Serendipity and Diversity Boosting for Personalized Streaming Media Recommendation

Gaetano Manzo[1,2,*,†], Yvan Pannatier[1,†], Gabriel Autès[3], Michaël De Lucia[4], Jean-Gabriel Piguet[1] and Jean-Paul Calbimonte[1,5]

[1]*University of Applied Sciences and Arts Western Switzerland HES-SO Valais-Wallis, Sierre, Switzerland*
[2]*National Institutes of Health (NIH), Bethesda, MD, USA*
[3]*SRG SSR, Genève, Switzerland*
[4]*Radio Télévision Suisse RTS, Genève, Switzerland*
[5]*The Sense Innovation and Research Center, Lausanne and Sion, Switzerland*

Abstract

Streaming media platforms constitute a significant source of information and entertainment for different population segments. Although major corporations have taken the lead in market share, public media companies have also started to produce and broadcast films, series, and documentaries centered on locally-created content. Moreover, beyond the purely commercial goals of major corporations, these public streaming platforms have the mission of expanding the cultural landscape of the viewers, for instance, through the exploration of content produced in other regions and other languages, especially in multicultural societies such as Switzerland. In such a context, this paper proposes a novel approach for personalized recommendations of streaming media content, focusing on serendipity and multicultural diversity, while minimizing the need for personal data sharing. The approach is based on the feature extraction from user media consumption and a combination of data-driven recommendation algorithms. The approach has been tested with real data from the public PlaySuisse streaming platform.

Keywords

Recommendation Systems, Serendipity, Multicultural Diversity, Feature extraction

1. Introduction

In recent years, traditional and public media have faced new competition from streaming media providers like Netflix, Disney+, and HBO. This has led to the creation of ideological bubbles, as noted by Pariser et al.[1]. These large corporations offer content on demand for a subscription fee and have significant influence over their viewers. To counter this, national public television and broadcasting companies have introduced similar services with different content, distribution, and impact visions. These public streaming services aim to provide spaces for open dialogue and

IIR2023: 13th Italian Information Retrieval Workshop, 8th - 9th June 2023, Pisa, Italy

*Corresponding author.
†These authors contributed equally.
gaetano.manzo@hevs.ch (G. Manzo); yvan.pannatier@hevs.ch (Y. Pannatier); Gabriel.Autes@srgssr.ch (G. Autès); Michael.Delucia@rts.ch (M. De Lucia); jean-gabriel.piguet@hevs.ch (J. Piguet); jean-paul.calbimonte@hevs.ch (J. Calbimonte)
0000-0002-1682-6865 (G. Manzo); 0000-0002-4890-3327 (Y. Pannatier); 0000-0002-3938-8327 (J. Piguet); 0000-0002-0364-6945 (J. Calbimonte)

cultural exchange while offering personalized content without limiting viewers to a standard profile.

This paper presents an innovative approach to personalize recommendations for streaming media content with a focus on cultural diversity and serendipity. The proposed method requires minimal explicit user profile information to preserve privacy, as most user data remains undisclosed. The approach utilizes knowledge extraction techniques to estimate user preferences based on viewing patterns. Additionally, it includes (i) knowledge inference methods to determine user scores for viewed content and (ii) knowledge discovery techniques to classify media consumption based on time and language preferences. The personalization effectiveness is evaluated using a combination of content-based, collaborative filtering, and user clustering methods. Results show that users within the same cluster have a high level of serendipity, which means they are more likely to discover new content with a low risk of disliking it.

The approach was developed and tested in collaboration with SRG SSR, a non-profit public media company that provides audiovisual services for the Swiss federal government. The testing and evaluation were carried out using real data from PlaySuisse, a streaming platform launched in 2020[1].

The rest of the paper is organized as follows. Section 2 presents the state-of-the-art of media recommendations, particularly on serendipity and cultural diversity. Section 3 describes the methodology and architecture of the system. Section 4 presents an evaluation of the approach using data from the PlaySuisse platform. Finally, Section 6 concludes the paper.

2. Related Work

Streaming media platforms have powerful recommendation algorithms that provide users with highly relevant suggestions. The main goal of these recommendations was to increase businesses' profit by meeting customer needs [2]. Consequently, the focus is on maximizing viewing time, avoiding switching to other content providers, and maintaining fidelity over time. However, these algorithms also have the effect of locking the user into a filter bubble [1]. *Serendipity* comprises different ways to reduce the effect of this bubble [3, 4]. When related to recommendation systems, Kotkov et al. define serendipity as a property that reflects how good a recommendation system is at suggesting serendipitous items that are relevant, novel, and unexpected [5]. Novelty and unexpectedness require serendipitous items to be relatively unpopular and significantly different from a user profile [6].

However, research on this topic has gathered comparatively less attention in the community. In their extensive review of the literature on recommendation systems between 2015 and 2020, Alhijawi et al. [2] explain that a recommendation system must try to fulfill five goals: relevance, diversity, coverage, novelty, and serendipity. According to their study, only 2.3% of the articles aimed to generate serendipitous recommendations. Among the principal works on this topic, we can mention SIRUP, which combines both novelty and coping potential metrics to generate recommendations for content-based filtering [7]. In other domains such as health information and news, computational serendipity models have been explored, formalizing the concepts of surprise and curiosity [8]. Finally, authors in [9] focused on the definition of user and content

[1]PlaySuisse platform: https://www.playsuisse.ch/.

elasticity, which can be quantified and used to build a relevance network for both surprising and attractive recommendations.

Contextual information, such as time, place, the company of other people, and other factors affecting the viewing experience, is not always considered by recommendation algorithms [10]. However, this information can be crucial. For instance, recommending a movie during the week to a user that only watches series may have poor relevancy. Contextual information can be applied at various stages of the recommendation process, including at the pre-filtering and the post-filtering stages [10, 11]. As detailed in [12], contextual attributes can be observable and unobservable. The latter introduces the additional challenge of discovering these features through latent and indirect information.

Recent recommendation systems use extensive user information. For instance, Sridhar et al. collect information about users by scraping social media such as Facebook [13], while Ramzan et al. perform sentiment analysis using user's feedback to improve the recommendations [14]. To a lesser extent, Subramaniyaswamy et al. retrieve user preference to rank their favorite movies [15].

Beyond existing works, this paper proposes a recommendation approach that focuses on cultural diversity as one of the main sources of serendipity, intending to enlarge the streaming media offer to users. At the same time, the approach considers context-aware parameters relative to watching time-of-day and day-of-week to increase relevance. Finally, and as opposed to most recommender systems in streaming media, this work considers almost no explicit demographic and rating information. Instead, these data items are obtained through a knowledge extraction process shown in the next section.

3. Methodology

This section describes the methodology employed for our personalized recommendation approach. We start by describing the dataset and the knowledge extracted. Then, we explore the core of the architecture —the Recommendation Engine— and the algorithm within. Finally, we discuss the metrics implemented to evaluate our system.

3.1. Dataset

The dataset is divided in two parts and corresponds to all data acquired during the platform's first year (2021). The first part includes 1855 rows (assets) and 182 columns (features). It contains information about the assets available on the platform, i.e. films, documentaries, and series that users can watch through the PlaySuisse streaming service. In addition to the basic asset information, such as name and the type of asset, we can find 25 columns corresponding to asset categories and 124 columns to subcategories. The remaining features are the language of the asset, platform information (e.g., publication date of the asset), and information such as the duration of the content.

The second part of the dataset includes more than 3.6 million rows and 12 columns. Each row corresponds to an interaction between a user and an asset, i.e., different time intervals in which a user watched a specific asset. Each column corresponds to a particular feature related to the user-asset interaction. We find information such as the user identifier, the watched asset's id, and the percentage of content watched.

3.2. Data Cleaning and Knowledge Extraction

The datasets from PlaySuisse do not provide personal information about the user, due to strict privacy protection policies. In addition, given that the data was produced in early versions of the platform, there are several inconsistencies in the dataset. Therefore our first goal was to clean the datasets, and to identify and extract implicit information about the users to be able to feed the recommendation algorithms.

We started by discarding irrelevant features and data, i.e., user-asset interactions with no user id; user-asset interactions with no asset id; user-assets with non-numeric asset id, etc. because these missing features made impossible to establish the relation between the user and the asset watched. One of the main issues that is present in both datasets is that some features are expressed in different formats. For example, *french language* can be expressed as *french*, *French*, *fr*, etc.. We decided to define one format for those features and standardize them. We finally choose to restrict the number of features for our first experiments, from the initial 182 features. At this stage, we kept the information on languages, categories, and asset types. In order to prevent any potential bias in the model's output, we made the decision to remove the subcategories due to their significant imbalance.

We created new features to enhance context-aware recommendation, personalization, and serendipity. The knowledge extraction of these features is crucial for the recommender engine since the provided dataset misses those essential features due to privacy concerns. Since each record in the user-asset interaction dataset has a timestamp, it is possible to extract two features: the day and time-of-day (e.g., morning, evening) the asset was watched. The third feature that was estimated is the rating that a user provides for the watched asset. This is essential information to understand if the user liked or not the asset. The estimation is performed in two steps. First, we calculate the percentage of the asset that has been watched by the user. We can infer it because we know when the user started/finished watching the asset, and we also have the duration of each asset. Once this information is obtained, a rank can be estimated by assuming that a user who has completed an asset is more likely to have liked it.

Finally, the last feature we engineered is an estimation of the principal language of the user. This feature enables us to facilitate cross-cultural exchanges by recommending assets viewed by users with a specific main language to users with another main language. To compute this estimation, we assumed that if a user is watching an asset without subtitles, he has strong knowledge of this language. On the other hand, if a user watches an asset with subtitles, he most likely needs it to understand the asset, and therefore the main language will be the one from the subtitle.

3.3. Recommendation Engine

The goal of the *Recommendation Engine* is to provide personalized recommendations based on the user-content interaction, the augmented features and the serendipity level. The engine is built using a combination of three approaches: content-based filtering, collaborative-based filtering, and clustering-user-based. A dedicated module weights and evaluates the algorithms, which fine-tune their parameters based on a feedback loop. Finally, the output is filtered to adjust the recommendation load suggested to the users. Below, we provide a detailed description of the algorithms, evaluation metrics, and filters adopted by the *Feature Engineering*.

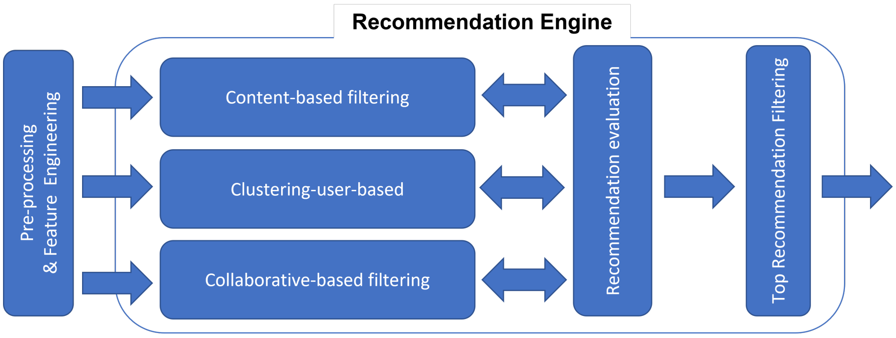


Figure 1: Recommendation Engine and its combination of content, clustering and collaborative filtering sub.components.

3.3.1. Algorithms

We first formalize the recommendation problem in order to formulate the different recommendation strategies. The fundamental concepts in this scope are *content items* (i.e., films, series, etc.) and *users*. We define the i-th content item as $C_i \forall i \in 1 \leq i \leq M$ and the j-th user as $U_j \forall j \in 1 \leq j \leq N$, with M and N the number of content items and users in the dataset, respectively. For each content item C_i a set of features X_i is related, where $X_i \in (x_i^{(0)}, x_i^{(1)}, .., x_i^{(Z)})$, with Z the number of item's features. Finally, we define the interaction of the j-th user with the i-th content defined as I_{ji}.

The *Content-based filtering* provides a list of content items based on the user-content interactions and measures the distance between the features vector of watched contents and the rest. The content items with the minimum distance are suggested to the user. In other words, given the j-th user,*Content-based filtering* is defined as:

$$\arg\min_X dist(X, \hat{X}) \quad \forall \quad X \notin I_j, \quad \hat{X} \in I_j, j \in U$$

where X are the feature vectors of the content not watched by the j-th user, $\hat{X}$ are the feature vectors of the content item watched by the j-th user, and $dist(X, \hat{X})$ is the distance (e.g., euclidean) between the two sets of feature vectors.

A user profile lists available categories with a score indicating the user's preference for each. By applying a dot product between each asset and the user profile, we generate a score for each asset, which enables us to create a ranking and provide personalized suggestions. As these suggestions are based on the user's preferences, the risk of rejection is low. However, they may lack serendipity. To overcome these limitations, the *Collaborative-based filtering* suggests the rest of the content items not watched by one user to the other, and vice versa. Given the i-th and j-th users, *Collaborative-based filtering* is defined as:

$$\arg\min_{I} dist(I_i, I_j) \quad \forall \quad i, j \in U$$

where I_i and I_j are the user-asset interactions of the i-th and j-th users.

The third method used by the recommendation engine is the *Clustering-user-based algorithm*. It starts by grouping users into clusters. Then it finds the closest clusters to the one of a particular user. Finally, it recommends to the user content items that are liked by other users belonging to the found cluster. Recommending items from a close cluster can increase the intercultural exchange and still ensure that the user may like the content because the distance between the two clusters is small. For each user, we define $\overrightarrow{w} = w_0, w_1, ..., w_Z$ its profile representing the weight accorded to the feature by the user. In other words, a feature appearing in most of the assets a user likes has a higher weight than a feature appearing in assets that a user dislikes. Given the i-th and j-th users, the *Clustering-user-based algorithm* is defined as:

$$\arg\min_{w} dist(w_i, w_j) \quad \forall \quad i, j \in U$$

where w_i and w_j are the user profile of the i-th and j-th users.

We adopt K-Means, DBScan, and Gaussian mixture algorithms to estimate the users'cohorts. In order to define the number of clusters needed for the K-Means algorithm, we used the Elbow method coupled with Silhouette graph computation.

3.4. Metrics and Filtering

The two remaining elements of our recommendation engine are the *Recommendation Evaluator* and *Recommendation Filter*. The first one evaluates the performance of the above-presented algorithms via metrics such as Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), Diversity, Novelty, HitRate and Serendipity. Moreover we also computed the coverage and catalog coverage [16]. Finally, *Recommendation Filtering* selects the top suggestions among the list of recommendations provided by the three algorithms to provide to the user.

4. Results

This section presents the results of the recommendation systems we developed. We begin by discussing our data exploration of the pre-processed dataset and the insights we gained from it. We then present the results obtained using different content rating estimations. After that, we illustrate the outputs of the Recommendation Engine. Finally, we compare the results using the metrics we previously introduced.

Table 1 presents five different scoring systems for our recommender engine. To evaluate these systems, we computed user profiles on different levels of granularity, including users with at least four views (the minimum), 10 views (the average number of views per user), 500 views (the maximum), and all users.

Table 2 shows the performance of our recommendation engine for the different scoring systems described in Table 1. The evaluation is based on several user-asset granularity levels.

Table 1
Scoring System used to compute user profile.

	System 1	System 2	System 3	System 4	System 5
Dislike	0	1	1	-1	-1
Neutral	0.5	1.5	2	1	0
Like	1	2	3	2	1

The table shows that the scoring system is a crucial component of our evaluation process that directly impacts the model's outcomes (i.e., user recommendations).

The evaluation is based on the mean of MAE and RMSE values, which are commonly used metrics to evaluate the accuracy of recommendation systems. The first three rows represent the evaluation results for users with the minimum, average, and maximum number of views, respectively. These values reflect the average deviation between the predicted ratings and the actual rating made by each user for each item. The last row of the table shows the overall evaluation results for all users in the dataset.

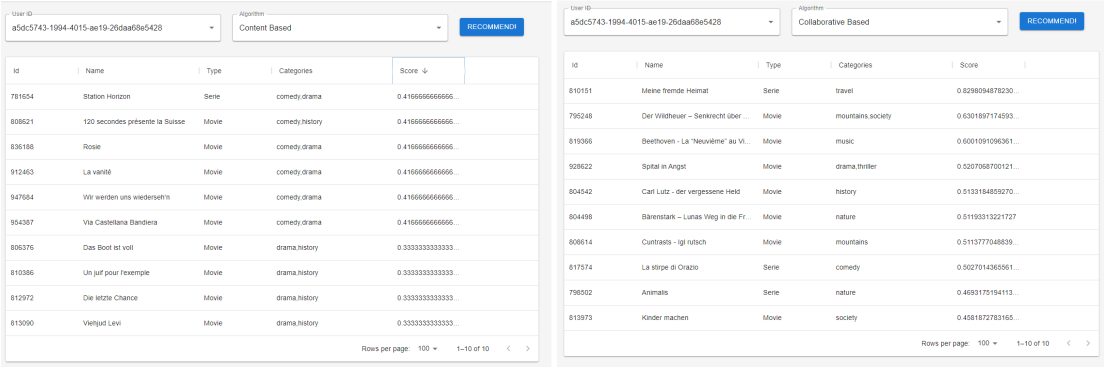
Table 2
Recommendation Engine of MAE/RMSE for each scoring system.

	System 1	System 2	System 3	System 4	System 5
4 Views (minimum)	0.58 / 1.97	0.7 / 1.93	0.68 / 1.90	1.12 / 2.33	1.02 / 2.67
10 Views (average)	0.52 / 1.71	0.24 / 0.61	0.12 / 0.34	0.4 / 1.44	0.96 / 3.12
500 Views (maximum)	10.32 / 17.26	30.08 / 51.19	41.24 / 69.87	15.92 / 26.33	4.84 / 7.77
Mean of all	0.57 / 2.17	0.65 / 2.47	0.68 / 2.59	0.71 / 2.69	0.74 / 2.83

Figure 2a is an example of the results we obtain from our recommendation engine when we fine-tune the evaluation towards a content-based approach. In a content-based approach, we recommend items that are similar in content to what the user has previously engaged with. For example, if a user has watched several action movies, we may recommend other action movies that share similar themes, actors, or directors. The figure shows the top 10 results generated by the recommendation engine using a content-based approach. These results are based on the user's previous engagement with assets and are sorted in descending order of their estimated rating. The top results are those that have the highest likelihood of being preferred by the user.

Figure 2b presents the top 10 results generated by our recommendation engine when the evaluation is fine-tuned towards a collaborative-based approach. In a collaborative-based approach, we recommend items based on the user's similarity to other users who have engaged with similar content. To generate these recommendations, we begin by calculating the correlation between the assets that each user has engaged with. We then use this correlation to identify other users who have engaged with similar content and calculate the correlation between these users and the current user. The result is a set of recommendations based on the engagement patterns of users with similar preferences. This approach is based on the hypothesis that users who have engaged with similar content are likely to have similar preferences. By identifying these patterns and leveraging them to generate recommendations, we can provide highly personalized and relevant suggestions to our users.

Figure 3 displays the results of applying the T-SNE algorithm to our dataset. Each data point



User ID: a5dc5743-1994-4015-ae19-26daa68e5428 | Algorithm: Content Based | RECOMMEND!

Id	Name	Type	Categories	Score
781654	Station Horizon	Serie	comedy,drama	0.4166666666666...
808621	120 secondes présente la Suisse	Movie	comedy,history	0.4166666666666...
836188	Rosie	Movie	comedy,drama	0.4166666666666...
912463	La vanité	Movie	comedy,drama	0.4166666666666...
947684	Wir werden uns wiederseh'n	Movie	comedy,drama	0.4166666666666...
954387	Via Castellana Bandiera	Movie	comedy,drama	0.4166666666666...
806376	Das Boot ist voll	Movie	drama,history	0.3333333333333...
810386	Un juif pour l'exemple	Movie	drama,history	0.3333333333333...
812972	Die letzte Chance	Movie	drama,history	0.3333333333333...
813090	Viehjud Levi	Movie	drama,history	0.3333333333333...

Rows per page: 100 1–10 of 10

User ID: a5dc5743-1994-4015-ae19-26daa68e5428 | Algorithm: Collaborative Based | RECOMMEND!

Id	Name	Type	Categories	Score
810151	Meine fremde Heimat	Serie	travel	0.8298094878230...
795248	Der Wildheuer – Senkrecht über ...	Movie	mountains,society	0.6301897174593...
819366	Beethoven - La "Neuvième" au Vi...	Movie	music	0.6001091096361...
928622	Spital in Angst	Movie	drama,thriller	0.5207068700121...
804542	Carl Lutz - der vergessene Held	Movie	history	0.5133184859270...
804498	Bärenstark – Lunas Weg in die Fr...	Movie	nature	0.51193313221727
808614	Cuntrasts - Igl rutsch	Movie	mountains	0.5113777048839...
817574	La stirpe di Orazio	Serie	comedy	0.5027014365561...
798502	Animalis	Serie	nature	0.4693175194113...
813973	Kinder machen	Movie	society	0.4581872783165...

Rows per page: 100 1–10 of 10

(a) content-based (b) collaborative-based

Figure 2: Top 10 recommendations for a user obtained with content- and collaborative-based filtering.

in the plot represents a user, and the T-SNE algorithm determines its position. To enhance our understanding of the user clusters, we color-coded each point based on the item categories that the user engaged with. By using T-SNE to reduce the dimensionality of our data, we can gain insights into the underlying structure of our user base. This information can be used to develop more targeted marketing strategies, personalize user experiences, and improve the performance of our recommendation engine. Table 3 illustrates the coverage and the catalog coverage, which represents the percentage of items that the recommender system is able to output and can effectively be recommended to a user, respectively [16].

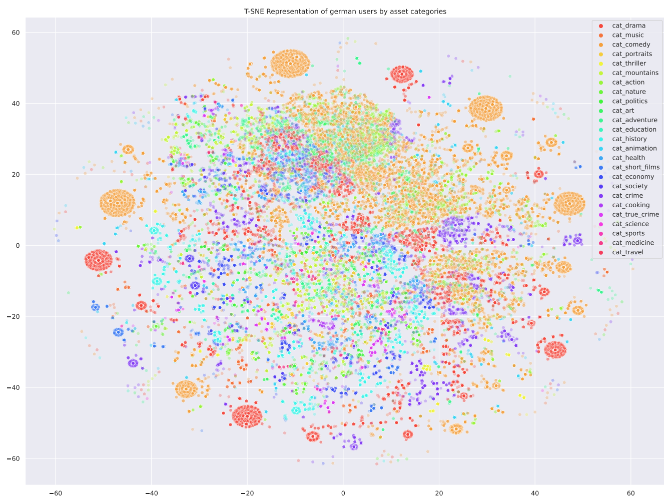


Figure 3: Result of T-SNE algorithm on the German speaker subset colored by most seen category.

Table 3
Coverage and Catalog Coverage for different algorithms.

	Random	Popular	Collaborative Filtering	Content Based	Cluster Based
Coverage	84.98	0.37	83.07	60.35	20.68
Catalog Coverage	20.05	0.37	19.72	8.85	10.17

5. Discussion

We developed a highly personalized recommendation system that is designed to enhance the diversity and serendipity of cultural content recommendations. Our system employs three algorithms that take into account users' preferences and interests, and requires minimal user information thanks to our cutting-edge feature engineering processes. Our recommendation engine enables users to discover a wide range of cultural content tailored to their users' profiles.

Table 2 demonstrates the effectiveness of our user rate estimation system, with System 1 exhibiting the highest overall performance. However, upon closer examination, we found that System 3 outperformed the other systems for users with a view count close to the average of ten user-asset interactions. This suggests that while System 1 may be the best choice for users with high levels of interaction, System 3 is better suited for users with more moderate levels of engagement.

Figure2a presents the results of fine-tuning the recommendation engine towards a content-based approach. We observed that, under this approach, the engine only recommends assets that correspond to the categories liked by the user (e.g., comedy and drama), which may not be optimal from a diversity and serendipity standpoint. Nonetheless, the novelty metric reveals that the engine still recommends assets with a moderate level of popularity, with an average rank of around 337. This suggests that while the engine may not provide the most diverse range of recommendations, it still manages to balance the user's preferences. In contrast, the collaborative-based algorithm recommends assets from a much broader range of categories. As a result, this approach has a significant potential to enhance the serendipity and diversity of recommendations. However, the novelty metric for this approach is relatively low, with a score of only 46, as demonstrated in Figure 2b. This suggests that while the collaborative-based algorithm may provide a more diverse range of recommendations, it may struggle to identify unique or novel content for users. Finally, by incorporating various filtering methods, as demonstrated in Figure 3, we achieved an MSE of 0.1625 and RMSE of 0.4032. These metrics indicate that our approach is highly effective at recommending assets from one cluster to another, thereby introducing users to a more diverse and serendipitous range of cultural content. As illustrated in Table 3, our engine was able to provide a wider range of engaging content to users, with an impressive coverage rate across the dataset.

While our personalized recommendation system has demonstrated promising results, its generalizability to other datasets may be limited. To confirm the system's broader applicability, further experimentation on diverse datasets will be conducted. Additionally, the system's effectiveness could be further improved by incorporating user feedback, which can help to fine-tune recommendations and better match individual user preferences and interests. In future work, we plan to incorporate user feedback to enhance the effectiveness and usability of the recommendation engine.

6. Conclusion

We presented a highly personalized recommendation system that is designed to enhance the diversity and serendipity of cultural content recommendations. Our approach leverages three different algorithms that consider users' interests and requires minimal user information thanks to our feature engineering techniques.

Our system enables users to discover a wide range of cultural content that is tailored to their unique tastes. Our results demonstrate the effectiveness of our approach. However, further experimentation on diverse datasets is required to confirm the system's broader applicability. Moreover, the system's effectiveness could be further improved by incorporating user feedback to fine-tune recommendations and better match individual user preferences.

Acknowledgements

This research was funded by the IMI Initiative for Media Innovation grant: Personalized Recommendation from Polarization to Discovery.

References

[1] E. Pariser, The filter bubble: What the Internet is hiding from you, Penguin, 2011.

[2] B. Alhijawi, A. Awajan, S. Fraihat, Survey on the objectives of recommender system: Measures, solutions, evaluation methodology, and new perspectives, ACM Comput. Surv. (2022). doi:10.1145/3527449.

[3] R. J. Ziarani, R. Ravanmehr, Serendipity in recommender systems: a systematic literature review, J. of Computer Science and Technology 36 (2021) 375–396.

[4] M. Ge, C. Delgado-Battenfeld, D. Jannach, Beyond accuracy: evaluating recommender systems by coverage and serendipity, in: Proceedings of the fourth ACM conference on Recommender systems, 2010, pp. 257–260.

[5] D. Kotkov, S. Wang, J. Veijalainen, A survey of serendipity in recommender systems, Knowledge-Based Systems 111 (2016) 180–192.

[6] M. De Gemmis, P. Lops, G. Semeraro, C. Musto, An investigation on the serendipity problem in recommender systems, Information Processing & Management 51 (2015) 695–717.

[7] V. Maccatrozzo, M. Terstall, L. Aroyo, G. Schreiber, Sirup: Serendipity in recommendations via user perceptions, in: Proceedings of the 22nd International Conference on Intelligent User Interfaces, 2017, pp. 35–44.

[8] X. Niu, F. Abbas, A framework for computational serendipity, in: Proc. of 25th Conference on User Modeling, Adaptation and Personalization, 2017, pp. 360–363.

[9] X. Li, W. Jiang, W. Chen, J. Wu, G. Wang, Haes: A new hybrid approach for movie recommendation with elastic serendipity, in: Proc. of the 28th ACM Intl. Conference on Information and Knowledge Management, 2019, pp. 1503–1512.

[10] M. M. Rahman, Contextual recommendation system, in: 2013 International Conference on Informatics, Electronics and Vision (ICIEV), 2013, pp. 1–6.
[11] H. Ito, T. Yoshikawa, T. Furuhashi, A study on improvement of serendipity in item-based collaborative filtering using association rule, in: 2014 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), IEEE, 2014, pp. 977–981.
[12] S. Kulkarni, S. F. Rodd, Context aware recommendation systems: A review of the state of the art techniques, Computer Science Review 37 (2020) 100255.
[13] S. Sekar, D. Dhanasekaran, G. Charlyn, C. Latha, Content-based movie recommendation system using mbo with dbn, Intelligent Automation and Soft Computing (2023).
[14] B. Ramzan, I. S. Bajwa, N. Jamil, R. U. Amin, S. Ramzan, F. Mirza, N. Sarwar, An intelligent data analysis for recommendation systems using machine learning, Scientific Programming 2019 (2019) 5941096. URL: https://doi.org/10.1155/2019/5941096. doi:10.1155/2019/5941096.
[15] S. V, L. R., M. Chandrashekhar, A. Challa, V. Varadarajan, A personalised movie recommendation system based on collaborative filtering, International Journal of High Performance Computing and Networking 10 (2017) 54. doi:10.1504/IJHPCN.2017.083199.
[16] M. Ge, C. Delgado-Battenfeld, D. Jannach, Beyond accuracy: Evaluating recommender systems by coverage and serendipity, in: Proceedings of the Fourth ACM Conference on Recommender Systems, RecSys '10, Association for Computing Machinery, New York, NY, USA, 2010, p. 257–260. URL: https://doi.org/10.1145/1864708.1864761. doi:10.1145/1864708.1864761.