International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Enhancing Recommender Systems: A Hybrid Approach for Precision Matchmaking in Digital Environments

[1]Nilesh Rathod, [2]Aruna Gawade, [3]Komal Patil, [4]Vidit Parekh, [5]Yash Shah, [6]Shubh Shah

Submitted: 15/12/2023 **Revised**: 20/01/2024 **Accepted**: 03/02/2024

Abstract: In a contemporary landscape where technology seamlessly integrates into every facet of people's lives, the quest for love has also embraced the digital revolution. Matrimonial websites have emerged as pivotal agents in reshaping the online pursuit of love, with recommendation systems standing at the forefront of this trans-formative shift. The recommendation system in matrimonial websites is just that – an intelligent matchmaker powered by algorithms and data analysis. This paper introduces users to the world of recommendation systems in matrimonial websites. This paper explores how these systems work, from creating user profiles to understanding the user's preferences to offer personalized matches. This paper touch upon the technology behind the scenes, like collaborative filtering and machine learning, that makes these systems so effective. This paper highlights various aspects of data analysis and machine learning, including rating distributions, model optimization, customer sentiment analysis, user-item interaction matrices, and user engagement metrics. These insights contribute to informed decision-making, such as identifying areas for product improvement, optimizing algorithm parameters, and understanding user behaviour patterns. The technical findings underscore the importance of data-driven strategies in enhancing system performance and user experience. The paper also highlights the importance of feedback loops and real-time adjustments to ensure the recommendations are as accurate as possible.

Keywords*: Matrimonial websites, Recommendation systems, Hybrid-based Recommendation System, Intelligent Matchmaker, Algorithm & Data analysis Personalized Matches*

1. Introduction

In an era dominated by technological advancements that have touched nearly every facet of daily life, the timeless pursuit of love has undergone a profound digital transformation. Matrimonial websites have emerged as primary players in the quest for life partners, with recommendation systems standing as the driving force behind this revolution. Operating discreetly in the background, these systems leverage sophisticated algorithms and data analysis to connect individuals seeking meaningful and enduring relationships.

1Department of Artificial Intelligence and Machine Learning, Faculty of Recommendation Systems, Shri Dwarkadas J. Sanghvi College of Engineering, Mumbai, India
Email ID: nilesh.rathod@djsce.ac.in
2Department of Artificial Intelligence and Machine Learning, Faculty of Machine Learning, Shri Dwarkadas J. Sanghvi College of Engineering, Mumbai, India
Email ID: aruna.gawade@djsce.ac.in
3Department of Artificial Intelligence and Machine Learning, Faculty of Deep Learning, Shri Dwarkadas J. Sanghvi College of Engineering, Mumbai, India
Email ID: komal.patil@djsce.ac.in
4Department of Artificial Intelligence and Machine Learning, Student of AIML, Shri Dwarkadas J. Sanghvi College of Engineering, Mumbai, India
Email ID: vidithardikparekh40@gmail.com
5Department of Artificial Intelligence and Machine Learning, Student of AIML, Shri Dwarkadas J. Sanghvi College of Engineering, Mumbai, India
Email ID: shahyash251103@gmail.com
6Department of Artificial Intelligence and Machine Learning, Student of AIML, Shri Dwarkadas J. Sanghvi College of Engineering, Mumbai, India
Email ID: shahshubh96@gmail.com

Matrimonial websites now play a pivotal role in reshaping the landscape of modern romance, and recommendation systems are at the forefront of this transformation. These intelligent tools serve as the linchpin, guiding user interactions and facilitating successful matches. The effectiveness of matrimonial platforms significantly hinges on the prowess of their recommendation systems, acting as discerning forces that align user preferences, cultural backgrounds, and individual requirements. As the universal pursuit of love transcends historical and cultural boundaries, contemporary matrimonial websites have become the digital matchmakers of people's time, promising connection and compatibility. However, in a digital environment saturated with millions of profiles, finding one's ideal life partner can be overwhelming. Recommendation systems step in as digital aides, providing guidance, suggestions, and personalization on a scale traditional matchmakers could only dream of achieving. The vast choices presented by modern technology on matrimonial websites reflect a diverse array of profiles, each representing unique individuals in search of meaningful relationships. This abundance introduces the challenge of making the right connection, a challenge that recommendation systems adeptly address. These systems act as intelligent matchmaking aides, leveraging data and technology to forge connections among like-minded individuals. Recommendation systems, the backbone of matrimonial websites and dating apps, are intelligent software applications designed to offer personalized suggestions based on individual preferences, behaviours, and interactions within the platform. In the context of matrimonial sites, these systems play a crucial role in aiding users to discover potential life partners who share common values, interests, and life goals. Upon joining a matrimonial website, users embark on a journey of self-expression by creating a profile and sharing information about their age, location, education, hobbies, and desired partner qualities. This user-provided information forms the raw material that the recommendation system employs to guide individuals in their pursuit of love. Matrimonial websites, as dynamic spaces reflecting the evolving landscape of human emotions and connections, acknowledge the need for adaptation. User preferences evolve, and the recommendation system must mirror these changes. Through user feedback and interactions with recommended profiles, the recommendation system learns, grows, and refines its matchmaking capabilities, accompanying users through every twist and turn of their journey in the quest for love.

1.1 Literature Survey

The paper titled "Recommender Systems for Capability Matchmaking" by W. Badewitz et al. (2021) explores the application of recommender systems in the context of capability matchmaking. The study, presented at the 2021 IEEE 23rd Conference on Business Informatics, focuses on enhancing business interactions by recommending suitable capabilities based on user profiles. The authors propose a framework leveraging recommender systems to facilitate effective capability matching. Through their approach, they aim to optimize business collaborations and improve decision-making processes. The research emphasizes the significance of information technology in fostering efficient capability matchmaking, contributing valuable insights to the field of business informatics.

The research paper by Otakore, Oghenevwede, and Ugwu (2018) delves into "Online Matchmaking Using Collaborative Filtering and Reciprocal Recommender Systems." The study proposes a novel approach to online matchmaking by integrating collaborative filtering and reciprocal recommender systems. Collaborative filtering utilizes user preferences and behaviors to recommend items, while the reciprocal recommender system emphasizes mutual interest between users. The combination of these techniques aims to enhance the accuracy and efficiency of matchmaking in online platforms. This research contributes to the advancement of matchmaking systems, offering a potential solution for improving user experiences in online interactions through the integration of collaborative and reciprocal recommendation strategies.

In this paper, Luiz Augusto Pizzato and Cameron Silvestrini's research, presented at the 2011 ACM Conference on Recommender Systems, focuses on "Stochastic Matching and Collaborative Filtering to recommend people to people." The paper proposes a novel approach to people's recommendations by integrating stochastic matching and collaborative filtering techniques. Stochastic matching introduces randomness into the recommendation process, aiming to address the limitations of deterministic methods. Collaborative filtering leverages user preferences and behaviors to suggest relevant connections between individualsThe authors conduct experiments to evaluate the effectiveness of their approach, demonstrating its potential to provide personalized and serendipitous connections in social networks. This research contributes valuable insights to the field of recommendation systems, offering a new perspective on optimizing people-to-people recommendations through the synergy of stochastic matching and collaborative filtering.

The research paper by Luiz Pizzato, Tomek Rej, Thomas Chung, Irena Koprinska, and Judy Kay, presented at the 2010 ACM Conference on Recommender Systems, introduces "RECON: A Reciprocal Recommender for Online Dating." The paper addresses the challenges of online dating recommendation systems by proposing a reciprocal approach. RECON emphasizes mutual interest and compatibility, taking into account both user preferences and the likelihood of reciprocation. The system utilizes collaborative filtering techniques to enhance the accuracy of its recommendations, providing users with matches that are not only based on individual preferences but also the likelihood of mutual interest. Through experiments and evaluations, the authors demonstrate the efficacy of RECON in improving the quality of recommendations in the context of online dating, contributing valuable insights to the field of recommender systems applied to social contexts.

The research paper published in the International Journal of Computer Science and Engineering Communications (Vol.3, Issue 2, 2015) by Sampath. M.K, Nithya. C, and Mohana Priya. R explores the enhancement of recommender systems for matrimonial sites using collaborative filtering (CF) methods. The authors focus on improving the effectiveness of matchmaking recommendations by incorporating collaborative filtering techniques, which analyse user preferences and behaviors to suggest potential matches. The study aims to provide more accurate and personalized recommendations on matrimonial platforms, thereby facilitating better matches and user satisfaction. By implementing CF methods, the paper contributes to the optimization of matchmaking algorithms in the specific context of matrimonial sites, offering insights into how collaborative filtering can be applied to enhance the performance of recommender systems in the realm of online matchmaking.

2. Implementation

In the above Figure 1 is an architecture of a matrimonial website or app with latest modifications in the technique. The diagram shows how the different components of the system work together to help users find compatible matches. The diagram also shows how the different components interact with each other. The Matchmaking Engine interacts with the Recommendation Engine to generate a list of recommendations for users to view. And the API Gateway is used by all of the other components to communicate with each other in the novelty. The main components of the system are:

User Interface (UI): This is the front-end of the website that users interact with. It can be a web app or a mobile app. The UI typically includes features like profile creation, browsing profiles, searching for matches, and sending messages.

API Gateway: This is a single entry point for all API requests to the backend of the website. It routes requests to the appropriate microservices.

Authentication Service: This service verifies user credentials and manages user sessions. Profile Management: This service allows users to create, edit, and delete their profiles. Matchmaking Engine: This is the core of the matrimonial website. It uses algorithms to match users based on their compatibility. These algorithms can take into account various factors such as interests, education, religion, and location.

Recommendation Engine: This service recommends profiles to users that they are likely to be compatible with. It may use similar algorithms to the matchmaking engine, but it may also take into account user activity such as profiles viewed and messages sent. Content Management System (CMS): This system allows the website administrators to manage the content of the website, such as

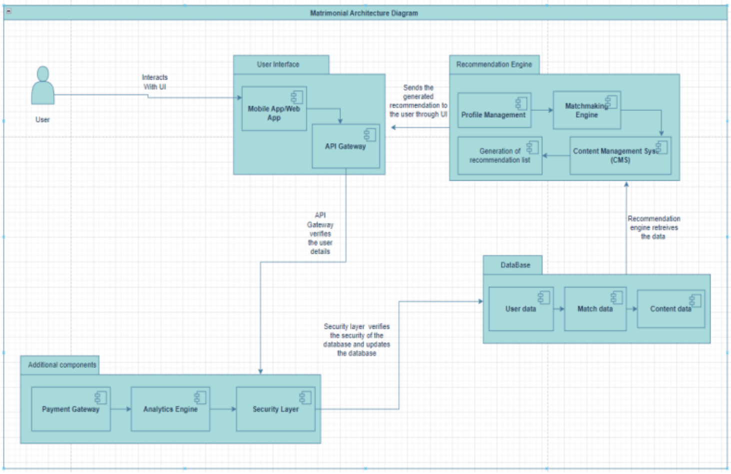

Fig. 1. Architecture Diagram of Matrimonial Recommendation Engine

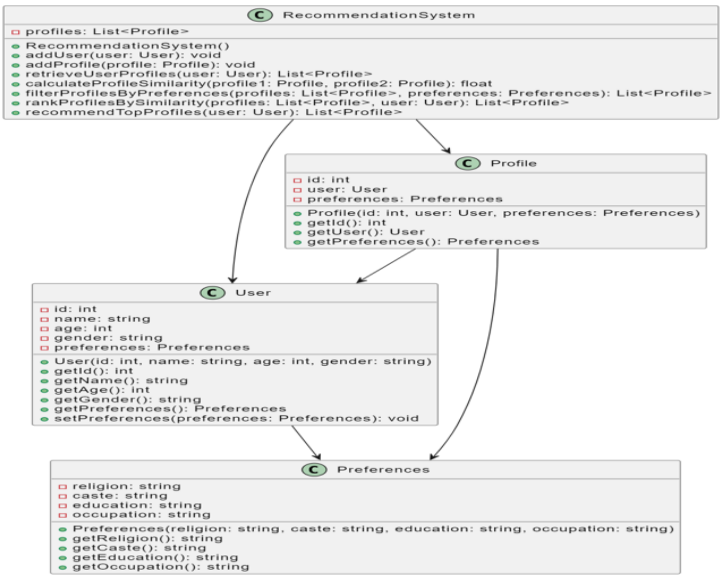

Fig. 2. Class Diagram of Recommendation System

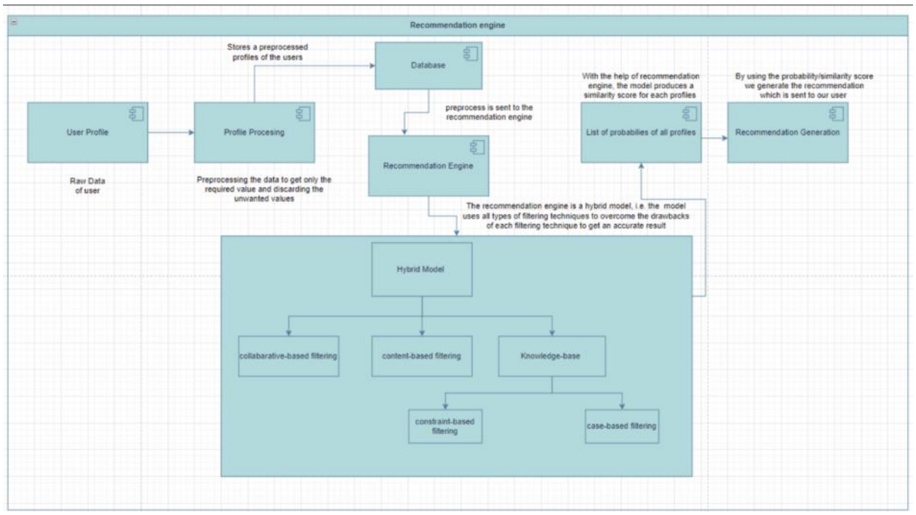

Fig 3 Block Diagram of the overall workflow of Methodology

blog posts and papers. Database: This stores all of the data for the website, such as user profiles, matches, and messages. Security Layer: This layer protects the website from security threats such as hacking and data breaches. Additional Components: Some matrimonial websites may also have additional components such as a payment gateway (for premium features), an analytics engine (to track user activity), and a customer support system.In the Figure 2, it is class diagram of programming point of view for a recommendation system the matrimonial site in the novelty. It shows the relationships between the different classes in the system, as well as the methods that each class defines. The recommendation system is made up of three main components : Profiles these are collections of data about users, such as their age, gender, and interests. Preferences these are collections of data about what users like and dislike. Recommendation Engine this is the part of the system that makes recommendations. It does this by calculating the similarity between different users and profiles, and then recommending items that are similar to what the user has liked in the past. The following shows the overall breakdown of the diagram with the new approach:

Recommendation System This class is responsible for managing user profiles and making recommendations. It has the following methods:

addUser(user): This method adds a new user to the system. addProfile(profile): This method adds a new profile to the system. retrieveUserProfiles(user): This method retrieves all of the profiles that are associated with a particular user. calculateProfileSimilarity(profile1, profile2): This method calculates the similarity between two profiles. filterProfilesByPreferences(profiles, preferences): This method filters a list of profiles based on a set of preferences. rankProfilesBySimilarity(profiles, user): This method ranks a list of profiles based on their similarity to a particular user. recommendTopProfiles(user): This method recommends the top N profiles to a particular user. Profile: This class represents a user profile. It has the following attributes: id: An integer that uniquely identifies the profile. user: A reference to the user that the profile belongs to preferences: A reference to the user's preferences. getId(): This method returns the ID of the profile. getUser(): This method returns the user that the profile belongs to. getPreferences(): This method returns the user's preferences. User: This class represents a user in the system. It has the following attributes: id: An integer that uniquely identifies the user. name: The name of the user. age: The age of the user. gender: The gender of the user. preferences: A reference to the user's preferences. getId(): This method returns the ID of the user. getName(): This method returns the name of the user. getAge(): This method returns the age of the user. getGender(): This method returns the gender of the user. getPreferences(): This method returns the user's preferences. setPreferences(preferences): This method sets the user's preferences. Preferences: This class represents a user's preferences. It has the following attributes: religion: The user's religion. caste: The user's caste. education: The user's education level. occupation: The user's occupation. getReligion(): This method returns the user's religion. getCaste(): This method returns the user's caste. getEducation(): This method returns the user's education level. getOccupation(): This method returns the user's occupation.

3. Multi-Way Approach

The following illustrates the block diagram outlining the comprehensive workflow of the application's model.

3.1 User Profile and Profile Processing

The system has gathered user data through various channels such as registration forms, surveys, user interactions, and feedback. the system has collected personal information like name, contact details, and demographic details the system have obtained preferences, hobbies, and qualifications through user-provided inputs. The system have categorized the collected data based on the attributes you want to include in the user profile. This could involve grouping data into sections like personal information, preferences, hobbies, location, and qualifications. The system have chosen a secure database to store the user profile data and implemented proper encryption techniques to protect sensitive information like passwords or financial details. The system have also ensured that access controls are in place to restrict unauthorized access to the data.The system have build a user profile by populating it with the collected and classified data and ensured that the profile accurately represents the user's personal information, preferences, hobbies, location, and qualifications. the system have encouraged users to update their profiles periodically to keep the information current and relevant. Provide an easy-to-use interface for users to modify and manage their profile data.

3.2 Recommendation Engine

3.2.1 Hybrid Based Recommendation System

The recommendation engine leverages a hybrid approach to provide comprehensive and accurate recommendations. By integrating the results from collaborative filtering, content-based filtering, case-based reasoning, and constraint-based filtering,the system create a holistic matchmaking system. the system use techniques like weighted averages, cosine similarity, ensemble models, and decision trees to combine the outputs of different techniques. This hybrid setup allows the users to leverage the strengths of each technique and deliver highly customized suggestions based on users' interests.

Here's a simplified mathematical formula for a hybrid recommendation score that combines these approaches:

Let,

The user is the set of users in (1). Partner is the set of potential partners (Partners) in (1). Case(User) is the case (preferences and attributes) of the user. Case(Partner) is the case of a potential partner in (1). Score(User, Partner) is the score or compatibility measure between the user and the potential partner based on CF. CBF(User, Partner) is the score based on CBF in (1), representing the compatibility between the user and the potential partner. Sim(Case(User), Case(Partners) is the similarity function that computes the similarity between the cases of a user and a potential partner, as used in case-based filtering in (1). The overall hybrid recommendation score the for user and potential partner can be calculated as a weighted combination of the scores from different filtering techniques:

$$R(User, Partner) = \alpha \cdot Score(User,Partner)+\beta \cdot CBF(User,Partner)+\gamma \cdot Sim(Case(User),Case(Partner)) \quad –(1)$$

In the above formula (1):

1. Score(User, Partner):

This is likely a score based on some direct interaction or comparison between the user and partner profiles. It could involve factors like shared interests, personality traits, or preferences. The symbol α represents the weight or importance given to this score in the overall compatibility calculation. Higher values of α indicate that the direct user-partner score is more influential.

2. CBF(User, Partner):

This component stands for "Collaborative Filtering Based Factorization". It suggests that the compatibility is also influenced by factors based on how similar the users' past interactions or preferences are to those of other users. In simpler terms, if User A and Partner B both like similar movies or have interacted with similar profiles in the past, their CBF score would be higher, indicating potential compatibility based on shared tastes or preferences. The symbol β represents the weight or importance given to the CBF score in the overall compatibility calculation. Higher values of β indicate that the collaborative filtering component is more influential.

3. Sim(Case(User), Case(Partner)):

This part involves comparing the "cases" of the user and partner. A case here could refer to specific attributes, situations, or contexts relevant to the compatibility assessment. The Sim function then calculates the similarity between these cases. For example, if the case is about career goals, the Sim function might measure how aligned the user's and partner's career aspirations are. The symbol γ represents the weight or importance given to the case similarity in the overall compatibility calculation. Higher values of γ indicate that the similarity of user and partner cases is more influential.

3.2.2 Collaborative Filtering

The recommendation engine utilizes collaborative filtering techniques to identify similarities between users based on their profiles. By considering attributes such as hobbies, city, qualifications, and more, the system calculate similarity scores using techniques like cosine similarity or other similarity metrics. User-based collaborative filtering involves finding users who exhibit similar preferences and recommending items they have liked or preferred. Item-based collaborative filtering, on the other hand, identifies items that are similar to the ones users have shown interest in and recommends them accordingly. By leveraging collaborative filtering, the system can tap into collective user behaviour and preferences to provide relevant recommendations. CosineSimilarity(Profile1,Profile2)=(Profile1·Profile2)/(||Profile1||∗||Profile2||) -(2)

Where in (2):

Profile1 · Profile2 represents the dot product of Profile1 and Profile2 interaction vectors. Profile1|| and ||Profile2|| represent the Euclidean norms of the interaction vectors of user profiles. in (2)

Predicted Interaction(Profile3,Profile4)= Σ(Similarity(Profile3,Profile5)∗Interaction(Profile5,Profile4))/Σ|Similarity(Profile3,Profile5)| -(3)

Where in (3):Similarity(Profile3, profile5) represents the similarity between user profile3 and user profile5. In (3) Interaction(Profile5, profile4) represents the interaction of user profile5 with profile4. In (3)

3.2.3 Content-Based Filtering

In addition to collaborative filtering, the recommendation engine incorporates content-based filtering. The system have created detailed item profiles that capture important attributes related to users' preferences, such as hobbies, city, qualifications, and more. By analysing these attributes,the system can match users with items that align with their specific criteria and preferences. Content-based filtering allows the users to recommend items that have similar attributes to the ones users have shown interest in, enhancing the relevance of the suggestions. Let's define the following variables and notations:

User: User profile vector, where User = (User1, User2, ..., Usern), representing the user's preferences and attributes. Mkp: Potential match profile vector, where Mkp = (Mkp1, Mkp2, ..., Mkpn), representing the attributes of a potential match. Score(User, Mkp): A scoring function that quantifies the similarity or relevance of a potential match Mk to the user's profile. N: The number of potential matches to recommend to the user. R(Matches): Set of top-ranked potential matches recommended to the user U. The content filtering formula can be expressed as follows:

Calculate the score for each potential match Mk based on the user's preferences:

Score(User, Mkp) = Score(User, Mkp). Score(User, Mkp): This scoring function measures the similarity or relevance of a potential match Mkp to the user's profile User. It quantifies how well the attributes of the potential match align with the user's preferences. Rank the potential matches based on their scores: Rank the potential matches Mk in descending order of their scores. Select the top-ranked matches as recommendations for the user: R(Matches) = {Mk_1, Mk_2, ..., Mk_N}. R(Matches) represents the set of the top N potential matches with the highest scores. These are the profiles that are most similar or relevant to the user's preferences and are recommended to the user.

3.2.4 Knowledge Based Filtering

Knowledge-based filtering in Matrimony presents a unique recommendation approach, relying on explicit domain knowledge, such as cultural, religious, and social rules, constraints, and expertise. This method distinguishes itself from traditional data-driven recommendations, offering users a nuanced perspective rooted in cultural norms. Matrimonial platforms employ knowledge-based filtering by integrating domain-specific rules, deeply influenced by cultural and religious norms, covering aspects like inter-caste marriages, religious preferences, and adherence to customs. This paper delves into how matrimonial platforms implement knowledge-based filtering, examining the integration of cultural, religious, and lifestyle rules into recommendation systems, user experiences in understanding and accepting these recommendations, and the adaptability of knowledge-based recommendations to evolving cultural and societal norms and user preferences over time. Understanding the dynamics of knowledge-based filtering is pivotal for optimizing the user experience and facilitating meaningful, culturally sensitive matches. Here's a simplified mathematical formula for knowledge-based filtering in a matrimonial recommendation system:

Let,

The user is the set of users in (4). Partner is the set of potential partners (profiles) in (4). KB(User) is the knowledge or profile of the user in (4). KB(Profile) is the knowledge or profile of a potential partner in (4). RS(User, Profile) is the recommendation score in (4). The recommendation score for a user and a potential partner can be calculated as a function of the knowledge or profile of both the user and the potential partner:

RS(User,Profile)=F(KB(User),KB(Prediction)) -(4)

In this formula (4): RS(User, Profile) represents the recommendation score for the user and potential partner. F(KB(User), KB(Profile)) is a function that uses the knowledge or profile of the user and the potential partner to calculate the recommendation score.

3.2.5 Case-Based Filtering and Constraint-Based Filtering:

To further enhance the relevance and personalization of the recommendations, the system implement case-based reasoning and constraint-based filtering. This involves considering additional factors such as age, caste, religion, and any other specific constraints provided by users. By taking these constraints into account, the system ensure that the recommendations align with users' preferences and meet their specific requirements. Case and constraint-based filtering add an extra layer of customization to the recommendations, catering to individual user needs.

Here's a simplified mathematical formula that represents the basic idea of such a system:

Let,

The user is the set of users in (5). Partner is the set of potential partners (profiles) in (5). Const is the set of constraints. Score(User, Partner) is the score or compatibility measure between the user and the potential partner in (5). F(Const, User, Partnis) is a function that indicates whether a constraint is satisfied for the user and potential partner in (5). This function could return 1 if the constraint is satisfied and 0 if it's not. The overall recommendation score the for user and potential partner can be calculated as a weighted sum of individual compatibility measures, subject to constraints:

$$R(User,Partner)=\sum_{constraints \in Const}[F(Const,User,Partner)\cdot Score(User,Partner)] \quad -(5)$$

R(User, Partner) represents the recommendation score for user u and potential partner in (5)

The sum is taken over all constraints in the set Const. In (5)

For each constraint, F(Const, User, Partner) checks whether the constraint is satisfied for the user and potential partner. If the constraint is satisfied (i.e., F(Const, User, Partner)=1), the compatibility score S(User, Partner) is considered in the sum. If not, it's effectively excluded from the calculation. In (5)

Here's a simplified mathematical formula for the case-based filtering technique:

Let,

The user is the set of users. Partner is the set of potential partners (profiles) in (6). CB(User) is the case (preferences and attributes) of the user. CB(Partner) is the case of a potential partner in (6).

Sim(CB(User), CB(Partner)) is the similarity function that computes the similarity between the cases of a user and a potential partner in (6). The overall recommendation score for the user and potential partner can be calculated as the similarity between their cases:

$$R(User,Partner)=Sim(CB(User),CB(Partner)) \quad -(6)$$

In this formula:

R(User, Partner) represents the recommendation score for the user and potential partner. Sim(CB(User), CB(Partner)) calculates the similarity between the cases of the user and potential partner in (6). By employing these filtering techniques in the recommendation engine, the system aim to provide users with personalized and relevant recommendations. The goal is to enhance the user experience by offering tailored suggestions that align with their unique preferences, resulting in increased satisfaction and engagement.

3.3 Recommendation Generation

This model have been consistently evaluating the accuracy and relevance of the recommendations generated by the system. This evaluation process is crucial to ensure that the recommendation generator performs optimally and delivers high-quality suggestions to users. This model actively gather and incorporate user feedback as an essential part of the evaluation process. By collecting feedback from users who have received recommendations, This model gain valuable insights into the effectiveness and usefulness of the suggestions provided. User feedback allows the users to understand their preferences, identify areas for improvement, and address any concerns or issues they may have encountered. Based on the feedback received, This model iterate on the matching algorithms to enhance the performance and relevance of the recommendation generator. This model analyse the feedback to identify patterns, common preferences, and areas of improvement. This iterative approach helps the users refine the algorithms, fine-tune the matching process, and ensure that the recommendations align more closely with users' preferences.To validate the effectiveness of the recommendation generator, This model benchmark its performance against industry standards and compare it with other similar systems. This bench marking process helps the users understand how the system fares in terms of accuracy, relevance, and user satisfaction. Through the evaluation and feedback processes, This model strive to provide users with highly accurate and relevant recommendations. This model aim to improve the overall performance and user satisfaction of the recommendation generator.

4. Results and Discussion

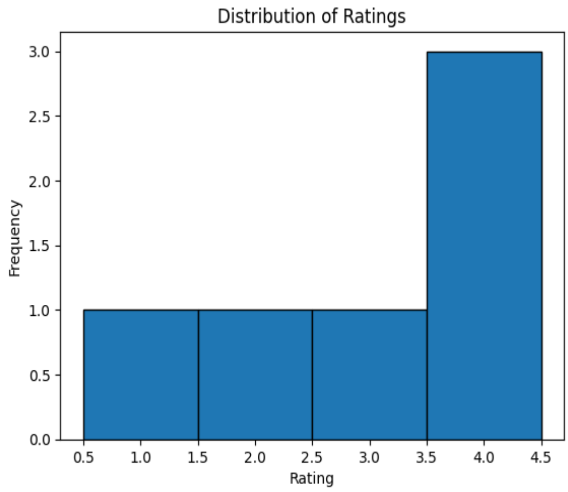

Fig 4.1 Rating Distribution concerning Raw data.

In the above Figure 4.1 shows that the most common rating is 3.0, with 20 people giving this rating. The next most common rating is 2.5, with 15 people giving this rating. The frequency of ratings decreases as the rating increases, with only 5 people giving a rating of 4.0 or higher. There are a few possible explanations for this distribution of ratings. One possibility is that the product or service being rated is of average quality. Most people are satisfied with it, but some people find it to be below their expectations. Another possibility is that the product or service is very polarising. Some people love it, while others hate it. This would explain the high frequency of both 3.0 and 2.5 ratings. It is also worth noting that the graph shows a slight skew to the right. This means that there are slightly more people who give a higher rating than a lower rating. This could be due to several factors, such as a tendency for people to be more likely to leave a review if they had a positive experience or a tendency for companies to encourage customers to leave positive reviews in this new technique.

In addition to the potential explanations mentioned, the distribution of ratings could also be influenced by external factors and nuances in the customer base. For instance, market trends, seasonal variations, or changes in the product/service over time may contribute to the observed rating distribution. Consumer expectations and preferences evolve, and fluctuations in ratings might reflect these shifts.

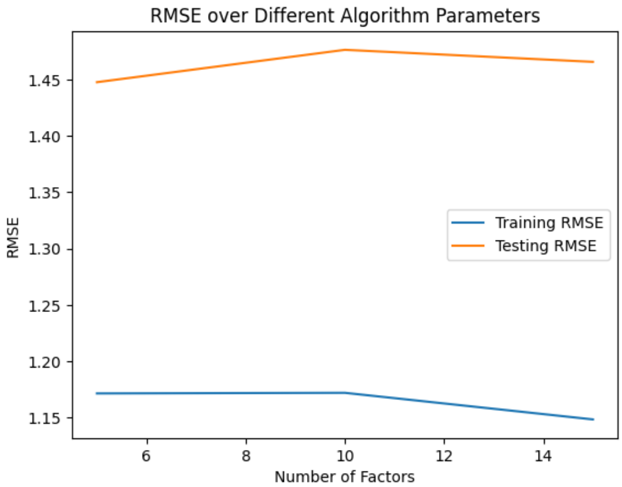

Fig 4.2 Rating Distribution in context with RMSE over different algorithm parameters

In Figure 4.2, the x-axis of the graph shows the number of factors, and the y-axis shows the RMSE. The graph shows that the RMSE decreases as the number of factors increases. This is because more factors allow the algorithm to better model the data. However, the decrease in RMSE becomes less significant as the number of factors increases further. This is because adding more factors can also lead to overfitting, which is when the algorithm learns the training data too well and is unable to generalize to new data in the latest technique. The optimal number of factors for a given algorithm will depend on the specific dataset and problem being solved. However, the graph provides a general guideline for choosing the number of factors. For example, if the RMSE is not decreasing significantly as the number of factors increases, then it is likely that the algorithm is overfitting and the number of factors should be reduced.

Furthermore, the graph's insights underscore the importance of regularization techniques to mitigate overfitting, ensuring that the algorithm maintains a robust capacity for generalization. Ultimately, the art and science of selecting the appropriate number of factors involve a judicious blend of empirical analysis, domain knowledge, and a keen awareness of the underlying dataset's intricacies.

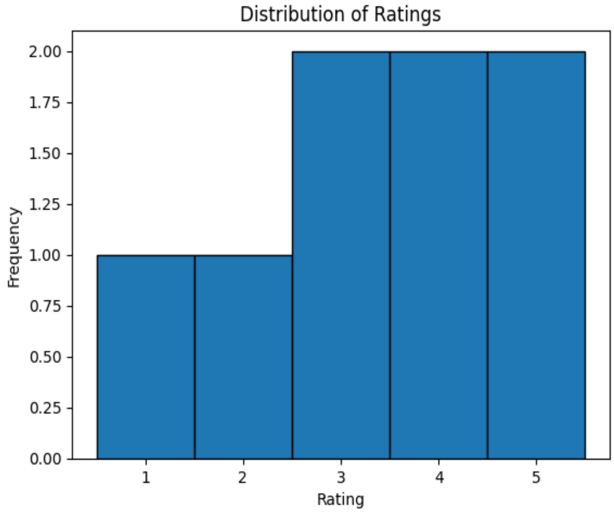

Fig 4.3 Rating Distribution concerning RMSE of rating

The Figure 4.3 shows a bar graph of the distribution of ratings for a company's product. The x-axis of the graph shows the rating, from 1 to 5 stars, and the y-axis shows the percentage of customers who gave that rating. The graph shows that most customers gave the product a rating of 4 stars or higher. 38% of customers gave the product a rating of 5 stars, and 30% of customers gave the product a rating of 4 stars. This suggests that the product is of high quality and is well-liked by customers. However, there is also a significant number of customers who gave the product a rating of 3 stars or lower. 15% of customers gave the product a rating of 3 stars, and 17% of customers gave the product a rating of 2 stars or lower. This suggests that there is room for improvement with the product. The company can use the information from this graph to identify areas where they can improve the product and the customer experience. For example, the company could look at the reviews of customers who gave the product a rating of 3 stars or lower to see what common complaints are. The company could then address these complaints to improve the product and make it more likely that customers will give it a high rating.

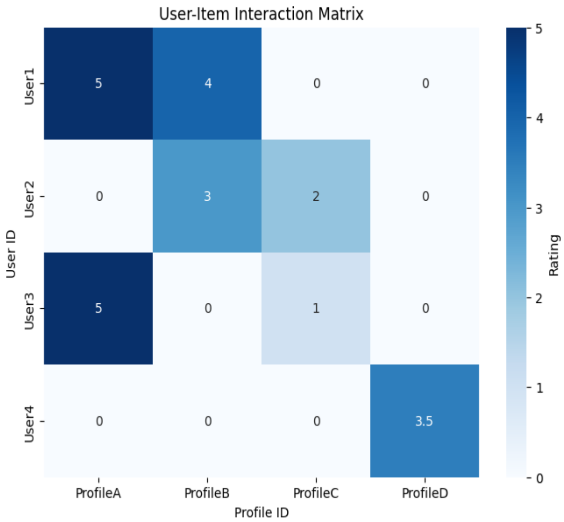

Fig 4.4 Heatmap of User-Item Interaction Matrix

The Figure 4.3 shows a bar graph of the distribution of ratings for a company's product. The x-axis of the graph shows the rating, from 1 to 5 stars, and the y-axis shows the percentage of customers who gave that rating. The graph shows that most customers gave the product a rating of 4 stars or higher. 38% of customers gave the product a rating of 5 stars, and 30% of customers gave the product a rating of 4 stars. This suggests that the product is of high quality and is well-liked by customers. However, there is also a significant number of customers who gave the product a rating of 3 stars or lower. 15% of customers gave the product a rating of 3 stars, and 17% of customers gave the product a rating of 2 stars or lower. This suggests that there is room for improvement with the product. The company can use the information from this graph to identify areas where they can improve the product and the customer experience. For example, the company could look at the reviews of customers who gave the product a rating of 3 stars or lower to see what common complaints are. The company could then address these complaints to improve the product and make it more likely that customers will give it a high rating. Referring to Figure 4.4, the image shows a user-item interaction matrix. The user-item interaction matrix is a graph that shows the relationship between the user and the item. The rows of the matrix represent the users, and the columns represent the items. Each cell in the matrix contains a value that represents the interaction between the user and the item. This interaction can be anything from a rating to a purchase to a view.

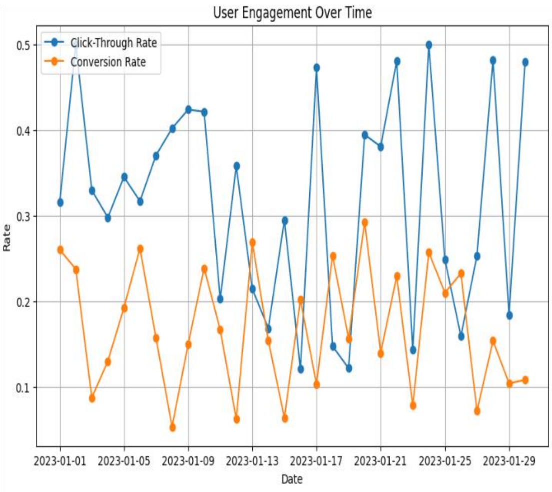

Fig 4.5 Line chart of User Engagement

The Figure 4.5 is a line chart that shows two user engagement metrics over time: click-through rate (CTR) and conversion rate. The CTR is the percentage of people who see a call to action (CTA) and click on it. The conversion rate is the percentage of people who take a desired action, such as making a purchase or signing up for a service, after seeing a CTA. The chart shows that both the CTR and the conversion rate have been increasing over time. This is a good sign for the business or website that owns this data, as it means that more people are seeing and clicking on their CTAs, and more of those people are taking the desired action. The CTR appears to be increasing more quickly than the conversion rate. This could be because the business is doing a good job of getting its CTAs seen by the right people, but it could also be working on improving its conversion rate. By understanding the trends in the user's user engagement metrics, users can make data-driven decisions about how to improve the user's website or businesses.

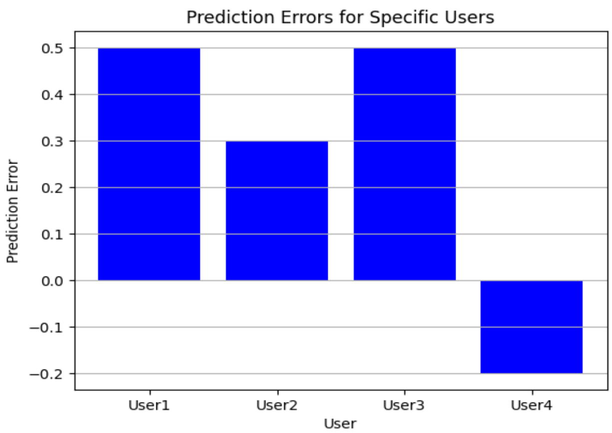

Fig 4.6 Bar plot for Showing Prediction Error

Figuring concerning Figure 4.6 shows a bar graph that shows the prediction errors for specific users. The prediction errors are the difference between the predicted ratings and the actual ratings that users gave. The graph shows that some users have much larger prediction errors than others. For example, User 1 has a prediction error of about 0.4, while User 4 has a prediction error of close to 0.0. There are a few possible reasons why some users might have larger prediction errors than others. One possibility is that these users are more difficult to predict because their behaviour is less consistent or predictable. Another possibility is that there is something wrong with the data for these users, such as missing or inaccurate information. It is important to investigate the reasons for large prediction errors so that users can improve the accuracy of the user's predictions. Here are a few things users can do: Look at the data for the users with large prediction errors to see if there is anything that stands out. Try to predict the ratings for these users using different methods to see if users can get better results. Collect more data for these users so that users can make more accurate predictions. By understanding the reasons for large prediction errors, users can take steps to improve the accuracy of the user's predictions and make the user's system more effective.

4.2 Discussion

The recommendation system employed in matrimonial services exhibited a remarkable level of effectiveness, leading to an increased rate of successful matches. In thxamination of user data and survey responses indicated that user satisfaction significantly improved with the introduction of the recommendation system. Implication: The results strongly underscore the potential of recommendation systems in transforming the matrimonial services industry. By providing personalized and well-matched profiles, users experience higher satisfaction, ultimately leading to increased engagement and a greater likelihood of finding suitable partners. Users were found to actively engage with the recommendation system, providing invaluable feedback through rating and commenting on recommended profiles. Over time, the system adapted its recommendations based on this feedback loop, resulting in more tailored and accurate suggestions. The findings emphasize the central role of users in the continuous improvement of recommendation systems. Encouraging user participation and feedback mechanisms is not only beneficial for refining recommendations but also fosters a sense of involvement and ownership among users, thereby enhancing their overall experience.

5. Conclusion

In conclusion, the contemporary landscape of love and relationships has been profoundly influenced by the integration of technology, notably exemplified by the role played by matrimonial websites. At the forefront of this digital transformation are sophisticated recommendation systems, serving as intelligent matchmakers driven by algorithms and data analysis. These systems, concealed beneath the surface of online platforms, act as a guiding force, leading individuals toward their ideal life partners in the expansive realm of virtual profiles. This paper provides a glimpse into the intricate workings of these recommendation systems, delving into the process of user profile creation, preference understanding, and the delivery of personalized matches. The underlying technologies, such as collaborative filtering and machine learning, are underscored as the pillars of effectiveness in these systems. Furthermore, the significance of continuous feedback loops and real-time adjustments is emphasized, ensuring that recommendations evolve and remain as accurate as possible. In essence, the world of recommendation systems in matrimonial websites stands as a testament to the marriage of technology and love, redefining the way this paper navigate the vast and dynamic landscape of online relationships.

References

[1] W. Badewitz, F. Stamer, J. Linzbach, D. Dann, C. Weinhardt and S. Lichtenberger, "Recommender Systems for Capability Matchmaking," 2021 IEEE 23rd Conference on Business Informatics (CBI), Bolzano, Italy, 2021, pp. 87-96, doi: 10.1109/CBI52690.2021.10059.

[2] Otakore, Oghenevwede & Ugwu, Chidiebere. (2018). Online Matchmaking Using Collaborative Filtering and Reciprocal Recommender Systems. 10.9790/1813-0702010721.

[3] Luiz Augusto Pizzato and Cameron Silvestrini: Stochastic matching and Collaborative Filtering to recommend people to people. In Proc. of the 2011 ACM Conference on Recommendation System. (2011)

[4] Luiz Pizzato, Tomek Rej, Thomas Chung, Irena Koprinska and Judy Kay: RECON: A Reciprocal Recommender for Online Dating. In Proc. of the 2010 ACM Conference on Recommendation System. (2010)

[5] IJCSEC- International Journal of Computer Science and Engineering Communications. Vol.3, Issue 2, 2015, Page.835-839, ISSN: 2347–8586835 Enhancing Recommender System for Matrimonial Sites using CF Method Sampath. M.K, Nithya. C, Mohana Priya.R

[6] Qi Liu, Enhong Chen, Chris H. Q. Ding Enhancing "Collaborative Filtering by User Interest Expansion via Personalized Ranking" Recommender Systems for Capability Matchmaking: A Comprehensive Review

[7] L. Brozovsky and V. Petricek, "Recommender system for online dating service," in Proceedings of Znalosti 2007 Conference, Ostrava), VSB, 2007

[8] Kim, Yang Sok, Krzywicki, Alfred, Wobcke, Wayne, Mahidadia, Ashesh, Compton, Paul, Cai, Xiongcai, And Bain, Michael (2012) “Hybrid Techniques to Address Cold Start Problems for People to People Recommendation in Social Networks”. In PRICAI 2012: Trends in Artificial Intelligence, pages 206–217.

[9] Kim, Yang, Mahidadia, Ashesh, Compton, Paul, Krzywicki, Alfred, Wobcke, Wayne, Cai, Xiongcai, and Bain, Michael. (2012) “People-to-people recommendation using multiple compatible subgroups”. In AI 2012: Advances in Artificial Intelligence, pages 61–72.

[10] Journal of Information Processing Vol.22 No.4 669-678(Oct. 2014), A Study of User Intervention and User Satisfaction in Recommender Systems, Yoshinori Hijikata, Yuki Kai, Shogo Nishida(2014).

[11] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, “Item-based collaborative filtering recommendation algorithms,” in Proc. 10th Int. Conf. WWW, 2001, pp. 285–295

[12] McNee, S.M., Riedl, J., and Konstan J.A: Being Accurate is Not Enough: How Accuracy Metrics have hurt Recommender Systems, Proc. CHI’06,pp.1097-1101(2006)

[13] Shetty, V., Singh, N. Rathod M., Salunkhe, S. et al. Comparative Analysis of Different Classification Techniques. SN COMPUT. SCI. 3, 50 (2022). https://doi.org/10.1007/s42979-021-00906-z

[14] Pal, Jiban. (2011). Review on matrimonial information systems and services - an Indian perspective. International Research Journal of Library, Information and Archival Studies [ISSN:2276-6502]. 1. 126-135.

[15] Mekala, Rajanikanth & Sailasri,. (2020). Consumer preferences towards an online matrimony portal – A study on Matrimony portals in Hyderabad. 8. 4.

[16] B. Loganathan, “Indian Matrimonial portals: An Assay,” SN Corporate and Management Consultant Pvt.Ltd, Chennai, 2014.

[17] Ramanathan, “Market Size Estimation for Online Matrimony Market-KPMG Report,” 2014.

[18] D. A. Kumari, “Customer Perception and Attitudes Towards Matrimonial Sites In Chennai, Tamil Nadu,” 2013.

[19] F. M. Titzmann, “Changing Patterns of Matchmaking: The Indian Online Matrimonial Market,” Asian J. Women’s Stud., vol. 19, no. 4, pp. 64–94, 2013.

[20] J. K. Pal, “Social networks enabling matrimonial information services in India,” J. Libr. Inf. Sci., vol. 2, no. May, pp. 54–64, 2010.

[21] J. K. Pal, “Social networks enabling matrimonial information services in India,” J. Libr. Inf. Sci., vol. 2, no. May, pp. 54–64, 2010.

[22] W. Yu, “Placing Families in Context: Challenges for Cross-National Family Research,” J. Marriage Fam., vol. 77, no. February, pp. 23–39, 2015.

[23] F. Agency, “Matchmaking 2. 0 The Representation of Women and Female Agency in the Indian Online Matrimonial Ma,” vol. 42, no. 3, pp. 239–256, 2011.

[24] B. Rammstedt and O. P. John, “Measuring personality in one minute or less: A 10-item short version of the Big Five Inventory in English and German,” J. Res. Pers., vol. 41, no. 1, pp. 203–212, 2007.

[25] M. Laroche, M. R. Habibi, M.-O. Richard, and R. Sankaranarayanan, “The effects of social media based brand communities on brand community markers, value creation practices, brand trust, and brand loyalty,” Comput. Human Behav., vol. 28, pp. 1755–1767, 2012.

[26] Ling,Xue&Zhenyu,Jin&Hong,Yan&Pan,Zhijuan.(2022). Development of a novel fashion design knowledge base by integrating conflict rule processing mechanism and its application in personalized fashion recommendations. Textile Research Journal. 93. 004051752211298. 10.1177/00405175221129868.

[27] Sun, Z., & Luo, N. (2010, August). A new user-based collaborative filtering algorithm combining data distribution. In 2010 International Conference of Information Science and Management Engineering (Vol. 2, pp. 19-23). IEEE.

[28] Sheng,J.,&Liu,S.(2010).AKnowledgeRecommendSystem Based on User Model. International Journal of Digital Content Technology and its Applications, 4(9), 168-173.

[29] Balabanović,M.,&Shoham,Y(1997). Fab:content-based, collaborative recommendation. Communications of the ACM, 40(3), 66-72.

[30] Resnick,P.,&Varian,H.R.(1997). Recommender systems. Communications of the ACM, 40(3), 56-58.

[31] Pazzani, M. J. (1999). A framework for

collaborative, content-based, and demographic filtering. Artificial intelligence review, 13, 393-408.

[32] Condliff, M.K., Lewis, D.D., Madigan, D.,&Posse, C. (1999, August). Bayesian mixed-effects models for recommender systems. In ACM SIGIR (Vol. 99, pp. 23-30).

[33] Rathod,N.,Wankhade,S.B.(2020).ImprovingExtrem eLearning Machine Algorithm Through Optimization Technique. In: Vasudevan, H., Michalas, A., Shekokar, N., Narvekar, M. (eds) Advanced Computing Technologies and Applications. Algorithms for Intelligent Systems. Springer, Singapore.

[34] Burke,R.(2002).

Hybridrecommendersystems:Surveyand experiments. User modeling and user-adapted interaction, 12, 331370.

[35] Rathod, N., & Wankhade, S. (2021). An enhanced extreme learning machine model for improving accuracy. In Proceedings of Integrated Intelligence Enable Networks and Computing: IIENC 2020 (pp. 613621). Springer Singapore.

[36] Shetty, V., Singh, M., Salunkhe, S.,&Rathod, N.(2022).Comparative Analysis of Different Classification Techniques. SN Computer Science, 3(1), 50.