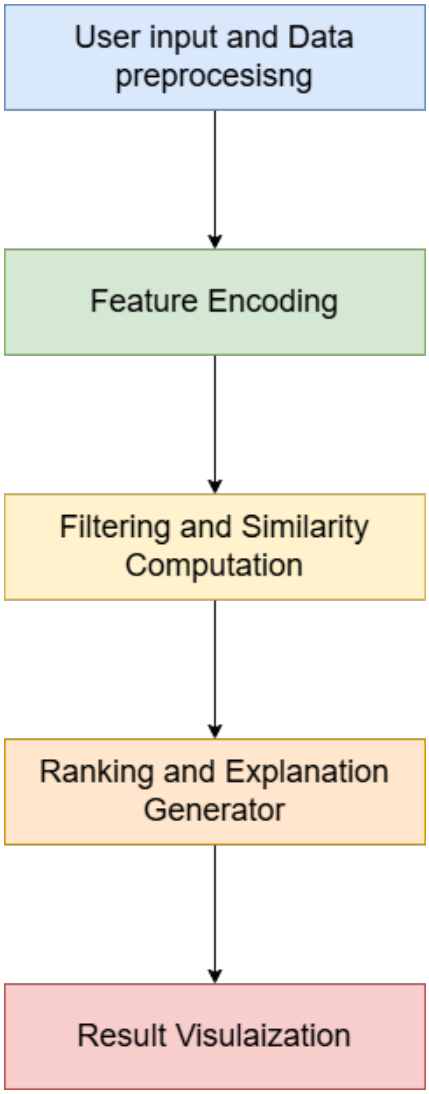

Figure 1: System Architecture Diagram

The third stage involves filtering and similarity computation, where an initial filtering step narrows down the set of services based on user preferences. Cosine similarity is then applied to measure the relevance between the encoded user profile and each service, producing similarity scores that reflect how well a service matches the user's requirements.

In the fourth stage, ranking and explanation generation, services are ranked based on their similarity scores. Each recommendation is assigned a match score and a qualitative label, such as High or Medium, based on predefined thresholds. To improve transparency, an explanation generator highlights the key factors contributing to each recommendation, allowing users to understand why a particular

service was suggested.
Finally, the result visualization stage presents the top-ranked services through a Streamlit-based user interface. The interface displays service details, match scores, quality labels, and explanations in an intuitive and user-friendly manner, completing the end-to-end recommendation workflow.

4 Methodology

4.1 Dataset Description

The dataset contains structured service information including service name, target business type, price category, language support, location, and descriptive text. A cleaned version of the dataset is used throughout the system to ensure consistency and reliability.

4.2 Data Cleaning and Preprocessing

Standard preprocessing steps are applied to ensure that the dataset is consistent, reliable, and suitable for machine learning tasks. Missing values in categorical fields are handled by assigning a default category, which prevents data loss while maintaining structural consistency. Categorical attributes are normalized by standardizing text formats, removing extra whitespace, and ensuring uniform casing to avoid redundant or mismatched categories during encoding. Duplicate service entries are identified and removed to prevent bias in similarity computation and ranking. These preprocessing steps collectively ensure that the dataset is machine-learning ready and can be safely used for feature extraction and similarity-based analysis.

4.3 Input Processing and Feature Encoding

User preferences are collected through the interactive interface and processed in a structured manner to align with the service dataset. Categorical inputs such as business type, budget, language, and location are transformed using One-Hot Encoding, enabling the system to represent discrete attributes numerically without imposing any artificial ordering. Textual information, including service descriptions and optional user-provided keywords, is converted into numerical feature vectors using TF-IDF vectorization. This approach captures the semantic importance of words while reducing the influence of commonly occurring terms. The combination of categorical and textual encodings results in a unified feature representation that supports meaningful similarity computation.

4.4 Service Filtering and Ranking

To improve efficiency and relevance, an initial filtering step is applied based on user preferences, reducing the search space by excluding clearly incompatible services. Following filtering, cosine similarity is used to compute the relevance between the user feature vector and each service feature vector. Cosine similarity is particularly well-suited for high-dimensional sparse data generated by one-hot and TF-IDF encodings. Services are then ranked in descending order of similarity scores, and the top-ranked services are selected as recommendations. This ranking strategy ensures that services most aligned with user requirements are prioritized.

4.5 Similarity Score and Match Quality

Each recommended service is assigned a similarity score ranging between 0 and 1, representing the degree of alignment between user preferences and service attributes. To make the results more interpretable, these continuous scores are mapped to qualitative match categories such as High and Medium using predefined threshold values. This classification allows users to quickly assess the strength of each recommendation without interpreting raw numerical values. The threshold-based approach also enables easy tuning and optimization of recommendation quality based on observed performance.

4.6 Recommendation Explanation

To enhance transparency and user trust, the system generates concise, human-readable explanations for each recommendation. These explanations are derived from matched attributes such as business type compatibility, budget alignment, language support, location relevance, and keyword similarity. By explicitly highlighting the factors that contributed to a recommendation, the system ensures that users understand the reasoning behind each result rather than receiving opaque outputs. This explanation mechanism improves interpretability and supports informed decision-making.

4.7 Streamlit User Interface

The Streamlit application provides an interactive interface for collecting user inputs and displaying recommendations along with scores, quality labels, and explanations.

Figure 2: User Input Interface

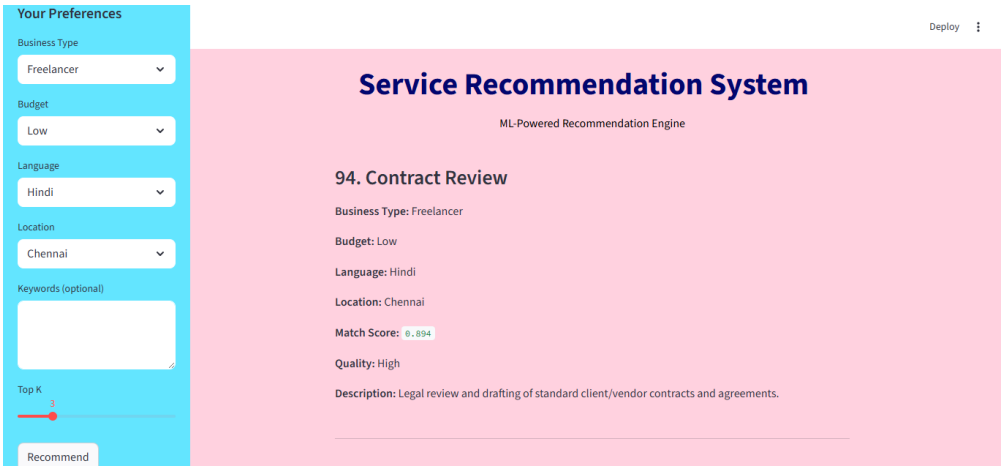

Figure 3: Recommendation Results

5 Evaluation and Optimization

The evaluation of the system focuses on the relevance and consistency of the generated recommendations rather than traditional classification metrics, as the task is similarity-based in nature. System performance is assessed by testing different user input combinations and observing how recommendation rankings change in response to variations in preferences such as business type, budget, location, and keywords.

Optimization is carried out by tuning similarity score thresholds used to assign match quality labels and by experimenting with feature configuration parameters in the text vectorization process. The system also addresses common edge cases, including missing inputs and unseen categorical values, through robust preprocessing and encoder configurations. These evaluation and optimization steps ensure stable, interpretable, and reliable recommendation outcomes across diverse usage

scenarios.

6 Results and Discussion

The system successfully generates personalized service recommendations that align with user preferences. Each recommendation is accompanied by a similarity score and a match quality label, making the results both informative and easy to interpret. Changes in user inputs such as business type, budget, location, or keywords lead to meaningful variations in the ranking of services, indicating that the similarity based approach responds appropriately to different preference combinations. The explanation component further supports transparency by clearly identifying the factors that contributed to each recommendation. Overall, the results demonstrate that the system is effective in providing relevant and interpretable recommendations for practical service selection scenarios.

7 Conclusion

This project presents the complete development of an ML powered service recommendation system, covering data preprocessing, feature encoding, similarity based ranking, and deployment through a Streamlit interface. The modular design of the system supports robustness and ease of extension, while the inclusion of explainable outputs enhances user understanding and trust. The project demonstrates the practical application of machine learning techniques for building intelligent recommendation systems and provides a strong foundation for future improvements such as incorporating user feedback or advanced recommendation strategies.
Future enhancements may include collaborative filtering, adaptive weighting strategies, and real-time feedback integration.

8 Future Scope

The system can be further enhanced by incorporating user feedback to improve personalization and recommendation quality over time. Integrating collaborative filtering techniques alongside the current similarity based approach can help capture user behavior patterns and increase recommendation diversity. Additionally, dynamic feature weighting and support for larger or real time datasets can improve scalability and performance. Deploying the system on cloud platforms and extending the interface with authentication features would enable real world application and usage.