A Smart Hybrid Enhanced Recommendation and Personalization Algorithm Using Machine Learning

Aswin Kumar Nalluri and Yan Zhang[a]

School of Computer Science and Engineering, California State University San Bernardino, 5500 University Parkway, San Bernardino, CA, 92407, U.S.A.

Keywords: Personalized Movie Recommendation, Hybrid Filtering, Content-Based Filtering, Term Frequency-Inverse Document Frequency, Collaborative Filtering, Alternating Least Squares.

Abstract: In today's era of streaming services, the effectiveness and precision of recommendation systems are pivotal in enhancing user satisfaction. Traditional recommendation systems often grapple with challenges such as data sparsity in user-item interactions, the need for parallel processing, and increased computational demands due to matrix densification, all of which hinder the overall efficiency and scalability of recommendation systems. To address these issues, we proposed the Smart Hybrid Enhanced Recommendation and Personalization Algorithm (SHERPA), a cutting-edge machine learning approach designed to revolutionize movie recommendations. SHERPA combines Term Frequency-Inverse Document Frequency (TF-IDF) for content-based filtering and Alternating Least Squares (ALS) with weighted regularization for collaborative filtering, offering a sophisticated method for delivering personalized suggestions. We evaluated the proposed SHERPA algorithm using a dataset of over 50 million ratings from 480,000 Netflix users, covering 17,000 movie titles. The performance of SHERPA was meticulously compared to traditional hybrid models, demonstrating a 70% improvement in prediction accuracy based on Root Mean Square Error (RMSE) metrics during the training, testing, and validation phases. These findings underscore SHERPA's ability to discern and cater to users' nuanced preferences, marking a significant advancement in personalized recommendation systems.

1 INTRODUCTION

In recent years, personalized recommendation systems have gained significant popularity due to the growing prevalence of online shopping platforms, social networks, and streaming services. Consider the last time you tried to choose a movie on a streaming site — it wasn't easy, was it? The challenge lies in the limitations of the engines behind those "Recommended for You" lists. These systems often rely on what you've already watched (collaborative filtering) (Ni et al., 2021) or suggest content based on genres you seem to prefer (content-based filtering) (Permana and Wibowo, 2023)(Philip et al., 2014). However, they frequently end up showing you more of the same, making it difficult to discover something new and exciting. This highlights the need for a smarter approach which truly understands your current mood by blending various advanced techniques from the world of machine learning, introducing you to content you'll genuinely enjoy.

In the competitive landscape of streaming platforms, the key to success hinges on engaging and delighting audiences. A crucial element in achieving this is providing movie recommendations that captivate viewers, almost like a touch of magic. Getting these recommendations right can increase user retention and encourage word-of-mouth promotion, which is vital in the ongoing streaming wars. It's not just about suggesting what an algorithm thinks you should watch; it's about understanding what viewers really want to see next, turning casual viewers into devoted fans eager to discover their next favorite movies.

This project introduces the Smart Hybrid Enhanced Recommendation and Personalization Algorithm (SHERPA) with the goal of revolutionizng movie recommendation processes. SHERPA combines collaborative filtering, content-based filtering, and advanced machine learning techniques to deliver tailored, accurate, and personalized content recommendations. Our goal is to simplify the movie discovery process by aligning recommendations with your preferences, not just based on what you've already

[a] https://orcid.org/0000-0002-5474-4019

Nalluri, A. and Zhang, Y.
A Smart Hybrid Enhanced Recommendation and Personalization Algorithm Using Machine Learning.
DOI: 10.5220/0013064100003838

In *Proceedings of the 16th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K 2024) - Volume 1: KDIR*, pages 465-472
ISBN: 978-989-758-716-0; ISSN: 2184-3228

seen. The focus is on creating a journey of content exploration that resonates with you, because, ultimately, every movie night should be about discovering something that truly hits the spot. SHERPA aims to eliminate the need for endless scrolling, ensuring that finding your next favorite movie is just a click away.

2 RELATED WORK

Traditional machine learning approaches in recommendation systems primarily focus on collaborative filtering and content-based filtering strategies (Ni et al., 2021). Collaborative filtering predicts user preferences by analyzing interactions and drawing insights from user behavior (Son and Kim, 2017). While this technique is widely used for its simplicity and effectiveness, it often faces challenges, particularly with new users (the cold start problem) and sparsity in user-item interactions (Wu et al., 2018)(Rahul et al., 2021).

Content-based filtering, on the other hand, suggests items based on their features and user preferences, emphasizing item metadata (Permana and Wibowo, 2023). However, this method may lead to a lack of diversity in recommendations, as it tends to suggest items similar to those the user has already interacted with (Philip et al., 2014).

Recent advancements in recommendation systems have made significant progress in overcoming these limitations. Techniques such as Singular Value Decomposition (SVD) have been employed to analyze user-item interactions and predict ratings by uncovering latent factors (Rahul et al., 2021). Additionally, new algorithms like Alternating Least Squares (ALS) with Weighted Regularization have enhanced collaborative filtering by prioritizing known interactions and incorporating regularization to prevent overfitting (SurvyanaWahyudi et al., 2017).

By combining these approaches, hybrid models that integrate elements of both content-based and collaborative filtering have been developed (Burke, 2002). These hybrid systems provide more comprehensive recommendations by considering both user behavior and content characteristics (Parthasarathy and Sathiya Devi, 2023). Zhou, et al. proposed an collaborative filtering algorithm Alternating-Least-Squares with Weighted-λ-regularization (ALS-WR), which is implemented on a parallel Matlab platform. They claimed that the performance of ALS-WR (in terms of root mean squared error (RMSE)) monotonically improves with both the number of features and the number of ALS iterations (Zhou et al., 2008). Chiny, et al. implemented a recommendation System based on TF-IDF and Cosine Similarity (Chiny et al., 2022). Hybrid systems not only improve the precision of recommendations but also offers a deeper understanding of user preferences and content relevance, paving the way for a new era in recommendation systems (Parthasarathy and Sathiya Devi, 2023).

3 DATASET AND PREPROCESSING

3.1 Dataset

The project involves two main datasets: the Movie Titles dataset and the Movie Ratings dataset, which are included in the Netflix Prize dataset posted on Kaggle (Netflix, 2006).

The Movie Titles dataset contains the information of 17,770 movies, with each movie represented as a tuple in the form: <Movie ID, Release Year, Movie Title, Director, Cast, Genre, Overview>. The original Movie Titles dataset contains Movie ID, Release Year, and Movie Title information of movies. We get extra information about these movies such as Director, Cast, Genre, Overview of Movie, from IMDB, an online database of information related to films, television series, etc.

This dataset provides a comprehensive overview of movies released from 1890 to 2005, with titles in English. The following is an examples of movie entries:

- Example: <1, 2003, Dinosaur Planet, Christian Slater, Scott Sampson, Animation, A four-episode animated series charting the adventures of four dinosaurs each on a different continent in the prehistoric world.>. This tuple shows that the movie ID is 1, the release year of this movie is 2003, the movie title is Dinosaur Planet, the director is Christian Slater, the cast is Scott Sampson, and the genre is Animation.

The Movie Ratings dataset comprises over 50 million ratings from 480,189 Netflix users, covering 17,770 movie titles, collected between October 1896 and December 2005. Each rating entry or instance contains User ID, Movie ID, Date of Rating, and Rating. Movie IDs are sequentially numbered from 1 to 17770. User IDs range from 1 to 2,649,429, with some numbers missing, representing a total of 480,189 users. Date of Ratings are consistently formatted as YYYY-MM-DD across all files. Ratings are on a five-star scale, ranging from 1 to 5 to show user opinion, where 5 represents the highest rating. To ensure customer privacy, unique customer IDs have been anonymized. The 50 million movie ratings dataset is splitted into three datasets. The training dataset contains total of $35,721,947$ ratings, the test dataset contains total of $7,654,704$ ratings, and the validation dataset contains total of $7,654,704$ ratings. The following is an examples of movie rating instances:

- Example: 1, 401047, 4, 2005-06-03
  This example shows that the user with ID 401047 rated the movie with ID 1 as 4 stars on June 3, 2005.

3.2 Data Preprocessing

During the data preprocessing stage, we structured unprocessed data to align with the machine learning model's format for effective learning. This involved parsing data from a file, extracting movie IDs, customer IDs, and ratings, and structuring them into a list. We converted this list into a pandas DataFrame for easier manipulation and handled format issues by skipping lines that didn't match the expected format. Additionally, we cleaned the data by replacing any NaN values with empty strings, preparing it for further analysis.

4 METHODOLOGIES

Recommendation systems use filtering algorithms to provide recommendations to users. These algorithms are classified or categorized majorly into collaborative-based filtering, content- based filtering, and hybrid algorithms. The proposed Smart Hybrid Enhanced Recommendation and Personaliza- tion Algorithm (SHERPA) integrates Term Frequency-Inverse Document Frequency (TF-IDF) for content-based filtering and Alternating Least Squares (ALS) with weighted regularization for collaborative filtering, offering a sophisticated method for delivering personalized suggestions.

4.1 Term Frequency-Inverse Document Frequency (TF-IDF)

Term Frequency-Inverse Document Frequency (TF-IDF) is a statistical measure used to evaluate how important a word in a document within a collection of texts known as a corpus (Rajaraman and Ullman, 2011). It is often used in text mining and information retrieval to weight and evaluate words differently based on their importance to a document relative to a collection. Words that are frequent in one document but less common across others receive a TF-IDF value suggesting they could be crucial, for comprehending the content of that document (Chiny et al., 2022).

Term Frequency (TF) is the number of times a term appears in a document relative to the total word count of that document. TF is calculated using Equation 1 as follows (Rajaraman and Ullman, 2011):

$$tf(t,\ d)\ =\ \frac{N_{t,d}}{N_d}, \tag{1}$$

where $N_{t,d}$ represents the number of times that term t occurs in document d, and N_d represents the total number of terms in the document d.

Inverse Document Frequency (IDF) measures the rarity of a term across all documents. IDF is calculated using Equation 2 as follows (Rajaraman and Ullman, 2011):

$$idf(t,\ D)\ =\ log\frac{N}{|d\ \in\ D : t\ \in\ d|}, \tag{2}$$

where N is the total number of documents in the collection in the corpus $N = |D|$; $|d\ \in\ D : t\ \in\ d|$ is the number of documents where the term t appears.

By combining Equation 1 and Equation 2, The TF-IDF score for term t in document d is calculated as follows:

$$tfidf(t,d,D) = tf(t,d) \times\ idf(t,D) \tag{3}$$

Words with high TF-IDF scores in a document are used more in that document and less in others, making them key indicators of what the document is about.

4.2 Singular Value Decomposition (SVD)

Singular Value Decomposition (SVD) is a matrix decomposition method that allows you to approximate a matrix as a product of 3 matrices (Kadhim et al., 2017). This process allows us to uncover connections in the data. For example, when we have information about how users rated items such as movies, but not every user rates every item, SVD comes in to complete the missing information (Widiyaningtyas et al., 2022). The SVD of an $m \times n$ complex matrix M is a factorization of the form

$$M = U \times \sum V^T, \tag{4}$$

where M is the original user item rating matrix, U is the matrix where each row represents a user in terms of latent factors, Σ is a diagonal matrix with singular values that indicate the importance of each latent factor, V^T is the transpose of a matrix where each column represents an item in terms of latent factors.

4.3 Alternating Least Squares (ALS)

Alternating Least Squares (ALS) is a technique that handles sparse data by optimizing matrix factorization process by breaking it down into two smaller or more manageable subproblems (Takács and Tikk, 2012). Unlike Singular Value Decomposition (SVD), which considers all entries in the user-item interaction matrix (including unknown or missing values), ALS focuses only on the known ratings and it scales well for large datasets and integrates regularization directly to prevent overfitting, making it ideal for collaborative filtering (Pilászy et al., 2010).

ALS with Weighted-λ-Regularization is an enhancement to the standard ALS approach. It introduces a regularization term to the optimization process, which helps to avoid overfitting a common problem where a model performs well on the training data but poorly on unseen data. The goal of ALS with Weighted-λ-Regularization is to find user and item feature matrices that predict how users would rate items, even new or previously unrated ones (Zhou et al., 2008).

The effectiveness of this method is measured by a loss function that captures two things (Zhou et al., 2008):

- How well the model predicts the known ratings.
- How complex the model is (the size of the user and item feature matrices).

The loss function is represented mathematically as:

$$
\begin{aligned}
f(U,M) = & \sum_{(i,j)\in I} (r_{ij} - u_i^T m_j)^2 \\
& + \lambda \left(\sum_i n_{u_i} \|u_i\|^2 + \sum_j n_{m_j} \|m_j\|^2 \right),
\end{aligned} \tag{5}
$$

where r_{ij} is the actual rating of item j by user i, u_i is the feature vector representing user i, m_j is the feature vector representing item j, I is the set of all (user, item) pairs for which the rating is known, λ is the regularization weight that controls the trade-off between fitting the training data well and keeping the model simple to avoid overfitting, n_{u_i} is the number of items rated by user i, which weighs the user's feature vector, n_{m_j} is the number of users who have rated item j, which weighs the item's feature vector.

Loss function with efficient weighted regularization controls the complexity of the model and prevents overfitting by penalizing large values of the user and item feature vectors.

ALS with Weighted-λ-Regularization is highly suitable for large-scale datasets because of its ability to efficiently handle sparse user-item matrices by focusing on observed interactions, reducing memory requirements, and allowing for parallel computation.

4.4 Content-Based Filtering

Content-Based Filtering is a method used by recommendation systems to suggest items to users based on the characteristics of the items themselves rather than on the user's interaction with other users (Van Meteren and Van Someren, 2000). This method uses item features (like overview, genre, director, cast in movies) to recommend items similar to what the user has liked and positively rated in the past (Philip et al., 2014).

Several algorithms are commonly used in content-based recommendation systems. TF-IDF is chosen over traditional techniques because it provides a more sophisticated way to evaluate the importance of words (or terms) in the content (Van Meteren and Van Someren, 2000). Unlike simple frequency counts, TF-IDF accounts for the rarity of terms across all documents, thus giving higher weight to terms that are unique to a particular item (Permana and Wibowo, 2023). This is crucial in differentiating items with similar but not identical content, as common terms do not overly influence the similarity score.

4.5 Collaborative Based Filtering

Collaborative filtering functions, as a recommendation system algorithm, forecasts a user's preferences by considering the preferences of users (Hameed et al., 2012). It operates on the premise that if users A and B share viewpoints on an item, it is probable that A will align with B's perspective on another item that A has not yet encountered (Wu et al., 2018) (Konstan and Riedl, 2012). By analyzing user item interactions like ratings or viewing history, the algorithm detects patterns and resemblances among users or items (Ni et al., 2021) (Goyani and Chaurasiya, 2020). This approach enables tailored recommendations by tapping into the preferences of the user community, making it widely adopted in suggesting movies, music, and various products. Figure 1 illustrates the mechanisms of collaborative and content-based filtering techniques. Collaborative filtering recommends items by identifying patterns among similar users, while content-based filtering suggests items based on their similarity to content previously liked by the user.

4.6 Hybrid Filtering

A Hybrid filtering algorithm enhances recommendation systems by merging collaborative and content-

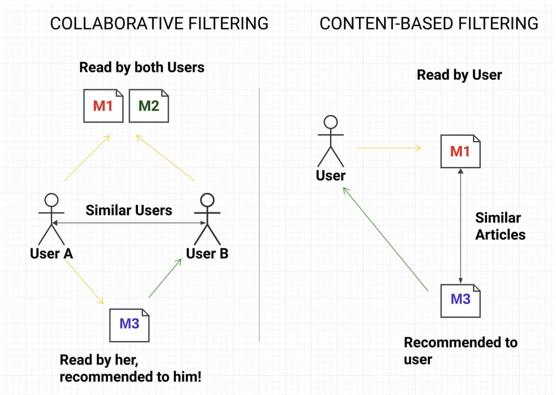

Figure 1: Comparison of collaborative and content-based filterings.

based filtering strategies leveraging the strengths of each to compensate for their shortcomings (Goyani and Chaurasiya, 2020)(Sharma et al., 2022). This strategy integrates the Singular Value Decomposition (SVD) technique, which forecasts user preferences based on patterns, in user item interactions with TF-IDF which examines item content to gauge its significance (Burke, 2002)(Thorat et al., 2015). By merging the personalized forecasts of SVD and the content specificity of TF-IDF, the hybrid model provides varied and thorough recommendations effectively tackling issues, like the cold start dilemma and enhancing recommendation accuracy (Parthasarathy and Sathiya Devi, 2023).

4.7 SHERPA

The proposed Smart Hybrid Enhanced Recommendation and Personalization Algorithm (SHERPA) is a recommendation system that intelligently combines the strengths of two methods: Alternating Least Squares (ALS) with Weighted Regularization for collaborative filtering, and Term Frequency-Inverse Document Frequency (TF-IDF) for content-based filtering, as shown in Figure 2.

By utilizing ALS with Weighted-λ-Regularization, SHERPA focuses on implicit data like known ratings and handles sparse data by optimizing matrix factorization process with loss function to avoid overfitting problem by computing independently user and item matrices across multiple processors or nodes in a cluster. At the same time, the incorporation of TF-IDF allows SHERPA works on explicit data by assigning weights ('Overview' - 45%, 'Genre' - 25%,'Director' - 15%, 'Cast' - 15%) to movie attributes based on their importance in a document. This weighting scheme helps identify the most distinctive and relevant terms for each

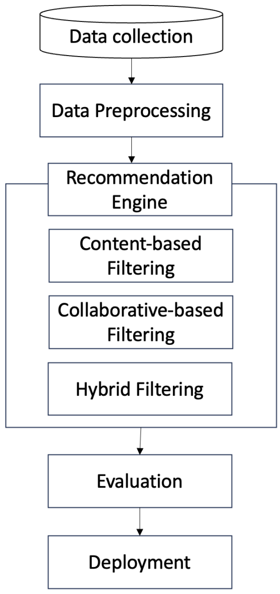

Figure 2: SHERPA Recommendation System Architecture.

document and transforms text-based movie attributes into numerical vectors. This vectorization allows the system to quantify and compare movie characteristics mathematically.

This dual strategy working on both implicit and explicit data enables SHERPA to effectively handles large datasets, supports scalability and parallelization. it addresses the limitations of traditional methods to deliver more relevant recommendation and enhancing user satisfaction.

5 EXPERIMENT AND EVALUATION

To demonstrate the capabilities of the proposed SHERPA algorithm, we implemented a series of experiments. In the experimental setup, a dual-core processor and at least 2 GB of RAM are essential for general system operation. For the computationally intensive tasks of training and test, a GPU with a minimum of 2 GB of VRAM is necessary. Examples of suitable GPUs include the NVIDIA GTX 1050 or higher-end models.

5.1 Evaluation Metric

Root Mean Square Error (RMSE) is a standard way to measure the error of a model in predicting quantitative data (Hyndman and Koehler, 2006). It's particularly useful in recommender systems to evaluate the difference between predicted and actual ratings. RMSE provides a way to quantify the magnitude of prediction errors, taking the square root of the average squared differences between the prediction and the actual observation. The formula of RMSE is:

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(p_i - a_i)^2}, \tag{6}$$

where p_i represents the predicted value for the ith instance, a_i is the actual value for the ith instance, N is the total number of instances.

A lower RMSE value indicates a better fit of the model to the data. It's especially effective in highlighting the impact of large errors, given that it squares the differences before averaging. However, it should be noted that RMSE can be sensitive to outliers and might not be well-suited if the error distribution is not uniform.

In the context of our paper, RMSE will serve as a key indicator of the accuracy of our recommendation system's predictions, allowing us to fine-tune the algorithm for optimal performance.

5.2 Evaluation Scenarios

We have designed two distinct scenarios to evaluate the performance of the SHERPA algorithm. One is designed for the existing users and the other is for new users. These scenarios are constructed to evaluate the system's responsiveness to each user's unique needs whether they're browsing casually or conducting specific searches based on their past interactions.

5.2.1 For Existing Users

For existing users, we designed two different scenarios to evaluate the proposed algorithm. One is to recommend movies to existing users who log in but do not conduct any search; the other is to recommend movies to existing users who log in and search a key word.

Existing User Log in and Without Search. When an existing user logs in without conducting any searching, the system uses their interactions to recommend movies. Since the user is simply browsing, collaborative filtering is used. This involves the algorithm analyzing the activities of users, with interests and suggesting movies that those users have enjoyed.

The following are the recommendation results from Hybrid and SHERPA approaches, the top 10 movies for existing user id = 401047 and without search keyword:

HYBRID Results:

1. Unknown Pleasures
2. The Swindle
3. Saint Sinner
4. Lone Wolf and Cub: Baby Cart in Peril
5. Die Hard 2: Die Harder
6. Seems Like Old Times
7. Kati Patang
8. Korn: Deuce
9. Hocus Pocus
10. The Usual Suspects

SHERPA Results:

1. Mel Gibson's Passion of the Christ
2. The Best of Friends: Vol. 4
3. Stargate SG 1: Season 7
4. The Winds of War
5. Stargate SG 1: Season 8
6. Friends: Season 6
7. Alias: Season 3
8. 24: Season 1
9. CSI: Season 3
10. Shania Twain: Up Close and Personal

Existing User Log in and Search with Keyword. When an existing user logs in and searches for a term like "The Company", the system transitions to the recommendation method. It combines the user's data with the search query to suggest options that cater not only to popular choices or similar users but also to results directly related to the search term.

The following are the recommendation results from Hybrid and SHERPA approaches, the top 10 movies for existing user id = 401047 and with search keyword "The Company":

HYBRID Results:

1. Center Stage
2. Ballet Favorites
3. Expo: Magic of the White City
4. A Raisin in the Sun
5. Robin and the 7 Hoods
6. Unknown Pleasures
7. Out of Sync
8. Orchestra Rehearsal
9. Category 6: Day of Destruction
10. The Usual Suspects

SHERPA Results:

1. Center Stage
2. Ballet Favorites
3. Expo: Magic of the White City
4. A Raisin in the Sun
5. Robin and the 7 Hoods
6. Swan Lake: Tchaikovsky (Matthew Bourne)
7. Out of Sync
8. Orchestra Rehearsal
9. Category 6: Day of Destruction
10. What Have I Done to Deserve This?

5.2.2 For New Users

New User Log in and Search with Keyword: When a new user looks up a term like "The Company" without any viewing history, the algorithm uses content-based filtering. This approach analyzes factors such as genre, storyline, and actors of the movie to suggest movies with similar content to "The Company". The aim is to provide tailored recommendations based solely on the search query.

The following are the recommendation results from Hybrid and SHERPA approaches, the top 10 movies for new user and with search keyword "The Company":

HYBRID Results:

1. Center Stage
2. Ballet Favorites
3. Expo: Magic of the White City
4. A Raisin in the Sun
5. Robin and the 7 Hoods
6. Unknown Pleasures
7. Out of Sync
8. Orchestra Rehearsal
9. Category 6: Day of Destruction
10. The Usual Suspects

SHERPA Results:

1. Center Stage
2. Ballet Favorites
3. Expo: Magic of the White City
4. A Raisin in the Sun
5. Robin and the 7 Hoods
6. Swan Lake: Tchaikovsky (Matthew Bourne)
7. Out of Sync
8. Orchestra Rehearsal
9. Category 6: Day of Destruction
10. What Have I Done to Deserve This?

5.3 Results

In this Section, we compare the SHERPA algorithm's performance against traditional hybrid systems using Root Mean Square Error (RMSE) metric across the training, test, and validation datasets as detailed below:

Table 1: The comparison of Hybrid and SHERPA algorithms.

Models	Training	Test	Validation
Hybrid	2.8289	2.9487	2.9492
SHERPA	0.8606	0.9039	0.9041
Improvement	69.6%	69.4%	69.3%

The comparison of the Hybrid and SHERPA algorithms across training, test, and validation datasets reveals significant differences in their performance. In the training dataset, the Hybrid model shows an RMSE of 2.8289, indicating some challenges in understanding user preferences, while SHERPA impressively reduces this to 0.8606, marking a substantial 69.6% improvement. Moving to the test dataset, Hybrid exhibits an RMSE of 2.9487, suggesting occasional inaccuracies, whereas SHERPA achieves a more reliable RMSE of 0.9039, a 69.4% enhancement. In the validation dataset, Hybrid scores 2.9492 in RMSE, highlighting room for improvement, whereas SHERPA excels with an RMSE of 0.9041, showcasing consistent and reliable performance.

SHERPA's success is attributed to its advanced matrix factorization technique, weighted-λ-regularization, parallelization for scalability, computational efficiency, hybrid filtering approach, and continuous learning, which collectively result in a 70% improvement over traditional Hybrid algorithms. SHERPA's balanced approach ensures both technical superiority and a more personalized recommendation experience for users.

6 CONCLUSION

This paper introduced Smart Hybrid Enhanced Recommendation and Personalization Algorithm (SHERPA), an advanced machine learning algorithm created to enhance and personalize the movie recommendation process. By combining content-based filtering using TF-IDF and collaborative filtering through ALS with Weighted Regularization, SHERPA has shown an improvement in recommendation accuracy and user satisfaction.

Through analysis using metrics like RMSE, SHERPAs performance compared to traditional hybrid models was highlighted. Notably SHERPA achieved a decrease in prediction errors with enhancements of around 70% across training, testing and validation datasets when compared to its predecessor. This emphasizes the algorithms improved capability to comprehend and forecast user preferences providing relevant content suggestions. Moreover, SHERPA's innovative methodology tackles issues seen in existing recommendation systems such as overfitting and addressing the cold start problem. This ensures a scalable solution that caters to user interactions. Its proficiency in managing datasets and customizing content based on user behaviors as well as item traits sets a new standard in recommendation system technology.

In summary, the SHERPA algorithm signifies a progression in recommendation systems. The users content discovery experience is enhanced by SHERPA, which also paves the way for advancements in machine learning and artificial intelligence research and development. In the changing world personalized recommendation systems like SHERPA play a crucial role in driving future innovations.

ACKNOWLEDGMENTS

This project is based upon work supported by the U.S. National Science Foundation under Grant No. 2142503.

REFERENCES

Burke, R. (2002). Hybrid recommender systems: Survey and experiments. *User Modeling and User-adapted Interaction*, 12:331–370.

Chiny, M., Chihab, M., Bencharef, O., and Chihab, Y. (2022). Netflix recommendation system based on tf-idf and cosine similarity algorithms. In *Proceedings of the 2nd International Conference on Big Data, Modelling and Machine Learning*, pages 15–20.

Goyani, M. and Chaurasiya, N. (2020). A review of movie recommendation system: Limitations, survey and challenges. *Electronic Letters on Computer Vision and Image Analysis*, 19(3):0018–37.

Hameed, M. A., Al Jadaan, O., and Ramachandram, S. (2012). Collaborative filtering based recommendation system: A survey. *International Journal on Computer Science and Engineering*, 4(5):859.

Hyndman, R. J. and Koehler, A. B. (2006). Another look at measures of forecast accuracy. *International journal of forecasting*, 22(4):679–688.

Kadhim, A. I., Cheah, Y.-N., Hieder, I. A., and Ali, R. A. (2017). Improving tf-idf with singular value decomposition (svd) for feature extraction on twitter. In *3rd international engineering conference on developments in civil and computer engineering applications*.

Konstan, J. A. and Riedl, J. (2012). Recommender systems: from algorithms to user experience. *User Modeling and User-adapted Interaction*, 22:101–123.

Netflix (2006). Netflix prize data on kaggle.com. Accessed: 2024-09-06.

Ni, J., Cai, Y., Tang, G., and Xie, Y. (2021). Collaborative filtering recommendation algorithm based on tf-idf and user characteristics. *Applied Sciences*, 11(20):9554.

Parthasarathy, G. and Sathiya Devi, S. (2023). Hybrid recommendation system based on collaborative and content-based filtering. *Cybernetics and Systems*, 54(4):432–453.

Permana, A. H. J. P. J. and Wibowo, A. T. (2023). Movie recommendation system based on synopsis using content-based filtering with tf-idf and cosine similarity. *International Journal on Information and Communication Technology*, 9(2):1–14.

Philip, S., Shola, P., and Ovye, A. (2014). Application of content-based approach in research paper recommendation system for a digital library. *International Journal of Advanced Computer Science and Applications*, 5(10).

Pilászy, I., Zibriczky, D., and Tikk, D. (2010). Fast als-based matrix factorization for explicit and implicit feedback datasets. In *Proceedings of the 4th ACM conference on Recommender systems*, pages 71–78.

Rahul, M., Kumar, V., and Yadav, V. (2021). Movie recommender system using single value decomposition and k-means clustering. In *IOP Conference Series Materials Science and Engineering*, volume 1022. IOP Publishing.

Rajaraman, A. and Ullman, J. D. (2011). *Mining of massive datasets*. Autoedicion.

Sharma, S., Rana, V., and Malhotra, M. (2022). Automatic recommendation system based on hybrid filtering algorithm. *Education and Information Technologies*, 27(2):1523–1538.

Son, J. and Kim, S. B. (2017). Content-based filtering for recommendation systems using multiattribute networks. *Expert Systems with Applications*, 89:404–412.

SurvyanaWahyudi, I., Affandi, A., and Hariadi, M. (2017). Recommender engine using cosine similarity based on alternating least square-weight regularization. In *International Conference on Quality in Research (QiR): International Symposium on Electrical and Computer Engineering*, pages 256–261. IEEE.

Takács, G. and Tikk, D. (2012). Alternating least squares for personalized ranking. In *Proceedings of the sixth ACM conference on Recommender systems*, pages 83–90.

Thorat, P. B., Goudar, R. M., and Barve, S. (2015). Survey on collaborative filtering, content-based filtering and hybrid recommendation system. *International Journal of Computer Applications*, 110(4):31–36.

Van Meteren, R. and Van Someren, M. (2000). Using content-based filtering for recommendation. In *Proceedings of the machine learning in the new information age: MLnet/ECML2000 workshop*, volume 30, pages 47–56. Barcelona.

Widiyaningtyas, T., Ardiansyah, M. I., and Adji, T. B. (2022). Recommendation algorithm using svd and weight point rank (svd-wpr). *Big Data and Cognitive Computing*, 6(4):121.

Wu, C. S. M., Garg, D., and Bhandary, U. (2018). Movie recommendation system using collaborative filtering. In *International Conference on Software Engineering and Service Science (ICSESS)*, pages 11–15. IEEE.

Zhou, Y. H., Wilkinson, D., Schreiber, R., and Pan, R. (2008). Large-scale parallel collaborative filtering for the netflix prize. In *The 4th International Conference on Algorithmic Aspects in Information and Management*, pages 337–348. Springer.