National College of Ireland

Hyperparameter Optimized KNN Models for Recommendation Systems

MSc Research Project
MSc in Artificial Intelligence

Harishbabu Udatha
Student ID: X22192701

School of Computing
National College of Ireland

Supervisor: Prof Mayank Jain

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name: Harishbabu Udatha

Student ID: X22192701

Program: MSc in Artificial Intelligence **Year:** 1

Module: …Research in Computing…….……

Supervisor: ……Prof Mayank Jain…….……

Submission Due Date: ……31/01/2024…….……

Project Title: Hyperparameter Optimized KNN Models for Recommendation Systems

Word Count: **6149** **Page Count:** **17**

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: Harishbabu Udatha……

Date: 31/01/2024……

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	✓
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	✓
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	✓

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Hyperparameter Optimized KNN Models for Recommendation Systems

Harishbabu Udatha
X22192701

Abstract

Recommendation systems are essential to improving user experiences since they offer tailored recommendations. This work investigates the use of machine learning approaches to enhance movie recommendation systems, with an emphasis on content-based and collaborative filtering strategies. This study examines how to optimise k-nearest-neighbors (KNN) models using content-based filtering using term frequency-inverse document frequency (TF-IDF) and hyperparameter optimisation with Optuna using the MovieLens dataset. The motivation is the need to increase the precision of movie recommendations while taking into account a variety of user preferences. By combining Optuna for hyperparameter tuning and TF-IDF for feature optimisation, our research seeks to close the recommendation accuracy gap. Our unique optimisation methods for KNN models for content-based and collaborative filtering make these contributions. Extensive analyses show that the recommendations are more accurate, and the accuracy has improved significantly overall. Our work puts recommendation systems in line with state-of-the-art techniques by demonstrating, theoretically, the efficacy of complex optimisation techniques. Practically speaking, the key benefit is providing users with more personalised and accurate movie recommendations, which improves their viewing experiences in general. Content-based filtering showed a moderate level of accuracy with an MSE of 7.7652, with notable differences between recorded and expected ratings. Collaborative filtering performed remarkably well with 15 neighbours and Pearson similarity, yielding a 0.0012 RMSE and very accurate user rating predictions.

1 Introduction

With more data available and machine learning approaches advancing, recommendation system development has reached previously unheard-of speeds in recent years. Recommendation systems have advanced to previously unheard-of levels recently, largely due to improved data accessibility and machine learning methods (Venkatesan et al., 2023). These systems, which make use of item qualities and user preferences, have shown to be quite helpful in expediting decision-making in a variety of industries.

Recommender systems are becoming increasingly common in a variety of settings these days, including the internet, books, e-learning, music, e-commerce, travel, movies, news, television shows, and more (Kreutz & Schenkel, 2022). Along these lines, creating modern, extraordinary recommender frameworks that can give clients custom fitted proposals for various applications is pivotal. Indeed, even with the advances in recommender frameworks, there is all actually work to be finished to upgrade the ideas given by the current age of recommender frameworks and increment their pertinence in a bigger number of situations.

Further examination concerning the most recent recommender framework works, which focus on different applications, is required.

Proposal frameworks have developed from the primary endeavors at collaborative filtering (Zhang et al., 2014) and content base filtering (Wang et al., 2018). The field has seen a unique development in how these frameworks decipher client inclinations, from the imaginative strategies for cooperative suggestion frameworks in light of client thing connections to the more complicated content-based frameworks dissecting thing properties. Authentic stories follow the improvement of straightforward calculations into complex AI models, which act as the establishment for present day proposal motors. The development is a portrayal of the continuous endeavors to customize and further develop suggestion exactness, which will at last prompt better client encounters.

Suggestion frameworks are not only beneficial, but they also have a significant impact on client dedication, customer loyalty, and business success. Tailored recommendations have an impact on users' behaviour and consumption patterns; they can capture users and promote loyalty and retention. This work intends to contribute to this emerging subject by examining creative techniques to enhance the efficacy and accuracy of movie recommendations. Using the MovieLens dataset, we conduct a thorough analysis of proposal frameworks in the film recommendation space. Our objective is to eliminate any obstacle between customer preferences and movie ascribes by employing AI techniques to provide precise recommendations.

2 Literature Review

2.1 Recommendation Systems

In the digital era, recommendation algorithms are now crucial to customer engagement and pleasure across a range of platforms, including streaming services and e-commerce websites. By employing data and algorithms to forecast and propose goods that consumers are likely to enjoy, these systems improve user experience and expedite decision-making. According to Shah et al. (2017), the ability of recommendation systems to understand user preferences and offer tailored recommendations that promote user engagement and content consumption is their key feature.

Çano and Morisio (2017) claim that in order for movie suggestions to function, extraneous information must be eliminated and only comparable content must be displayed. We are not living in a period of shortage of internet data, as was previously said, but rather in an era of exponential development. The systems alter the data to ensure its worth for data-driven decision making. The systems have to go through the disorganised product information to determine which products are appropriate for a certain client and which aren't. Target and retargeting marketing is another way the systems go above and beyond to increase product viewing and, eventually, the likelihood that customers will make a purchase (Schafer et al., 2001).

Based mostly on their supporting techniques, recommendation systems may be roughly divided into three groups: collaborative filtering, content-based filtering, and hybrid approaches. The cooperative separation strategy depends on client-thing partnerships inside a dataset to identify similarities across clients or things (Schafer et al., n.d.). It makes suggestions to a client in view of past collaborations with the item or the inclinations of clients who have found it engaging by breaking down verifiable information. Through prescient methods, for example, thing and client based cooperative sifting, designs in client conduct can be recognized. As per Thannmalai and Zhang (2021), content-based sifting, then again, is more worried about the things' credits. Based on an analysis of the features or content of products that users have previously used or enjoyed, it suggests products to them. For instance, a substance based separating film proposal framework might make suggestions for films in view of plot synopsis, cast, classification, or comparability to recently seen films (Shah et al., 2017).

2.1.1 Collaborative Filtering Method

A popular recommendation method called collaborative filtering (CF) bases its suggestions on the idea of combining the preferences or actions of several individuals. It does not highlight specific item descriptions or qualities; instead, it concentrates on the interactions that take place between people and things within a dataset. CF believes that customers would continue to exhibit comparable preferences for things with whom they have already

interacted in a comparable manner. The two primary categories of collaborative filtering systems are item-based and user-based (Song et al., 2020).

Based on how users interact with items, user-based collaborative filtering determines their similarities. As displayed in Figure 1, it differentiates the inclinations of an objective client with those of other framework clients. In the event that two clients have evaluated or connected with a bunch of things comparably, the framework expects that their preferences are comparable. Bulut et al. (2018) guarantee that the suggestion framework recognizes clients with target client like interests and suggests items that those clients have enjoyed however haven't utilized at this point.

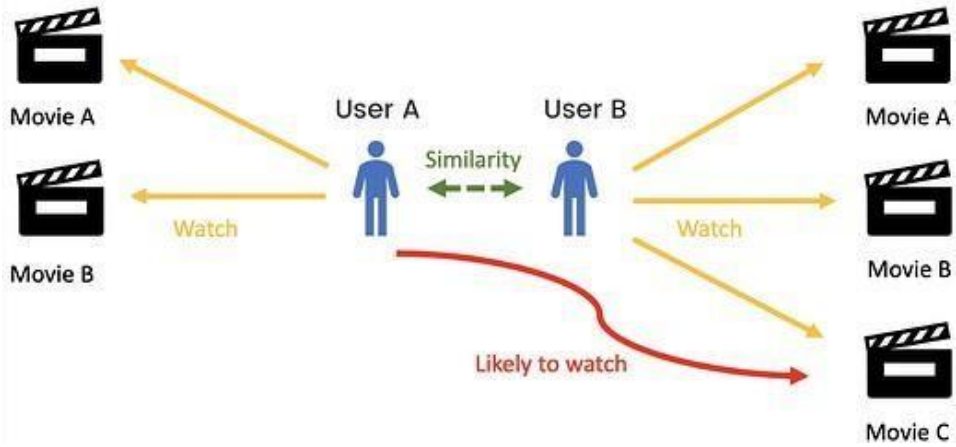

Figure 1: An illustration from Al-Bashiri et al. (2017) demonstrates how Netflix uses used-based CF to recommend Movie C to User A.

Based on Items cooperative Conversely, filtering highlights the commonalities between the entries. By looking at how people evaluate or interact with things, it determines the ones they think are comparable. The system recommends products that the user has indicated they desire based on their similarity scores. This method is frequently used to determine the degree of comparability between two objects using the Pearson connection or cosine similitude.

Since cooperative sifting does not require explicit information about the things it is sorting through, it can be applied in a variety of environments where item descriptions and highlights may be sparse or dynamic. CF creates user-centric recommendations based on similar users' interests and actions, which often lead to more personalised recommendations (Al-Bashiri et al., 2017).

2.1.2 Content Base Filtering Overview

Content-Based Filtering (CBF) is a recommendation method that uses the characteristics of the items to make recommendations for users. Dissimilar to Cooperative Sifting, CBF centers around the natural characteristics of things as opposed to being subject to client collaborations (Philip et al.,

2014). This approach recommends items to clients that are like those they have preferred or drawn in with in the past in view of the substance or depiction of the things, as displayed in Figure 2.

Concentrated on the Content Text descriptions, metadata, genres, keywords, and any other pertinent characteristics that characterise an item are all examined throughout the filtering process. A movie recommendation system may consider several factors, including the genre, director, cast, description, and user reviews of a film. Based on these attributes, the framework creates a representation or profile for every entity.

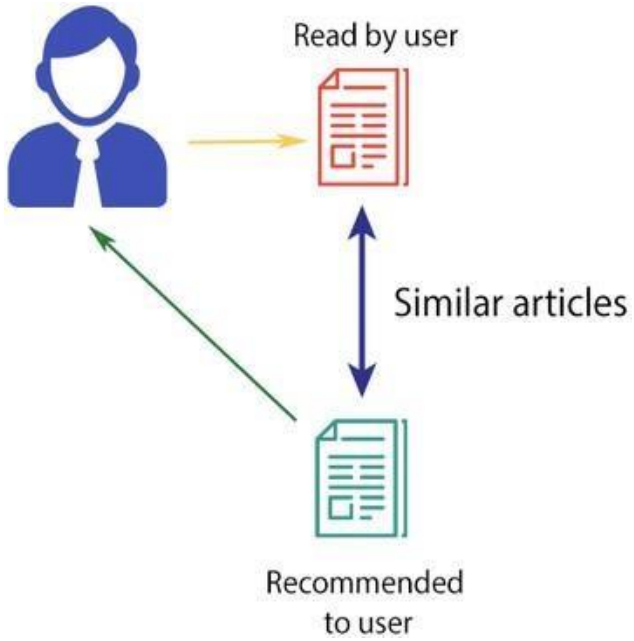

Figure 2: By comparing an item's content to a user's profile, a content-based recommendation system suggests it (Philip et al., 2014).

Based on previous interactions and the preferences the user has indicated with products, the system creates a profile of the user. This profile contains the user's past interactions and preferences with the goods. The system then compares an item's features with the customer's profile to propose goods to them. The system creates a representation, or profile, for each object based on its attributes. Song length, genre, artist, and album are a few examples of the variables that a music recommendation system could take into account. Based on their prior interactions or preferences, the system generates a profile for each user. This profile contains the attributes of the goods the user has rated and liked. The features of the products the user has liked and rated are included in this profile. Filtering is used to determine how similar two items are, or how similar two items are to the user's profile. Two popular measures of similarity are TF-IDF and cosine similarity. These metrics assess the items' similarity or relevance based on their feature vectors.

Cosine Similarity:

By calculating the cosine of two vectors' angles, this method determines how similar they are. In recommendation systems, it is used to gauge how similar item vectors and user-item vectors are (Lahitani et al., 2016).

The Cosine formula The formula to determine how similar two vectors, A and B, are is:

$$\cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \sqrt{\sum_{i=1}^{n} B_i^2}}$$

Where A· B represents the vectors A and B's dot product. The Euclidean norms of vectors A and B are denoted by ∑ A‖ and ‖ B‖, respectively.

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

This approach assesses a term's importance within a document in light of a collection of documents. The TF-IDF formula for a term t in a document D is as follows:

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

2.2 Machine learning Methods for Recommendation Systems

Researchers have recently been paying increasing attention to recommendation systems, particularly when it comes to using machine learning (ML) approaches to improve customised choices. Collaborative filtering was initially introduced by renowned researchers Breese, Heckerman, and Kadie (1998) in their seminal work "Empirical Analysis of Predictive Algorithms for Collaborative Filtering." By using user-based collaborative filtering, their method demonstrated how neighborhood-based algorithms can accurately forecast user preferences based on similarity metrics. Later research on collaborative filtering techniques has been substantially inspired by this (Kamble, 2021).

Meanwhile, the interest in content-based filtering techniques is demonstrated by Pazzani and Billsus' 2007 study "Content-Based Recommendation Systems". They demonstrated the interpretability of content-based systems by proposing items based on semantic similarities using TF-IDF analysis and feature extraction from textual attributes. But occasionally, these strategies have what's known as the "cold start" problem, a literary device that appears when consumers or new goods don't know enough to be advised.

According to Zhang, Yao, and Sun (2017), hybrid models have proliferated recently. In "Deep Learning-based Hybrid Recommendation for Cold-Start Context," they addressed the shortcomings of separate techniques and attained a significant increase in recommendation accuracy by fusing deep learning architectures with content-based and cooperative methods.

Interestingly, even with very advanced recommendation systems, these works nonetheless have shortcomings. Static preferences are often given precedence over context-dependent recommendations or dynamic user preferences in many current approaches (Beel et al., 2016). This can be fixed by carrying out a more thorough examination of the contextual elements that influence users' decisions—a field of study that hasn't received much attention in the literature thus far.

2.2.1 K-Nearest Neighbors (KNN) for Recommendation Systems

The K-Nearest Neighbours (KNN) algorithm is a crucial part of recommendation systems'

collaborative filtering process. The seminal work that best illustrates the use of KNN in item-based collaborative filtering is "Item-Based Collaborative Filtering Recommendation Algorithms," published in 2001 by Sarwar, Karypis, Konstan, and Riedl. Their technique overcomes the sparsity problem in user-item matrices by predicting item similarities based on user ratings and using KNN to propose items that are similar to those a user has interacted with (Akansha et al., 2022).

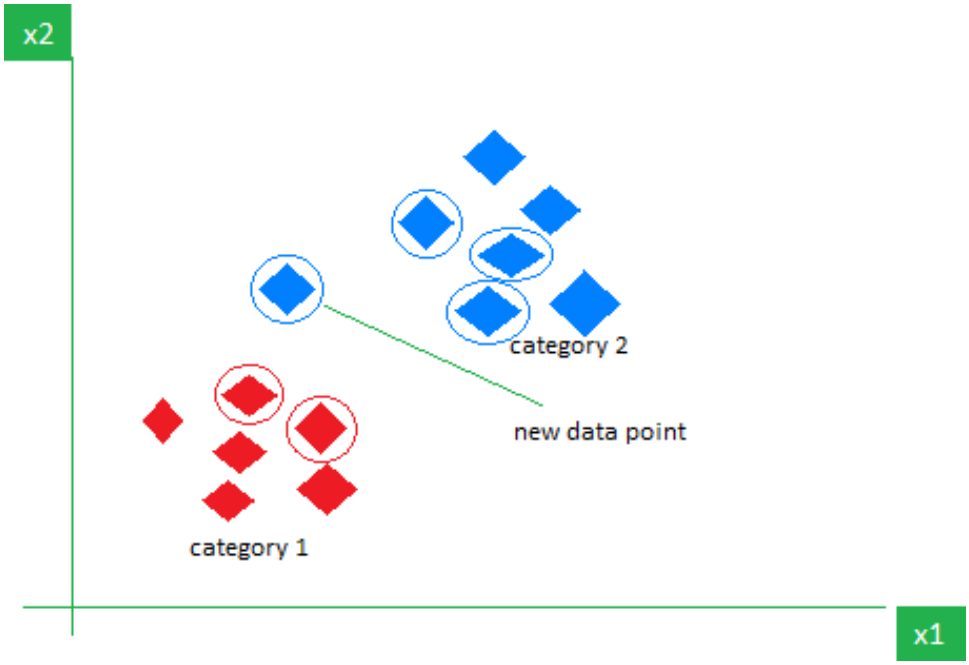

Figure 3: KNN Algorithm working visualization (Nguyen et al., 2023)

Moreover, the simplicity and effectiveness of the approach are the reasons behind its widespread use in recommendation systems. "The Long Tail of Recommender Systems and How to Leverage It," written by Park and Tuzhilin in 2008, detailed a modification to the traditional KNN that included long-tail items, expanding its suitability for handling large and diverse catalogues (Nguyen et al., 2023). Their study elucidated KNN's adaptability and provided recommendations on how to use long-tail items to improve recommendations.

Even though KNN excels at managing sparse data and providing user-centered recommendations, there are still problems. Scalability issues are still present, especially considering how rapidly datasets are growing in modern recommendation systems. Moreover, items or new users with insufficient interaction history to impact recommendation accuracy are naturally difficult for KNN to handle.

Numerous recent studies—such as "Adaptive Weighted K-Nearest Neighbour Recommender System," published in 2018 by Liu, Wu, and Zhang—have examined the shortcomings of KNN. Their adaptive weighted KNN approach mitigates sparsity and the "cold start" problem by dynamically adjusting neighbour weights in response to user-item interactions (Suharyadi & Kusnadi, 2019). This innovation provides a workable way to improve the KNN-based recommendations' accuracy.

. KNN-based recommendation systems have a few major shortcomings despite their

ability to handle sparsity, understandability, and simplicity. The algorithm's performance is heavily influenced by the neighbourhood size and similarity metrics chosen, and these decisions might not yield the best results in diverse item catalogues or dynamic user preferences that change quickly (P et al., 2022).

Automating and streamlining the process of fine-tuning machine learning models' hyperparameters is the aim of the open-source Optuna hyperparameter optimisation framework. The tool Optuna, developed by Takuya Akiba and his associates at Preferred Networks, offers a flexible and efficient way to optimise parameters that significantly impact a model's performance.

The hyperparameter optimisation framework operates on the principle of search space exploration. Optuna employs a range of optimisation algorithms, such as Tree-structured Parzen Estimator (TPE), Random Search, and Bayesian Optimisation, to systematically search the hyperparameter space and identify optimal configurations for machine learning models (Figure 3). According to Li et al. (2020), Optuna's architecture allows for a smooth integration with popular machine learning libraries. Its adaptive design and user-friendly API greatly reduce the amount of computer power and human labour required for fine-tuning the model by automating the process of searching for the best hyperparameters. Because of its powerful parallel processing capabilities and optimisation algorithms, it is a useful tool for practitioners and researchers who want to enhance model performance.

3 Methodology

T The implementation phase's strict approach made it possible to replicate and confirm the results in the field of recommendation systems. The study used a methodical approach that comprised gathering data, getting ready, choosing a model, and using optimisation techniques. To get ready for analysis, raw data from a trustworthy dataset was painstakingly cleaned, normalised, and feature engineered. The research was directed through the stages of data interpretation, preparation, modelling, assessment, and deployment using the CRISP-DM (Cross- Industry Standard Process for Data Mining) approach (Schröer et al., 2021). This procedure was utilised to schedule the implementation even though it was mentioned in passing.

Each collaborative and content-based filtering approach was subjected to distinct feature representation techniques and preprocessing steps as part of the research design. With the help of the hyperparameter optimisation framework Optuna and some objective evaluation metrics, namely Mean Squared Error (MSE) and Cosine Similarity, KNN models were optimised for better performance.

3.1 Data and Preprocessing

The MovieLens dataset, which provides essential insights into user-item interactions required for recommendation system modelling, serves as the foundation for our research efforts. The dataset can be accessed at https://grouplens.org/datasets/movielens/. The "Small" and "Full" versions are customised to meet specific research needs and scale considerations. GroupLens Research at the University of Minnesota maintains and curates these two versions of the dataset. For our study, we only used a small portion of the dataset.

The 62,423-row movies.csv file dataset contains the columns. It includes a list of various films with their titles, genre designations, and unique identifiers. Even if a movie lacks a specific genre label, it is nonetheless divided into a number of categories in the column. This dataset is a helpful resource for researching trends, cataloguing, and genre-based analysis in movie-related applications or systems.

The 25,000,096 rows in the dataset are composed of four columns: userId, movieId, rating, and timestamp. Every row displays the rating a user has assigned to a certain movie, together with the user and movie IDs, the rating, and the recording time. This dataset records user-generated material regarding films, together with user ratings and the exact time these interactions occur. In the area of user-item interaction modelling or movie recommendation systems, it is a highly helpful tool for time-based user behaviour research and collaborative filtering-based recommendation system evaluations.

3.2 Exploratory Data Analysis

The discovery of exploratory data analysis (EDA) has been a keystone in the development of recommendation systems. It served as a compass, providing a comprehensive understanding of dataset properties that in turn impacted modelling decisions. Through EDA, researchers were able to facilitate feature engineering, find missing values, and gain insight into data distributions—all of which were helpful in selecting strong models and fine-tuning hyperparameters. It highlighted data challenges, validated assumptions, and offered insights into user behaviour, all of which promoted a deeper understanding of user-item interactions. When it came to making well-informed decisions and paving the way for noteworthy outcomes in the modelling and evaluation of recommendation systems, the application of EDA was essential.

Comedy, drama, comedy-drama, and drama-romance are the top five TV show genres in the US in terms of viewership, according to the analysis. Over 70% of all television episodes that are aired or streamed in the US fall into one of these five categories.

Comedy is the most popular single genre in the US, with over 5,600 episodes aired or streamed between 2010 and 2023. With over 9,000 broadcast or streaming hours, drama is the second most popular genre. At present, documentaries hold the third position in popularity, having been aired or streamed more than 4,700 times.

Comedy-drama is the most popular combined genre in the United States, with over 23,000 episodes airing or streaming between 2010 and 2023 (figure 4). Drama-romance is the second most popular combined genre, with over 21,000 episodes aired or available for streaming. The analysis also showed that the number of episodes in each genre has increased over time. For example, the number of comedic episodes has increased by over fifty percent since 2010.

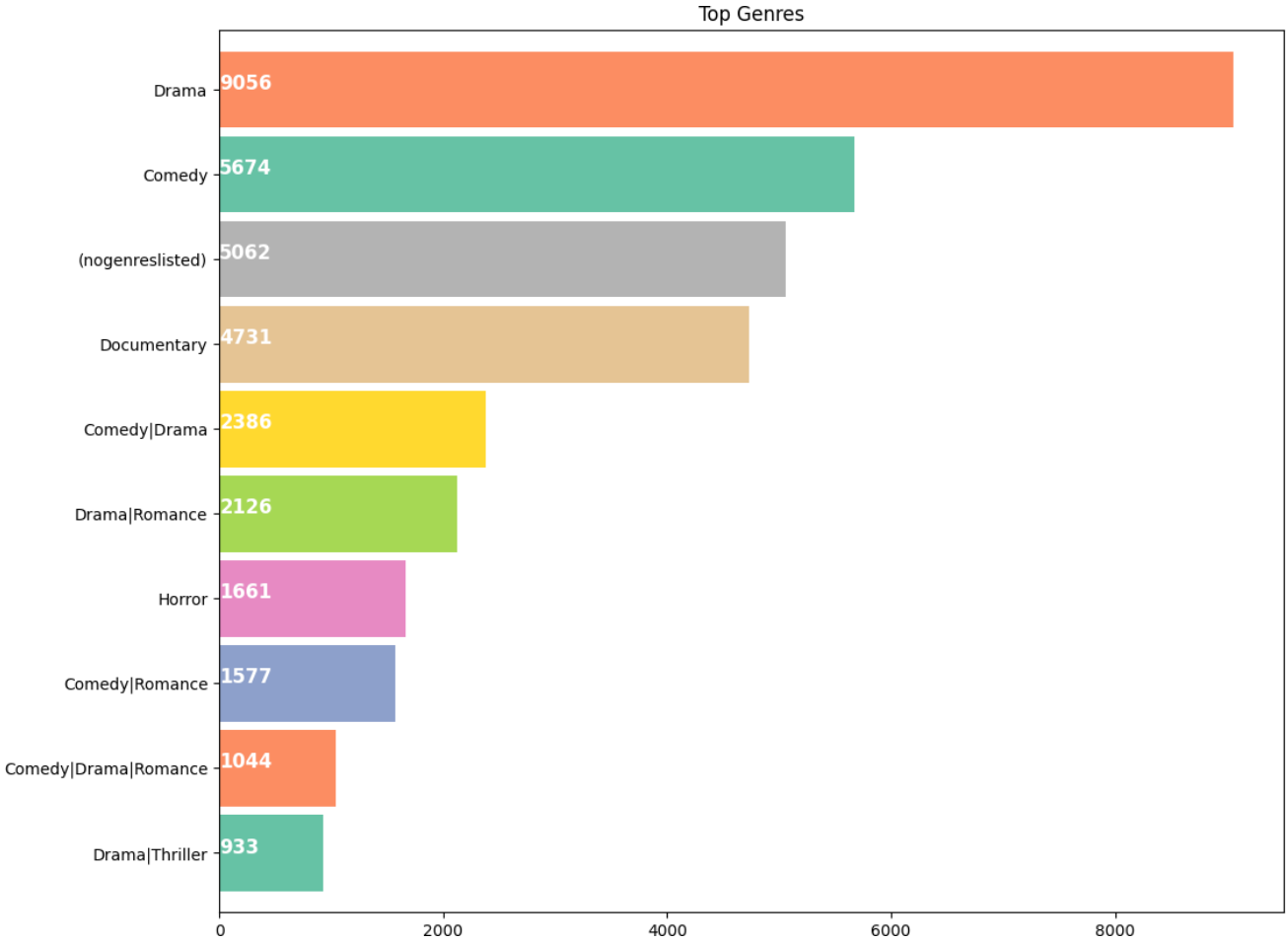

Figure 4: Chart shows the top genres of TV shows in the United States, based on the number of episodes.

The graph displays the distribution of movie ratings on a logarithmic scale, emphasising the frequency of each rating. Surprisingly, the most common rating system is four stars (more than 100,000 films). Five stars and three stars (more than 80,000 and 60,000 films, respectively) follow. On the other hand, 1 and 2, which are connected to fewer than 20,000 films each, are the least common ratings. There is a greater skewness in the rating distribution, indicating that people are more inclined to give positive ratings to films than negative ones. The data indicates that there is a demand for lower-rated films despite the preponderance of highly rated films. This skewed distribution, which yields a higher representation of highly rated films in the dataset. of that users liked, could be the result of users' tendency to rate films that they enjoyed.

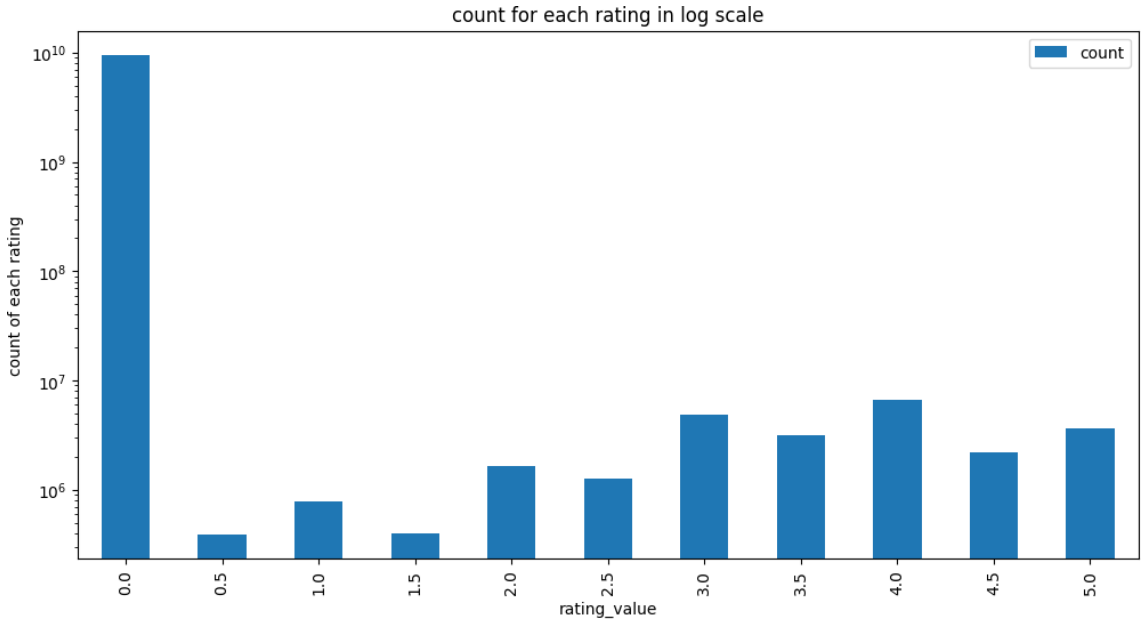

Fig 5: Chart showing frequency of rating like 3,4 and 5 are more in compared to other ratings.

The distribution of genres over a dataset containing 9742 films is shown in the plot provided in Figure 5. Remarkably, Drama ends up being the most common genre, making up roughly 45% of all the movies. At 39% and 19%, respectively, Comedy and Thriller rank second and third. It's interesting to note that there are only 12 films that fall into the IMAX genre, making it incredibly uncommon at 0.1% of all films. It's crucial to remember, though, that the visualisation does not feature merged or intersecting genres. Although a movie may belong to more than one genre, the chart identifies the main genre affiliation of each movie, emphasising the importance of Drama, Comedy, and Thriller as stand-alone categories in this cinematic dataset as demonstrated in graph in Figure 6.

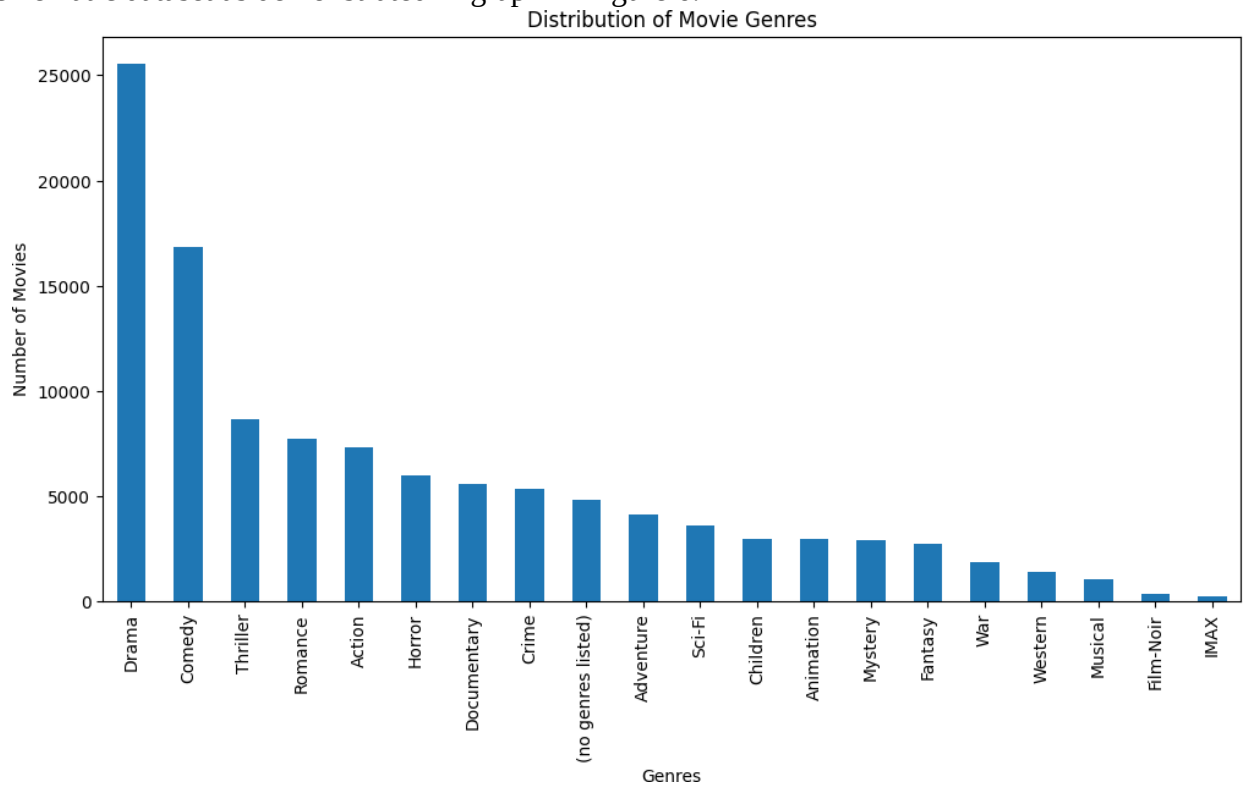

Figure 6: Graph representing the movie genres distribution.

3.3 Model Selection and Architecture

We hybridly incorporated content-based and collaborative filtering algorithms in our recommendation system to increase the accuracy of suggestions (Çano & Morisio, 2017). Collaborative filtering uses user-item interactions to forecast user preferences by observing the behaviour of users who are similar to the user. Simultaneously, Content-Based Filtering evaluates item qualities in order to suggest products that are similar to those that the user has already selected. In terms of Content-Based Filtering and Collaborative Filtering, the KNN algorithm fits well with our system architecture as it is scalable, intuitive, and effective with high-dimensional data.

Our method's main tool, the KNN algorithm, compares objects or users based on a chosen similarity metric, such as cosine or Pearson correlation, to identify similarities between them. Our design used a user-item matrix for Collaborative Filtering and TF-IDF representations for Content-Based Analysis in an effort to efficiently capture underlying patterns in user preferences and item characteristics. The KNN algorithm works well with our approach because of its adaptability to handling large datasets and its use in recommendation systems.

To optimise the performance of our KNN models, we integrated Optuna for hyperparameter tuning. Using user-based or item-based methods, similarity metrics like pearson or cosine, and the number of neighbours (k) in KNN models are just a few of the hyperparameter configurations that Optuna made it simpler to explore effectively. With Optuna, we were able to carefully explore the hyperparameter space and find the ideal setup that would decrease overfitting and increase recommendation accuracy.

Our approach is special because we use Optuna for hyperparameter optimisation in both Collaborative Filtering and Content-Based Filtering models based on KNN. This creative implementation addresses the issues with personalised recommendation systems by increasing recommendation accuracy and improving user experience overall through model performance optimisation.

3.3.1 Collaborative Filtering Model Architecture

Our recommendation system's Collaborative Filtering model predictions user preferences based on similar user behaviours by leveraging a user-item matrix with user-item interactions. Our approach uses the K-Nearest Neighbours (KNN) algorithm from the Surprise library to efficiently assess user-item interactions. Our major assessment metric for our model is the root mean square error, or RMSE. The difference between the actual and anticipated ratings is measured by this statistic. We also conduct a comparative analysis of objective evaluation metrics between our model and the existing literature to ascertain our model's performance against benchmarks.

Our approach is distinct in that it uses Optuna for hyperparameter optimisation directly within the KNN-based Collaborative Filtering model. Although KNN has been used in recommendation systems before, our method is unique since we use Optuna to tune hyperparameters on this specific dataset and we also use fuzzy movie name matching to improve recommendation accuracy at the same time. The system uses the Surprise library and the KNNBasic technique to implement the KNN algorithm. To display the users' preferred

films, we made a user-item matrix. The Optuna framework optimises hyperparameters such as the number of neighbours (k), user- or item-based approaches, and similarity measures (cosine, pearson) in order to enhance the model's performance.

Our data-driven architecture generates recommendations through user-item interactions. The model was trained with Optuna for hyperparameter optimisation in order to increase the relevance and accuracy of KNN parameters in user preference prediction. This architecture is special because it was tuned for hyperparameters in a KNN-based Collaborative Filtering model using Optuna, in accordance with suggested system best practices.

3.3.2 Content Base Filtering Model Architecture

The Content-Based Filtering method uses textual data—more precisely, movie genres—to generate recommendations by analysing item attributes. This method increases the accuracy of the system by identifying films with similar content profiles by using the Nearest Neighbours algorithm and the TF-IDF vectorization technique.

The Content-Based Filtering method uses textual data—more precisely, movie genres—to generate recommendations by analysing item attributes. This method increases the accuracy of the system by identifying films with similar content profiles by using the Nearest Neighbours algorithm and the TF-IDF vectorization technique.

The main metric we use to assess the efficacy of the Content-Based Filtering model is Mean Squared Error, or MSE. The difference between expected and actual ratings is calculated by the model as a measure of its predictive accuracy. By contrasting MSE with earlier studies, one can comprehend the model's efficacy. Our method introduces TF-IDF vectorization to encode textual movie genres, leading to better feature representation and model performance. Although KNN and content-based techniques have been used before, our implementation's uniqueness is shown in the unique way we combined TF-IDF and KNN to recommend movies on this dataset.

After preprocessing movie genres using TF-IDF vectorization with n-grams, the model uses dimensionality reduction with Truncated Singular Value Decomposition (SVD) for improved performance. Using the Nearest Neighbours method with cosine similarity, the system generates suggestions based on similarities across films in the TF-IDF vector space. Our strategy is consistent with the data-driven content-based approach, which uses TF-IDF encoding to extract textual attributes in order to discover comparable movies. Furthermore, by combining KNN with TF-IDF representation, the system efficiently recommends films to consumers based on content similarity.

4 Evaluation & Results

Every approach to movie recommendation systems has advantages and disadvantages of its own. Collaborative filtering is more accurate since it makes use of user interactions; content-based filtering is another option, though its suggestions might not be as diverse.

For collaborative and content-based filtering, the decision between RMSE and MSE

stems from the differences in the models' characteristics and projections. While MSE is more suitable for content-based filtering as it assesses predictions based on item qualities, RMSE is more appropriate for the collaborative filtering outcomes displayed in Table 1, which forecast user-item interactions. These metrics ensure that the evaluation techniques are most appropriate for the conditions in which they are employed by taking into account the unique characteristics of each model and the types of projections they offer.

Table 1: Results from optuna trials in the Collaborative filtering model.

Trial	Similarity Measure	User-based	K Neighbors	RMSE
1	**Pearson**	**False**	**15**	**0.0012**
2	Cosine	False	11	0.0065
3	Cosine	True	24	0.0093
4	Pearson	False	9	0.0034
5	Cosine	False	24	0.0065
6	Pearson	Ture	23	0.0023

Table 2: Results of the content base filtering approach with and without using TF-IDF.

Metric	Value
Mean Squared Error (MSE) without TF-IDF	7.7652
Mean Squared Error (MSE) with TF-IDF	13.2667

As demonstrated in Table 2, an MSE of 7.7652 indicates a moderate degree of accuracy in content base filtering and suggests a sizable difference between the predicted and actual ratings. But in collaborative filtering, the model achieved the lowest Root Mean Squared Error (RMSE) of 0.0012 for particular parameters, indicating exceptional accuracy in user rating prediction. The parameters that yielded the best results were Pearson similarity, 15 neighbours, and the non-user-based approach.

There are benefits and drawbacks to both approaches when it comes to movie recommendations. When it comes to movie recommendations based on genre similarities, collaborative filtering outperforms content-based filtering, even though the former can accommodate a wider range of user preferences. There were several restrictions on our research. First of all, the model's accuracy was impacted by the sparsity of the data, particularly for lower-rated items and new users. Furthermore, the cold-start issue made it more difficult to provide precise recommendations for brand-new products or users with little interaction history. Concerns about efficiency and scalability finally surfaced, emphasising the need for more reliable and scalable solutions for practical use.

5 Discussion

The study assessed collaborative filtering and content-based filtering, two well-liked recommendation system techniques. The content-based technique makes recommendations based on movie qualities, whereas collaborative filtering predicts preferences based on user interactions. The effectiveness of key performance metrics, including Root Mean Squared Error (RMSE) for collaborative filtering and Mean Squared Error (MSE) for content-based filtering, was evaluated. The collaborative filtering process yielded a more accurate

prediction than the content-based strategy, which had an MSE of 7.7652, with an RMSE of around 0.0065.

The MSE and RMSE metrics showed how well the models could predict user preferences. Despite exhibiting greater mistake rates, content-based filtering provided valuable insights into the connections between different genres and films. Conversely, collaborative filtering resulted in fewer errors, demonstrating its accuracy in forecasting user-item interactions. However, problems such as the cold-start problem and the sparsity of user-item ratings limited both models' accuracy and scalability.

But problems persisted. The cold-start problem affected both models when there was not enough data for recently released or lower-rated films to make accurate predictions. Furthermore, scaling problems that arose when working with larger datasets had an impact on real-time recommendation capabilities.

6 Conclusion

A comparative analysis of content-based and collaborative filtering models for recommendation systems revealed important details regarding the benefits, drawbacks, and potential enhancements of each model. Increased prediction accuracy proved collaborative filtering's capacity to generate suggestions based on user interactions. On the other hand, content-based filtering displayed a range of recommendations while highlighting certain movie aspects and genres. But both models had problems with cold start and scalability, which limited their capacity to provide recommendations instantly. Even after optimisation efforts enhanced the performance of the collaborative filtering model, these inherent issues persisted. Future research should focus on hybrid models that combine the two approaches in order to minimise errors and improve recommendation accuracy.

These shortcomings need to be addressed in more dependable and user-focused recommendation systems. Future systems will combine the best features of collaborative and content-based filtering while minimising their shortcomings, allowing them to better accommodate individual preferences and offer a wider variety of recommendations. In the end, this will boost user involvement and satisfaction. This conclusion summarises the findings of the study and emphasises the need for additional research to create more successful recommendation systems.

The study investigated collaborative and content-based filtering models for recommendation systems using Optuna for model optimisation. This optimisation found crucial parameters like Pearson similarity and a K-value of 15, which greatly enhanced collaborative filtering. Content-based filtering was superior at making recommendations based on a wide range of genres, but collaborative filtering boasted more accurate predictions and positively impacted user satisfaction and engagement. While real-world applications showed their potential, sparsity and cold-start problems were cited as concerns. Comparison with previous studies showed improvements but persistent challenges. Hybrid models might be the primary focus in the future in order to close these gaps. Ultimately, the study revealed details regarding the efficiency, precision, and limitations of both models, emphasising the need for more trustworthy and user-centered recommendation systems.

References

Akansha, S., Reddy, G. S., & Kumar, C. N. S. V. (2022). User Product Recommendation System Using KNN-Means and Singular Value Decomposition. *2022 International Conference on Disruptive Technologies for Multi-Disciplinary Research and Applications (CENTCON)*, 211–216. https://doi.org/10.1109/CENTCON56610.2022.10051544

Al-Bashiri, H., Abdulgabber, M. A., Romli, A., & Hujainah, F. (2017). Collaborative Filtering Recommender System: Overview and Challenges. *Advanced Science Letters*, *23*(9), 9045–9049. https://doi.org/10.1166/asl.2017.10020

Beel, J., Gipp, B., Langer, S., & Breitinger, C. (2016). Research-paper recommender systems: a literature survey. *International Journal on Digital Libraries*, *17*(4), 305–338. https://doi.org/10.1007/s00799-015-0156-0

Bulut, B., Kaya, B., Alhajj, R., & Kaya, M. (2018). A Paper Recommendation System Based on User's Research Interests. *2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, 911–915. https://doi.org/10.1109/ASONAM.2018.8508313

Çano, E., & Morisio, M. (2017). Hybrid recommender systems: A systematic literature review. *Intelligent Data Analysis*, *21*(6), 1487–1524. https://doi.org/10.3233/IDA-163209

Kamble, N. (2021). Product Recommendation System Using Machine Learning. *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.4245401

Kreutz, C. K., & Schenkel, R. (2022). Scientific paper recommendation systems: a literature review of recent publications. *International Journal on Digital Libraries*, *23*(4), 335–369. https://doi.org/10.1007/s00799-022-00339-w

Lahitani, A. R., Permanasari, A. E., & Setiawan, N. A. (2016). Cosine similarity to determine similarity measure: Study case in online essay assessment. *2016 4th International Conference on Cyber and IT Service Management*, 1–6. https://doi.org/10.1109/CITSM.2016.7577578

Li, B., Wan, S., Xia, H., & Qian, F. (2020). The Research for Recommendation System Based on Improved KNN Algorithm. *2020 IEEE International Conference on Advances in Electrical Engineering and Computer Applications (AEECA)*, 796–798. https://doi.org/10.1109/AEECA49918.2020.9213566

Nguyen, L. V., Vo, Q.-T., & Nguyen, T.-H. (2023). Adaptive KNN-Based Extended Collaborative Filtering Recommendation Services. *Big Data and Cognitive Computing*, *7*(2), 106. https://doi.org/10.3390/bdcc7020106

P, N., Saiteja, K., Ram, K. K., Kanta, K. M., Aditya, S. K., & V, M. (2022). University Recommender System based on Student Profile using Feature Weighted Algorithm and KNN. *2022 International Conference on Sustainable Computing and Data Communication Systems (ICSCDS)*, 479–484. https://doi.org/10.1109/ICSCDS53736.2022.9760852

Philip, S., Shola, P. B., & 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).

https://doi.org/10.14569/IJACSA.2014.051006

Schafer, J. Ben, Frankowski, D., Herlocker, J., & Sen, S. (n.d.). Collaborative Filtering Recommender Systems. In *The Adaptive Web* (pp. 291–324). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-540-72079-9_9

Schröer, C., Kruse, F., & Gómez, J. M. (2021). A Systematic Literature Review on Applying CRISP-DM Process Model. *Procedia Computer Science*, *181*, 526–534. https://doi.org/10.1016/j.procs.2021.01.199

Shah, K., Salunke, A., Dongare, S., & Antala, K. (2017). Recommender systems: An overview of different approaches to recommendations. *2017 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS)*, 1–4. https://doi.org/10.1109/ICIIECS.2017.8276172

Song, B., Gao, Y., & Li, X.-M. (2020). Research on Collaborative Filtering Recommendation Algorithm Based on Mahout and User Model. *Journal of Physics: Conference Series*, *1437*(1), 012095. https://doi.org/10.1088/1742-6596/1437/1/012095

Suharyadi, J., & Kusnadi, A. (2019). Design and Development of Job Recommendation System Based on Two Dominants on Psychotest Results Using KNN Algorithm. *International Journal of New Media Technology*, *5*(2), 116–120. https://doi.org/10.31937/ijnmt.v5i2.954

Thannimalai, V., & Zhang, L. (2021). A Content Based and Collaborative Filtering Recommender System. *2021 International Conference on Machine Learning and Cybernetics (ICMLC)*, 1–7. https://doi.org/10.1109/ICMLC54886.2021.9737238

Venkatesan, V. K., Ramakrishna, M. T., Batyuk, A., Barna, A., & Havrysh, B. (2023). High-Performance Artificial Intelligence Recommendation of Quality Research Papers Using Effective Collaborative Approach. *Systems*, *11*(2), 81.https://doi.org/10.3390/systems11020081

Wang, D., Liang, Y., Xu, D., Feng, X., & Guan, R. (2018). A content-based recommender system for computer science publications. *Knowledge-Based Systems*, *157*, 1–9. https://doi.org/10.1016/j.knosys.2018.05.001

Zhang, R., Liu, Q., Chun-Gui, Wei, J.-X., & Huiyi-Ma. (2014). Collaborative Filtering for Recommender Systems. *2014 Second International Conference on Advanced Cloud and Big Data*, 301–308. https://doi.org/10.1109/CBD.2014.47