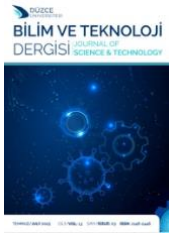


Düzce University Journal of Science & Technology

Research Article

Mitigating Popularity Bias in Fair and Explainable Recommender Systems Using SHAP

Tuğba TÜRKOĞLU KAYA [a,*]

[a] *Department of Computer Engineering, Faculty of Engineering, Ardahan University, Ardahan, TÜRKİYE*

** Corresponding author's e-mail address: tugbaturkoglu@ardahan.edu.tr*

DOI: 10.29130/dubited.1667105

ABSTRACT

Popularity bias is a prevalent issue in recommendation systems, where popular items dominate recommendation lists, leading to reduced diversity and fairness. Traditional methods evaluate popularity bias based on overall item frequency, disregarding individual user tendencies. This study introduces a novel post-processing ranking method called Dynamic User Tendency Re-ranking (DUTR) to mitigate popularity bias in multi-criteria recommendation systems by incorporating user-specific preferences. DUTR leverages SHAP (SHapley Additive exPlanations) analysis to determine the influence of different criteria on user decision-making. Unlike conventional methods, which classify item popularity based on general trends, DUTR dynamically assesses each user's priority preferences. It then classifies items as popular or less popular based on individual preference patterns. This approach ensures that recommendation lists align more closely with user-specific interests while maintaining a balance between popular and less popular items. To validate the effectiveness of DUTR, extensive experiments were conducted on the YM10 and YM20 datasets. The results show that DUTR significantly reduces popularity bias while improving diversity and fairness in recommendations. Moreover, the integration of SHAP values enhances the explainability of the recommendation process, providing users with personalized and transparent suggestions. In conclusion, comparative analysis with existing techniques demonstrates that DUTR outperforms traditional methods in balancing popularity and personalization.

Keywords: *Recommender systems, popularity bias, user tendency, SHAP, multi-criteria systems*

Adil ve Açıklanabilir Öneri Sistemlerinde SHAP ile Popülerlik Yanlılığını Giderme

ÖZET

Popülerlik yanlılığı, öneri sistemlerinde yaygın bir sorundur; popüler öğeler öneri listelerine hakim olur ve bu durum çeşitliliğin ve adaletin azalmasına neden olur. Geleneksel yöntemler popülerlik yanlılığını genel öğe sıklığına göre değerlendirirken, bireysel kullanıcı eğilimlerini göz ardı etmektedir. Bu çalışma, çok kriterli öneri sistemlerinde popülerlik yanlılığını azaltmak amacıyla, kullanıcıya özgü tercihler içeren yeni bir son işlem sıralama yöntemi olan Dinamik Kullanıcı Eğilimi Yeniden Sıralama (DUTR) yöntemini önermektedir. DUTR, kullanıcıların karar verme süreçlerinde farklı kriterlerin etkisini belirlemek için SHAP (SHapley Additive exPlanations) analizinden yararlanmaktadır. Geleneksel yöntemler öğelerin popülerliğini genel eğilimlere göre sınıflandırırken, DUTR her kullanıcının öncelikli tercihlerini dinamik olarak değerlendirmektedir. Daha sonra, bireysel tercih kalıplarına göre öğeleri popüler veya daha az popüler olarak sınıflandırmaktadır. Bu yaklaşım, öneri

Received: 27/03/2025, Revised: 01/05/2025, Accepted: 07/05/2025

listelerinin kullanıcıların özel ilgi alanlarıyla daha iyi örtüşmesini sağlarken, popüler ve daha az popüler öğeler arasında bir denge oluşturmayı hedeflemektedir. DUTR'nin etkinliğini doğrulamak için YM10 ve YM20 veri setleri üzerinde kapsamlı deneyler gerçekleştirilmiştir. Sonuçlar, DUTR'nin popülerlik yanlılığını önemli ölçüde azalttığını ve önerilerin çeşitliliğini ve adaletini artırdığını göstermektedir. Ayrıca, SHAP değerlerinin entegrasyonu, öneri sürecinin açıklanabilirliğini geliştirerek kullanıcılara kişiselleştirilmiş ve şeffaf öneriler sunmaktadır. Sonuç olarak, mevcut tekniklerle yapılan karşılaştırmalı analizler, DUTR'nin popülerlik ve kişiselleştirme arasında denge sağlamada geleneksel yöntemlerden daha başarılı olduğunu ortaya koymaktadır.
Anahtar Kelimeler: *Öneri sistemleri, popüler yanlılığı, kullanıcı eğilimi, SHAP, çok ölçütlü sistemler*

I. INTRODUCTION

In recent years, rapid advancements in science and technology have led to the emergence of large volumes of user-related data. Users' feedback on the products and services they experience forms a fundamental component of this massive data accumulation. However, when individuals seek to purchase a product or service online, this overwhelming amount of information can result in a loss of time during the decision-making process [1]. Recommender systems are effective mechanisms that mitigate the problem of information overload by analyzing heterogeneous data sources from different user groups, enabling users to spend less time selecting products and services [2]. Moreover, in these systems, which are primarily based on user satisfaction, having more detailed information about users allows for more effective customer analysis.

In particular, multi-criteria recommender systems can produce more accurate and personalized predictions by incorporating detailed information about user preferences. For instance, when deciding on hotel accommodations, users' expectations may vary significantly. While one user may prioritize the cleanliness of the hotel, another may focus on internet speed, and yet another may consider the hotel's location as the most important factor. To capture such diverse user preferences, it is essential to allow users to evaluate multiple attributes of the products or services they experience. Accordingly, the ratings provided for specific sub-criteria influence the overall score of the hotel and enable a more comprehensive assessment. As a result, evaluating a product or service through multiple criteria facilitates a more holistic understanding of customer preferences and contributes to increasing customer satisfaction. Therefore, creating recommendation lists that best meet users' expectations by leveraging the wide range of available data is considered a critical factor for the success of recommender systems.

Although recommender systems aim to enhance user satisfaction by providing effective recommendations, one of their major limitations is the problem of popularity bias. Popularity bias refers to the tendency of recommender systems to disproportionately suggest popular items while giving insufficient exposure to less popular ones. While the frequent recommendation of popular products may seem beneficial to users, it can also lead to certain drawbacks under specific conditions. This issue primarily arises from the unequal distribution of user preferences across items—where a small proportion of items receive a large number of interactions, while the majority of items remain underrepresented. Additionally, popularity bias may lead to the manipulation of the system through social bots and fake reviews aimed at increasing the visibility of specific products. Consequently, recent academic research has focused on examining the effects of popularity bias and developing methods to mitigate its negative impact beyond mere accuracy measures.

Moreover, recent studies on recommender systems have proposed various techniques to reduce popularity bias, which are generally classified based on their integration into the recommendation process as pre-processing, in-processing, and post-processing methods [3]. Among these, post-processing techniques are the most commonly used for addressing popularity bias. These techniques involve re-ranking the recommendation list produced by the algorithm or generating an alternative list.

However, a review of the existing literature reveals that while popularity bias has been extensively addressed in single-criteria systems, there is a significant gap in multi-criteria systems.

This study aims to tackle the problem of popularity bias in recommender systems by introducing a novel, user-centric perspective that diverges from traditional methods, which typically rely on global metrics such as the overall frequency of product selections. Unlike these conventional approaches that often assume uniform importance across all users and criteria, this study emphasizes the heterogeneity of user preferences by incorporating individual criterion-level priorities into the recommendation process. Specifically, the method utilizes SHAP (SHapley Additive exPlanations) analysis to compute the relative importance—or weights—of each criterion for each user based on their historical interactions and ratings [4]. This approach enables a fine-grained and interpretable understanding of how different criteria influence user decisions and allows for a more personalized treatment of popularity. By leveraging SHAP values, the system can identify which criteria (e.g., quality, price, usability, etc.) are most influential for a particular user and adjust the recommendation logic accordingly. Consequently, items that are globally popular but misaligned with a user's top-priority criteria may be deprioritized, while less popular items that closely match the user's preferences can be surfaced more prominently. This redefinition of popularity bias—grounded in individual-level explainability and multi-criteria modeling—supports a more balanced item exposure, enhances the perceived relevance of recommendations, and ultimately contributes to the fairness and transparency of the system. In this context, the study not only presents a methodological innovation but also expands the theoretical framework for understanding popularity bias in multi-criteria recommender systems, addressing existing limitations in the literature and proposing a more dynamic and user-aware solution.

The contributions of the study are listed below:

- *User-Centered Definition of Popularity:* Unlike conventional approaches based on the Pareto principle, this study personalizes and dynamically defines popularity by considering each user's individual criterion preferences.
- *SHAP-Based Criterion Analysis:* By quantifying the importance of criteria influencing users' decision-making processes through SHAP analysis, the study provides a more detailed and explainable approach to recommender systems.
- *Mitigating Popularity Bias:* The proposed method increases the likelihood of recommending less popular items by prioritizing products aligned with users' individual preferences, thus addressing a key limitation of conventional methods.
- *Personalized Recommendation Lists:* The study enhances user experience by generating recommendation lists tailored to each user's criterion priorities, ensuring a fairer recommendation process.
- *An Alternative Approach to the Literature:* This study offers an alternative solution to existing approaches addressing popularity bias, contributing to the dimensions of accuracy, fairness, and explainability in recommender systems.

This study consists of seven sections. While the literature review related to the popularity bias on recommender systems is included in Section 2, information about the methods and approach used throughout the study is given in Section 3. The proposed method is presented in Section 4 and Sections 5 and 6 describe the implementation and results of the proposed method and other approaches. In the last section, the general conclusions and suggestions from the study are presented.

II. RELATED WORK

The concept of popularity bias emphasizes the importance of considering less demanded products rather than focusing solely on popular services for users. In this context, Anderson [5] suggested that, due to the unlimited shelf space of digital platforms, the total sales of less demanded products could reach

levels that compete with popular products. Similarly, Brynjolfsson, Hu, and Smith [6] examined how long-tail products create economic value in the market and argued that digital environments facilitate the discovery of these products by offering more personalized recommendations tailored to user preferences. Celma and Cano [7] explored this issue in the field of music recommendation systems. Their research demonstrated that popular artists tend to dominate recommendation systems, which hinders the discovery of lesser-known artists.

Park and Tuzhilin [8] developed strategies to increase the recommendation of less popular items located in the long tail of recommendation systems. Their study highlighted the importance of reducing popularity bias to improve user access to more diverse and niche content and examined how long-tail items affect the overall performance of recommendation systems. Chen et al. [9] investigated the impact of missing data on recommendation systems in the context of user activity and item popularity. Another study by Kamishima et al. [10] proposed a neutrality-focused approach to mitigate popularity bias in recommendation systems, increasing the likelihood of recommending less popular items.

Abdollahpouri, Burke, and Mobasher [11] developed a learning-to-rank algorithm that aims to balance popular and less popular items. This approach has been shown to improve recommendation quality in terms of both accuracy and diversity. In another study, Abdollahpouri, Burke, and Mobasher [12] introduced a popularity-aware weighting method to increase the recommendation of long-tail items. Their subsequent research [13] focused on reducing popularity bias through personalized re-ranking of recommendation lists. This approach balances the recommendation list by accounting for each user's individual tolerance for popular and long-tail items.

In a later study, Abdollahpouri, Mansoury, Burke, and Mobasher [2] emphasized that popularity bias causes unfairness in recommendation systems and proposed various solutions to mitigate this bias. The study by Abdollahpouri et al. [14] evaluated popularity bias from a user-centered perspective and demonstrated that personalized approaches effectively reduce this bias.

Yalcin and Bilge [15] analyzed the unfair effects of popularity bias on users in recommendation systems. Their study identified the user groups most affected by popularity bias by considering different user characteristics and examined the effectiveness of strategies to mitigate this bias. In another study by the same researchers, they addressed popularity bias in recommendation systems through the "blockbuster" concept, analyzing the imbalances caused by the excessive recommendation of popular products and their effects on user experience [16]. In a separate study, they investigated how users' personality traits influence their susceptibility to popularity bias in recommendation systems [17]. In a study by Tacli et al. [18], two new methods—"Better-Than-Average" and "Positively-Rated"—were proposed to more accurately measure users' tendencies toward popular products. These methods aim to mitigate the effects of popularity bias by considering the products users favor.

All these studies aim to offer fairer, more diverse, and user-centered recommendations by developing various methods to mitigate popularity bias in recommendation systems. However, there is a noticeable gap in addressing this problem within multi-criteria recommendation systems. Therefore, this study aims to address this gap by proposing a novel approach.

III. PRELIMINARIES

This section presents the methods and approaches used in the study.

A. USER POPULARITY TENDENCY

This section provides information on the most commonly used methods in the literature for determining user popularity tendencies in post-processing techniques, as well as the methods employed in this study.

Ratio of Popular Items (RPI): This method is an approach designed to measure the density of popular items in users' profiles. Initially, all items are analyzed based on the number of ratings they receive and classified as either popular or non-popular. This classification is commonly based on the Pareto Principle (80/20 rule), which states that items receiving at least 20% of all ratings in the system are labeled as "popular" (head), while the remaining items are labeled as "non-popular" (tail) [14, 19]. After this classification, the proportion of popular items within the total items rated by a given user is calculated. This ratio indicates the user's inclination toward popular items.

Average Popularity of Rated Items (APRI): This method is used to measure users' tendencies toward popular items. The popularity of each item is determined by the ratio of the number of users who rated the item to the total number of users in the system. After calculating this popularity score, a user's popularity tendency is determined as the weighted average of the popularity scores of the items rated by that user [13].

Better Than Average (BTA): This method enhances the *RPI* and *APRI* methods by considering only the items that users like when calculating their popularity tendencies. Since rating an item does not necessarily mean that the user likes it, the *BTA* method establishes a threshold preference level for each user. This threshold is determined as the average of all ratings given by the user, and only items rated above this threshold are considered liked items [18]. In this context, the RPI_{BTA} method extends the classic *RPI* approach by calculating the proportion of popular items only among the items rated above the user's threshold preference level. Similarly, the $APRI_{BTA}$ method refines the *APRI* approach by computing the weighted average of the popularity scores only for the items liked by the user. Both methods focus on liked items rather than all rated items to model user preferences more precisely.

B. POPULARITY BIAS MITIGATION METHODS

This section considers four post-processing methods that stand out in the literature.

Popularity-Aware Item Weighting (Paiw): The Paiw method computes logarithmic inverse weights based on item popularity levels. In this process, less preferred (tail) items receive higher weights, while popular items are assigned lower weight values. These weights are then combined with recommendations to generate ranking scores. A control parameter (λ=0.5) is used to balance these two components (weights and recommendations) in the recommendation system [12].

Augmentation Approach (Aug): This method incorporates inverse weights of item popularity levels into the predicted scores to enhance recommendation accuracy. It aims to reduce popularity bias while improving recommendation precision [15].

Multiplicative Approach (Mul): This method penalizes popular items to encourage the recommendation of less popular items. Unlike the Aug method, it uses item weights as multiplicative factors when computing final ranking scores. The goal is to include more non-popular items in the recommendation lists [15].

Dynamic User-Oriented Re-ranking (DUoR): The DUoR method aims to balance popular and non-popular items when generating recommendation lists. At each step, it compares the current list with the user's popularity tendency and selects candidate items accordingly. If the popularity ratio is high, less popular items are preferred, and if it is low, popular items are chosen. This process continues until the recommendation list is complete. The DUoR method seeks to improve recommendation accuracy while mitigating popularity bias by better reflecting user preferences [18, 20].

IV. PROPOSED METHOD

A common issue in recommendation algorithms is popularity bias, which often leads to recommendation lists dominated by popular items. However, to enhance user experience, recommendations should not solely focus on popular items but instead provide a balanced presentation of both popular and non-popular items tailored to the user's profile. Recommendation systems that better align with users' interests and increase diversity can improve overall satisfaction. Accordingly, techniques aimed at reducing popularity bias should ensure that recommendation lists maintain this balance. This approach enables the development of fairer and more user-friendly recommendation systems, ultimately maximizing user satisfaction [18, 20].

Therefore, this study proposes a novel post-processing ranking technique called Dynamic User Tendency Re-ranking (DUTR), which aims to address the problem of popularity bias in multi-criteria systems by considering user tendencies. In this method, the priorities and weights of all users and items in the multi-criteria system were initially determined. For this determination, the SHAP (SHapley Additive exPlanations) analysis method was employed to measure how well the model captures user interests and to improve fairness. The next step involved determining users' popularity tendencies. At this stage, the popularity ratios for each user were calculated using the methods defined in Section III. In the subsequent step, the classification of items as head (popular) or tail (non-popular) was conducted based on each user's individual preferences at the item level. To illustrate this method with an example, consider user u_1, who prioritizes sub-criteria c_1, c_2, c_3, and c_4, with c_1 having a high priority and c_3 being of low priority for this user. Following this, the high and low priority criteria are determined for each item (see Table 1).

Table 1. *Sample item priorities.*

Items	High Priority	Low Priority
i_1	c_1	c_3
i_2	c_1	c_4
i_3	c_3	c_2
…	…	…

Unlike the existing method based on the Pareto principle in the literature, this stage focuses on user tendencies. Accordingly, in the step of classifying items as head or tail, items that exhibit similar tendencies to the user are considered. In the given example, since i_1 and i_2 show similar tendencies to the user, they are classified as head items, while i_3 is included in the tail items list. This process is repeated for each user, and the popularity status of items is examined separately for each individual. In the final step, which involves generating recommendations, the approach adopted by Gulsoy et al. [20] and Tacli et al. [18] is utilized to create a recommendation list for each user. In the approach determined by the researcher, it was aimed to provide a balance between the popularity rate in the recommendation list and the original user popularity. However, in this study, while providing this balance, the SHAP values of the popularity rate of the products added to the recommendation list were taken into account.

The algorithmic steps of the proposed approach:

The Proposed Approach: DUTR

Input: user_preference_data, candidate_products, user_popularity, N
Output: recommendation_list
1. recommendation_list = []
2. for i in range(N):
   2.1. Select the product with the highest estimated value based on the criteria the user shows high preference for → best_product
   2.2. Add best_product to recommendation_list.
   2.3. Calculate recommendation_popularity = popularity rate of recommendation_list.
   2.4. if recommendation_popularity> user_popularity then:
      2.4.1. Select the product with the highest estimated value based on the criteria the user shows low preference for → low_preference_product
      2.4.2. Add low_preference_product to recommendation_list.

2.5. else:
2.5.1. Select the product with the highest estimated value based on the criteria the user shows high preference for → high_preference_product
2.5.2. Add high_preference_product to recommendation_list.
3. Present the top-N recommendation_list to the user.

The main distinction of the proposed method from existing approaches in the literature is its consideration of both user and item tendencies. Consequently, each recommendation list presented to users follows a fairer approach that prioritizes user satisfaction.

V. EXPERIMENTS

In this section, firstly, the dataset, experimental methodology and performance evaluation criteria are briefly stated.

A. DATASET

In this study, the widely used Yahoo! Movies (YM) multi-criteria datasets are utilized in the field of recommendation systems (RS) [21, 22]. In the YM dataset, users evaluate movies based on four different criteria—acting, storyline, directing, and visuals—and provide an overall rating to express their general opinion of each film. Two subsets, YM10 and YM20, are used in this study. The YM10 dataset includes at least 10 ratings per user and movie, while the YM20 dataset includes at least 20 ratings. The original dataset contains ratings in the ranges of (13, 12, 11), (10, 9, 8), and (1), which have been converted to a (5), (4), and (1) scale, respectively, to fit a 1-to-5 rating system [23].

Table 2 summarizes the number of users and items (movies/hotels) in the YM10 and YM20 datasets. The YM20 subset consists of 429 users and 491 movies, while the YM10 subset includes 1,827 users and 1,471 movies.

Table 2. *YM dataset features.*

Dataset	#Users	#Items	#Ratings	Sparsity Ratio
YM20	429	491	8157	83,65%
YM10	1827	1471	34,846	97,69%

B. METHODOLOGY

In the experiments, the leave-one-out strategy was used to evaluate the performance of the recommendation system. Within this strategy, in each iteration, one user from the dataset was selected as the test user, while the remaining users formed the training set. For the test user, prediction values for all items were calculated using the Collaborative Filtering (CF) algorithm. In this calculation, the number of neighbors for the CF algorithm was set to n = 50, and predictions were generated based on the 50 most similar neighbors. Among the predicted items, the top M = 100 items with the highest predicted ratings were selected as the candidate item set (C) for the proposed DUTR method. Using the DUTR method, a top-10 recommendation list was generated for the test user. During this process, a popularity balancing mechanism was applied to dynamically balance popular and less popular items in the list. This methodology aims to enhance recommendation accuracy, while also reducing popularity bias and emphasizing the significance of user preferences.

C. PERFORMANCE MEASURE

This section presents the performance metrics used to evaluate the success of the proposed method and other related studies.

Precision: This metric measures the accuracy of the recommendation system and indicates how many of the recommended items are actually of interest to the user [24].

$$Precision = \frac{|True\ Positives(TP)|}{|All\ Recommendation\ (TP + FP)|} \quad (1)$$

where, TP (True Positives) represents correctly recommended items that genuinely interest the user, while FP (False Positives) refers to incorrectly recommended items that do not actually interest the user.

Recall: This metric measures the coverage rate of the recommendation system and indicates how many of the items that interest the user have been recommended [24].

$$Recall = \frac{|True\ Positives\ (TP)|}{|All\ Relational\ Items\ (TP + FN)|} \quad (2)$$

where, FN (False Negatives) represents items that were not recommended but are of interest to the user.
F1-Measure: It balances *Precision* and *Recall* [24].

$$F1 - Measure = 2x\frac{PrecisionxRecall}{Precision + Recall} \quad (3)$$

Novelty: Novelty refers to recommending items that the user is not familiar with but finds surprising [25].

$$Novelty = \frac{1}{|R|}\sum_{i \in R} -log_2(p(i)) \quad (4)$$

here, R represents the size of the recommendation list presented to the user, and p(i) denotes the probability that item i has been rated by any user.

Average Percentage of Long-Tail Items (APLT): This metric calculates the average percentage of long-tail items in the generated recommendation list [18, 20].

$$APLT = \frac{|i \in T \cap N|}{N} \quad (5)$$

here, T represents the set of long-tail items, while N refers to the recommendation list generated for the user.

Long-Tail Coverage (LTC): This metric measures how well the recommendation list covers long-tail items [18, 20].

$$LTC = \frac{|I_{N \cap T}|}{|T|} \quad (6)$$

Gini Index: The Gini index is a metric used in decision tree algorithms to measure the impurity or diversity of a dataset. It quantifies how mixed the classes are within a node: a Gini index of 0 indicates perfect purity (all elements belong to the same class), while higher values indicate greater class diversity. Essentially, it reflects the likelihood of incorrectly classifying a randomly chosen element if it were labeled according to the class distribution in the node. The lower the Gini index, the better the split for classification purposes.

$$Gini(D) = 1 - \sum (p_i)^2 \quad (7)$$

where, *D* is the dataset, p_i is the proportion of class *i* in dataset *D*.

VI. EXPERIMENTAL RESULTS

In this section, we present the experimental results obtained from the YM10 and YM20 datasets using four user popularity calculation methods (APRI, BTA_{APRI}, BTA_{RPI}, and RPI) and our proposed DUTR method with the other baseline popularity bias method (DuOR, Paiw, Mul, Aug). The performance of each method is evaluated across multiple metrics, including Precision, Recall, F1-Measure, Novelty, APLT, and LTC. We provide a detailed comparison to highlight the strengths and weaknesses of each approach and demonstrate the advantages of the DUTR method in balancing accuracy and novelty.

The results obtained for YM10 are shown in Figure1-4. Accordingly, the results shown in Figure 1, the APRI method demonstrates balanced performance across various metrics in identifying popular users. The precision values, particularly for the DUoR and DUTR categories, reach 0.0490 and 0.0411, respectively, indicating its effectiveness in accurately capturing relevant users. However, the recall values are lower compared to other methods, suggesting that APRI may overlook some potential popular users. The novelty and APLT values, except for the DUoR category, are relatively high, showing that the APRI method provides more diverse and unconventional recommendations

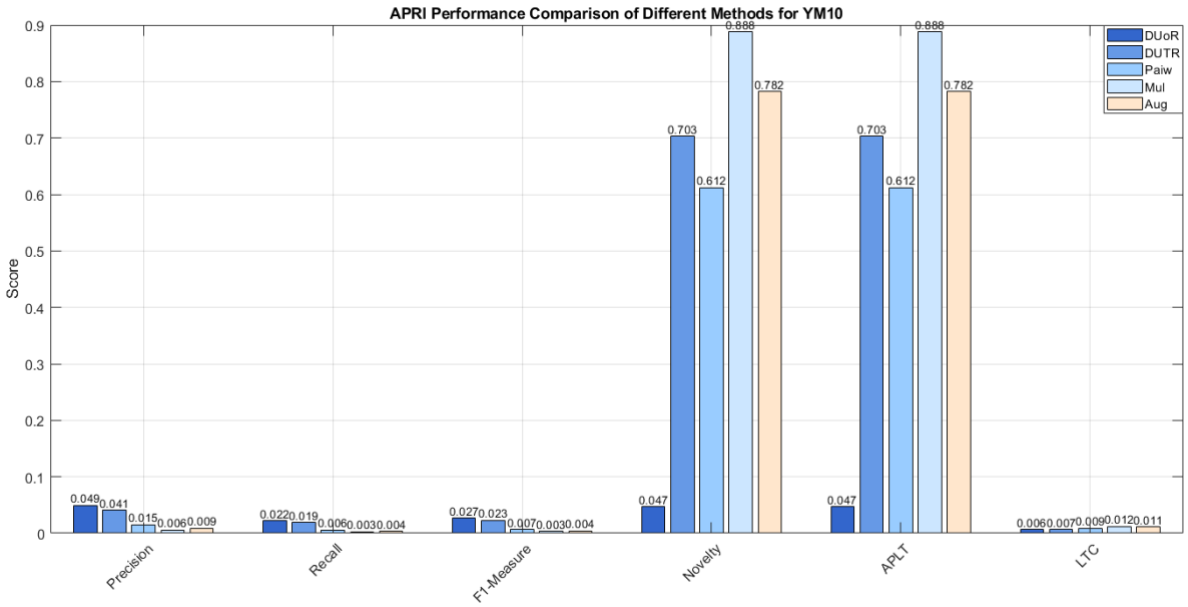


Figure 1. *APRI performance comparison of different methods for YM10*

The BTA_{APRI} method exhibits a performance similar to APRI while offering some improvements in novelty and APLT metrics (see Figure 2). The precision values remain almost unchanged; however, there is a slight decrease in recall values. A notable advantage of this method is its ability to deliver more diverse and innovative user recommendations, particularly in cases where fewer recommendations are made.

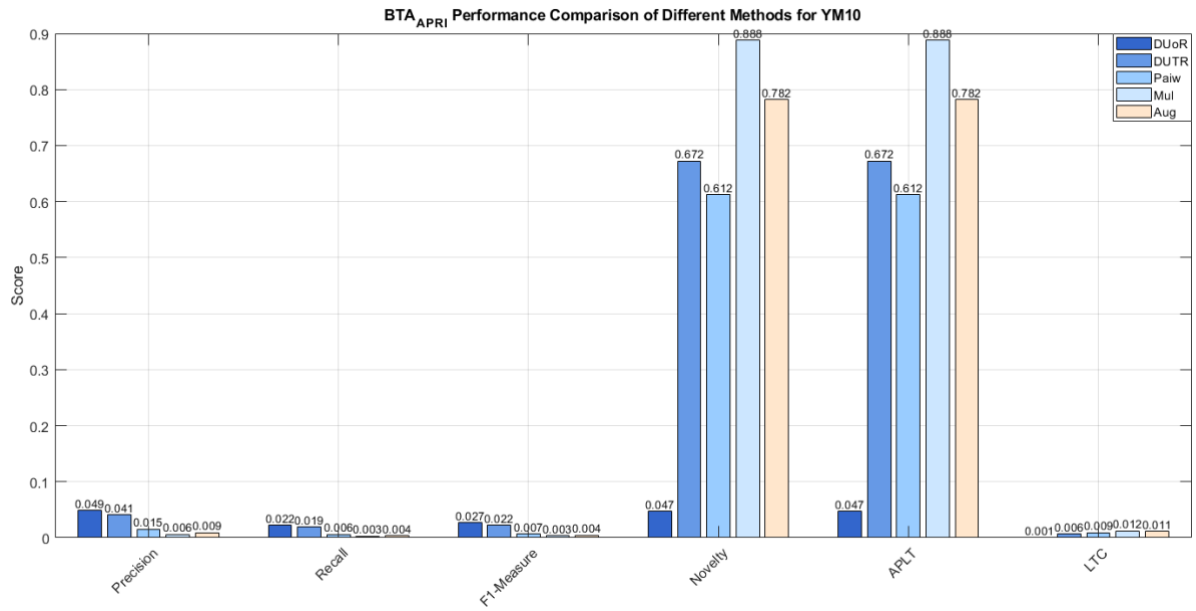


Figure 2. *BTA_{APRI} performance comparison of different methods for YM10*

According to Figure 3, the BTA_{RPI} method significantly outperforms the other methods in terms of novelty and APLT metrics. Notably, for the DUoR and DUTR categories, the novelty values reach 0.7975 and 0.7995, respectively. However, the precision and recall values are lower, indicating that while the BTA_{RPI} method provides more novel recommendations, it falls short in accuracy. This method is particularly suitable for identifying less frequently recommended users.

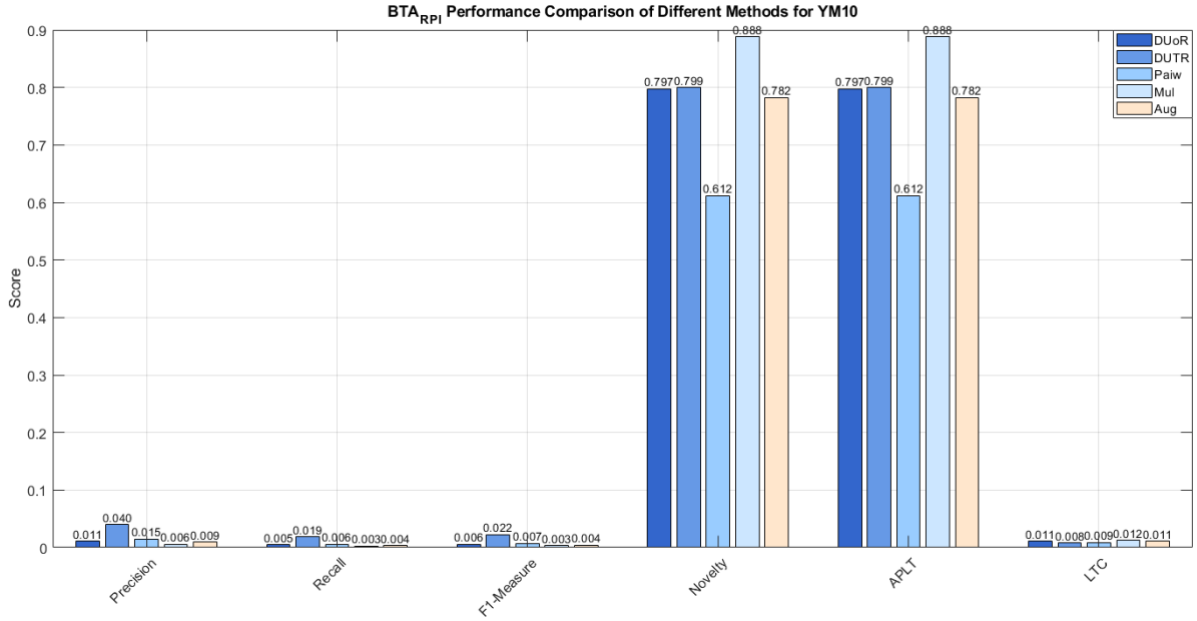


Figure 3. *BTA_{RPI} performance comparison of different methods for YM10*

The RPI method delivers precision and recall values comparable to APRI and BTA_{APRI}, but it underperforms in the novelty and APLT metrics across several categories. For instance, the novelty value for the DUoR category is as low as 0.1783, indicating that RPI tends to favor more common users. While this method ensures a broader inclusion of users in the recommendation list, it is limited in terms of diversity and innovation.

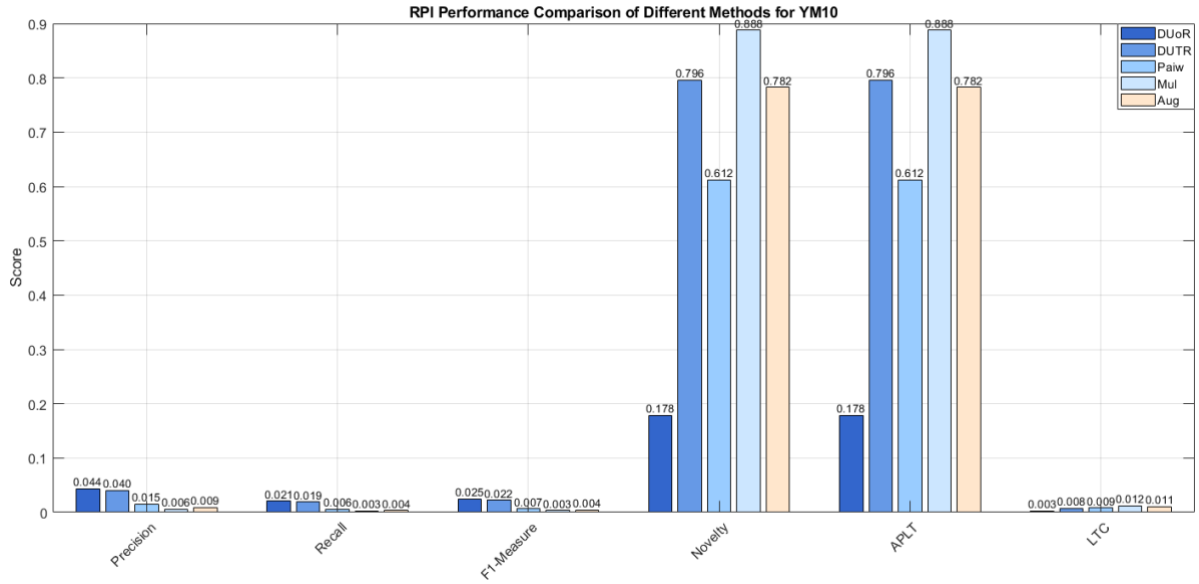


Figure 4. *RPI performance comparison of different methods for YM10*

Generally, in terms of precision, the APRI and BTA_{APRI} methods deliver the highest performance, but the DUTR method surpasses them by utilizing a user-tendency design that accurately reflects user preferences. Regarding recall, the DUTR method excels by capturing more prioritized users compared to the other methods, ensuring a more comprehensive recommendation list. While the BTA_{RPI} method offers the highest novelty, the DUTR method achieves a balanced approach by recommending both popular and less popular users, thereby providing a more diverse set of recommendations while maintaining relevance.

The APRI method's performance on the YM20 dataset reveals higher precision values compared to the YM10 dataset, particularly for the DUoR (0.1436) and DUTR (0.1186) categories (see Figure 5). This indicates an improved ability to accurately identify popular users. However, similar to the YM10 dataset, the recall values remain relatively low, with the highest being 0.0397 for the DUoR category. This suggests that while the method identifies relevant users, it may fail to capture the full spectrum of popular users. The novelty and APLT values display a wide range, with particularly high values for the Mul category (0.9014), reflecting the method's capacity to provide diverse and less conventional recommendations in certain cases. Additionally, the LTC values increase across categories, particularly for Mul (0.0539), indicating greater temporal coverage in recommendations. Overall, the APRI method performs better in precision on the YM20 dataset but continues to face challenges in achieving high recall across all categories.

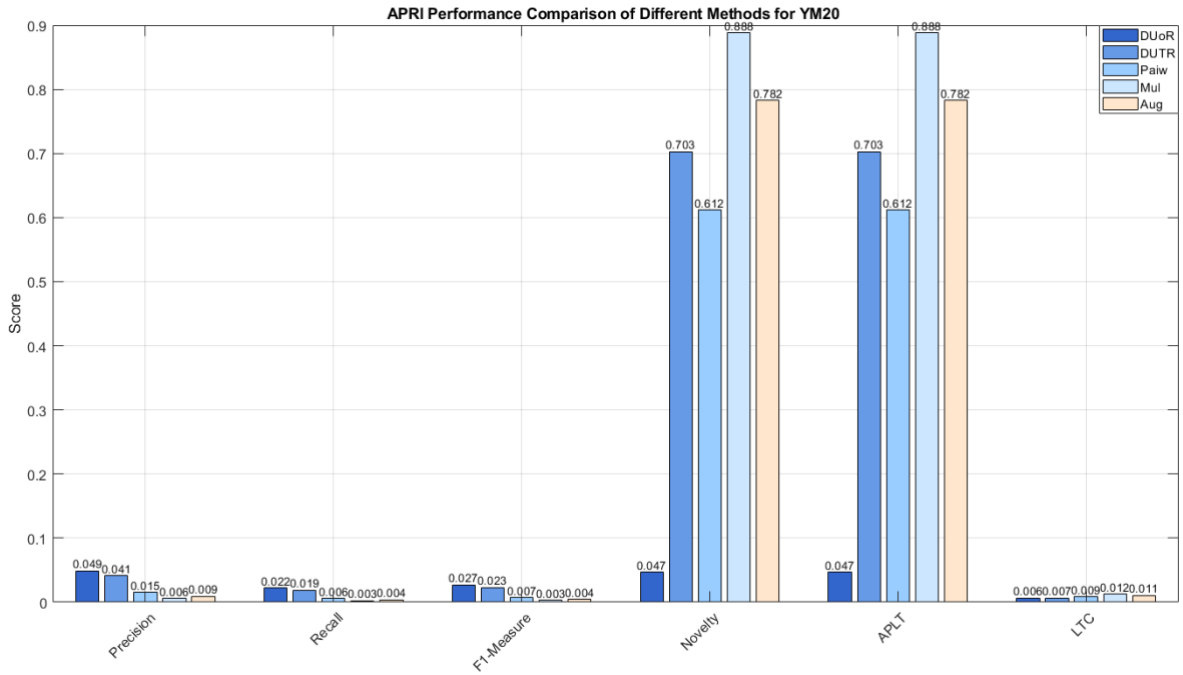


Figure 5. *APRI performance comparison of different methods for YM20*

The BTA_{APRI} method on the YM20 dataset demonstrates a performance pattern similar to the APRI method but with slight variations (see Figure 6). Precision values for the DUoR and DUTR categories (0.1444 and 0.1179) are comparable to APRI, indicating that the method maintains a consistent level of accuracy in identifying popular users. However, there is a marginal decrease in recall across all categories, with the highest value being 0.0390 for DUoR. This suggests that the method is slightly less effective in capturing the full range of popular users. The novelty and APLT values for the Mul category remain high at 0.9010, showcasing the method's ability to recommend diverse users. Meanwhile, the LTC values are slightly lower than those of the APRI method, suggesting a minor reduction in the temporal diversity of recommendations. Overall, BTA_{APRI} maintains a similar performance level to APRI, with slight trade-offs between precision, recall, and novelty.

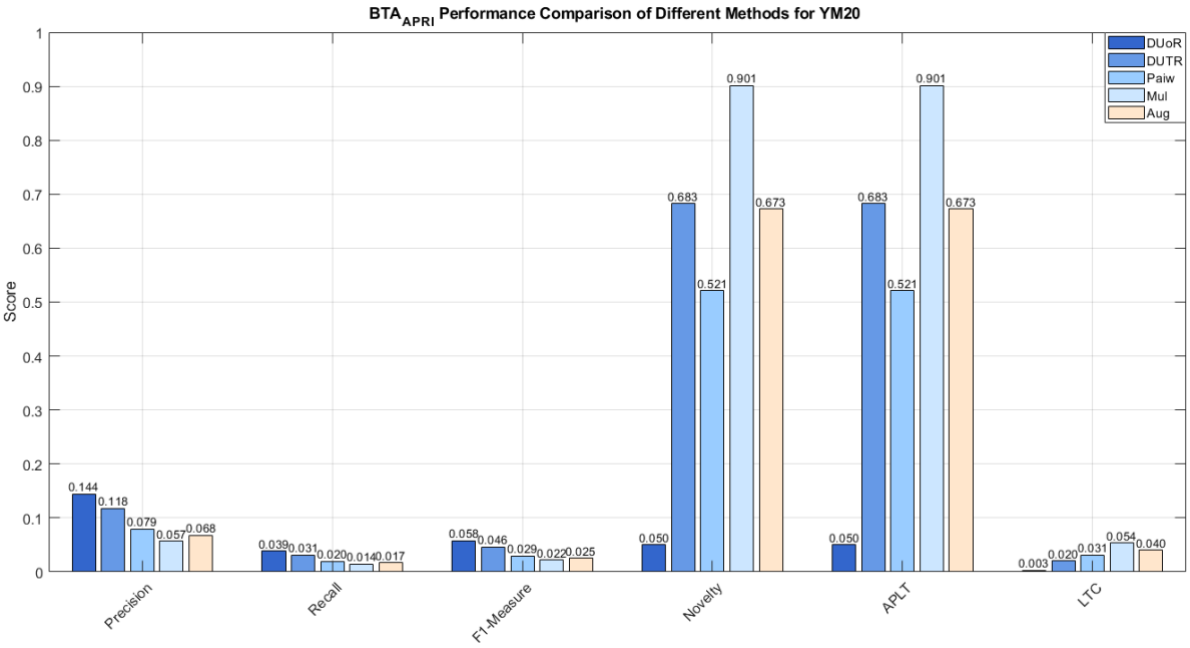


Figure 6. *BTA_{APRI} performance comparison of different methods for YM20*

According to Figure 7, the BTA_{RPI} method exhibits a distinct performance profile on the YM20 dataset, particularly excelling in the novelty and APLT metrics. The novelty values for the DUoR (0.8270) and DUTR (0.8455) categories significantly surpass those of other methods, indicating a strong ability to provide diverse and unique recommendations. However, this increase in novelty comes at the expense of precision and recall, which are generally lower across all categories. For example, the precision for the DUoR category is 0.0760, while the recall value is 0.0202. This suggests that while the BTA_{RPI} method is highly effective in identifying novel users, it may fail to capture a comprehensive set of relevant users. The LTC values vary, with the DUoR category showing the highest value (0.0495), reflecting a broader temporal scope. Overall, the BTA_{RPI} method is most suitable when the goal is to prioritize diverse and less conventional user recommendations, although it may sacrifice accuracy in doing so.

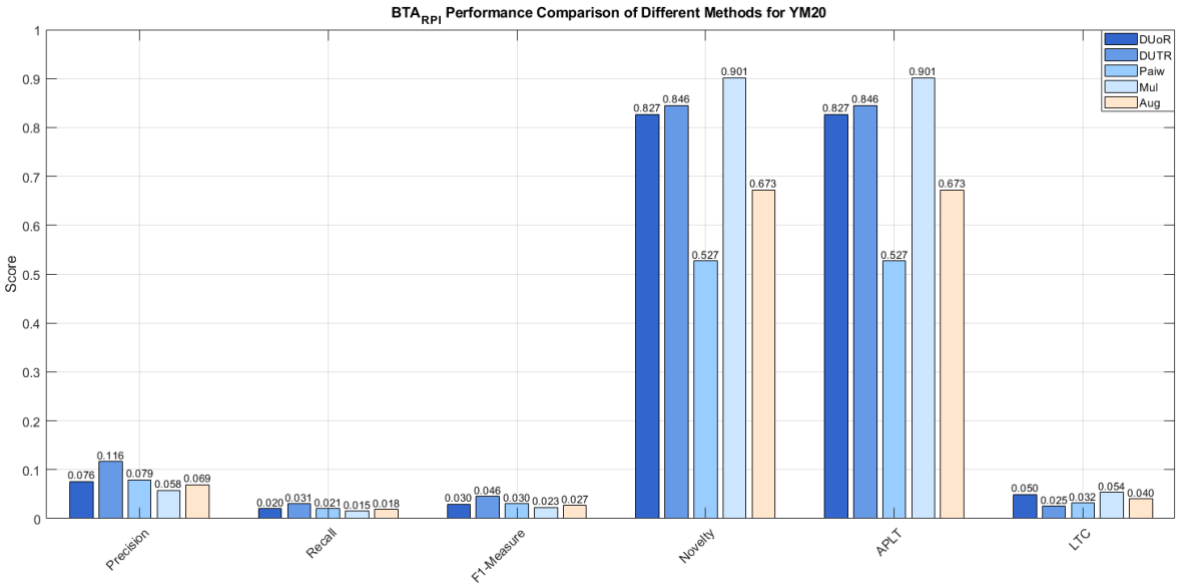


Figure 7. *BTA_{RPI} performance comparison of different methods for YM20*

The RPI method shows varying performance across different evaluation metrics on the YM20 dataset. In terms of precision, the RPI method provides high values for the DUoR scenario (0.1273), which is only slightly lower than the APRI and BTA_{APRI} methods. However, it shows comparable precision to the DUTR method (0.1163) while outperforming the other scenarios. Recall results indicate that RPI underperforms compared to APRI and BTA_{APRI}, especially in the DUoR scenario (0.0347), which is lower than the best-performing DUTR method. F1-Measure values follow a similar trend, with the RPI method reaching a maximum of 0.0515 in the DUoR scenario.

The novelty metric reveals significant variation: while the RPI method exhibits moderate novelty for the DUoR scenario (0.2068), it achieves one of the highest novelty values (0.8455) for the DUTR scenario. This indicates that the RPI method tends to recommend less conventional users under specific scenarios. For APLT, the results align closely with the novelty metric, suggesting a similar pattern. LTC values are slightly higher in the DUoR scenario (0.0124) but remain comparable across other scenarios, indicating a balanced long-tail coverage.

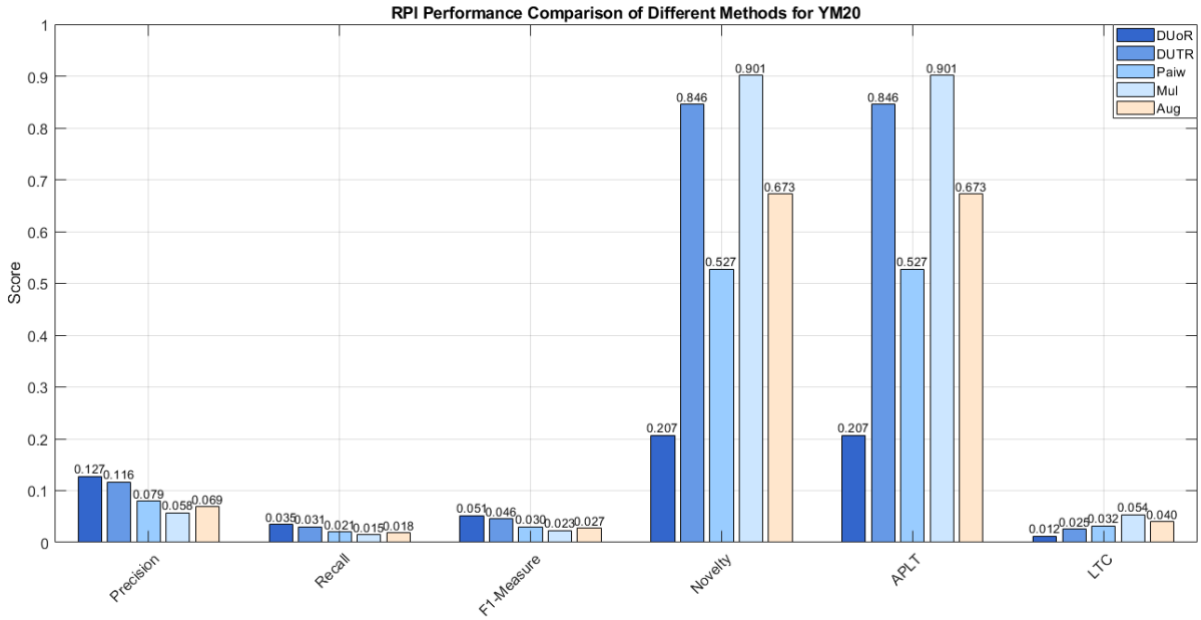


Figure 8. *RPI performance comparison of different methods for YM20*

In the comparative analysis for YM20, the APRI and BTA_{APRI} methods perform consistently well in precision, especially in the DUoR scenario, while the DUTR method achieves competitive performance across all scenarios and shows strength in recall. The BTA_{RPI} method stands out for providing the highest novelty (0.8270 in the DUoR scenario), while the RPI method offers a trade-off between precision and novelty. The DUTR method maintains a balance across all metrics, providing a comprehensive solution that captures both popular and less popular users effectively.

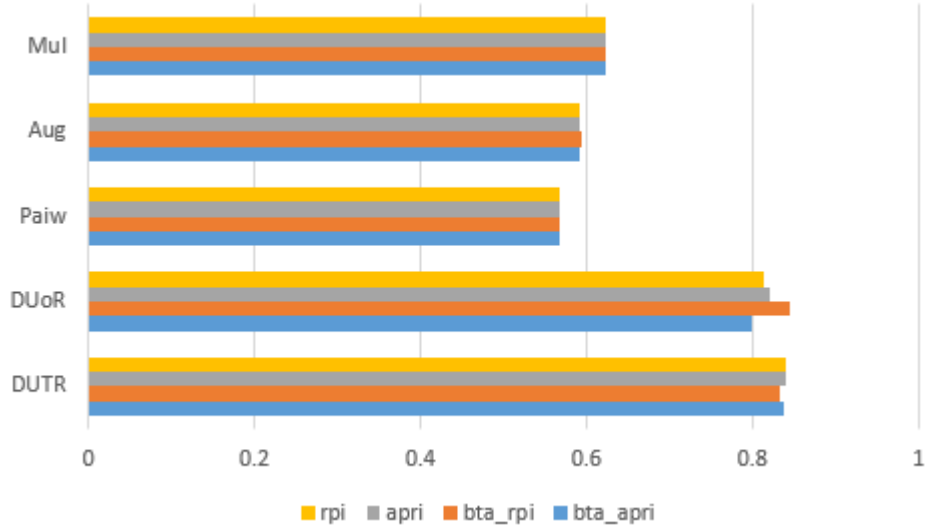
When comparing the YM10 and YM20 datasets, similar trends emerge across all methods. The DUTR method consistently performs well across both datasets, especially in recall and F1-Measure, due to its user-tendency design. The BTA_{RPI} method remains the best in terms of novelty across both datasets, while APRI and BTA_{APRI} methods maintain superior precision.

The RPI method exhibits a balanced performance, particularly excelling in novelty in the DUTR scenario but trailing behind in recall. Overall, the DUTR method's ability to capture both popular and less popular users provides a holistic advantage, while other methods show strength in specific evaluation metrics. The comparative analysis underscores DUTR's versatility and its capacity to address the limitations of traditional methods by achieving a balance between accuracy and novelty.

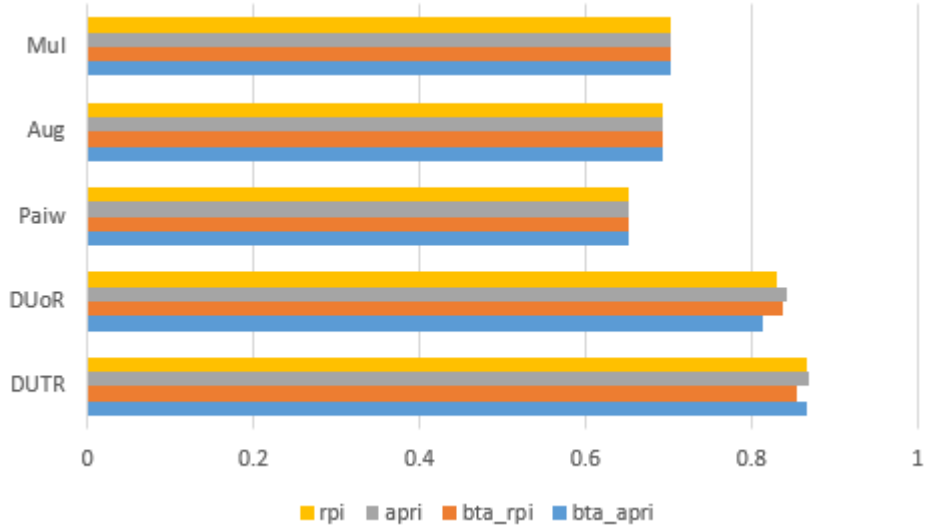
The Figure 9 presents a comparison of Gini index values obtained for five different methods (Mul, Aug, Paiw, DUoR, and DUTR) across four different strategies (RPI, APRI, BTA_{APRI}, BTA_{RPI}). A higher Gini index indicates greater heterogeneity among the classes, implying less pure data partitions from a classification perspective. The Mul, Aug, and Paiw methods yielded similar Gini values across all strategies, with values consistently around 0.65. This suggests that these methods exhibit comparable levels of class purity in their partitions. In contrast, the DUoR method, particularly when combined with the rpi strategy, produced a notably high Gini index (~0.85), indicating that this configuration leads to more imbalanced class distributions and less pure nodes. The DUTR method, on the other hand, stands out with generally lower Gini index values, especially under the bta_apri strategy where the Gini value falls below 0.75. This implies that DUTR is more effective in generating homogeneous class structures. These findings suggest that, compared to other methods, DUTR has a superior capability to produce more discriminative and purer splits within datasets.
In the same experimental study conducted on the other dataset, YM10, the Gini index values of five different methods—DUTR, DUoR, Paiw, Aug, and Mul—are compared across four recommendation strategies: BTA_{APRI}, BTA_{RPI}, APRI, and RPI. The results indicate that the DUTR method consistently yields the highest Gini index values across all strategies. Notably, it reaches a peak value of 0.8685

under the APRI strategy, suggesting that DUTR prioritizes the dominant criteria of users, resulting in a more concentrated recommendation list and, consequently, a less equitable distribution of recommendations. In contrast, the DUoR method exhibits relatively lower Gini values compared to DUTR, implying a more balanced distribution of recommendations. However, DUoR does not completely eliminate user-centric concentration and instead maintains a moderate trade-off between personalization and fairness.

The Paiw method achieves the lowest Gini index values, ranging narrowly between 0.6527 and 0.6528 across all four strategies. This outcome demonstrates Paiw's superior ability to ensure a more equitable recommendation distribution and preserve diversity within the recommendation lists. Similarly, the Aug and Mul methods produce comparable Gini values, offering a moderate level of fairness; Aug reports a value of 0.6929, while Mul follows closely with 0.7038, indicating a mid-level concentration in their recommendation outputs. Overall, while DUTR stands out in terms of personalization, Paiw emphasizes fairness by ensuring a more uniform distribution of recommendations. DUoR, Aug, and Mul strike varying degrees of balance between these two competing objectives, highlighting the importance of aligning method selection with the specific goals of the recommendation system in question.



(a)

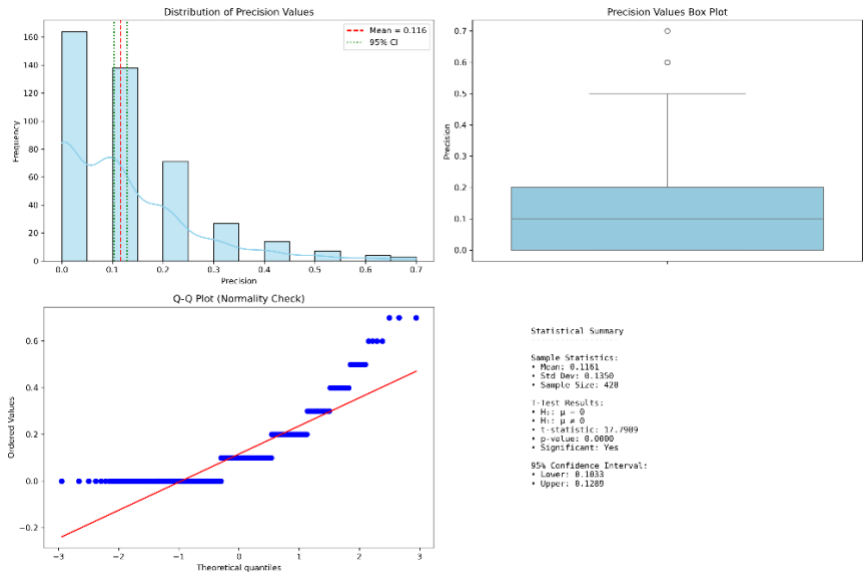


(b)

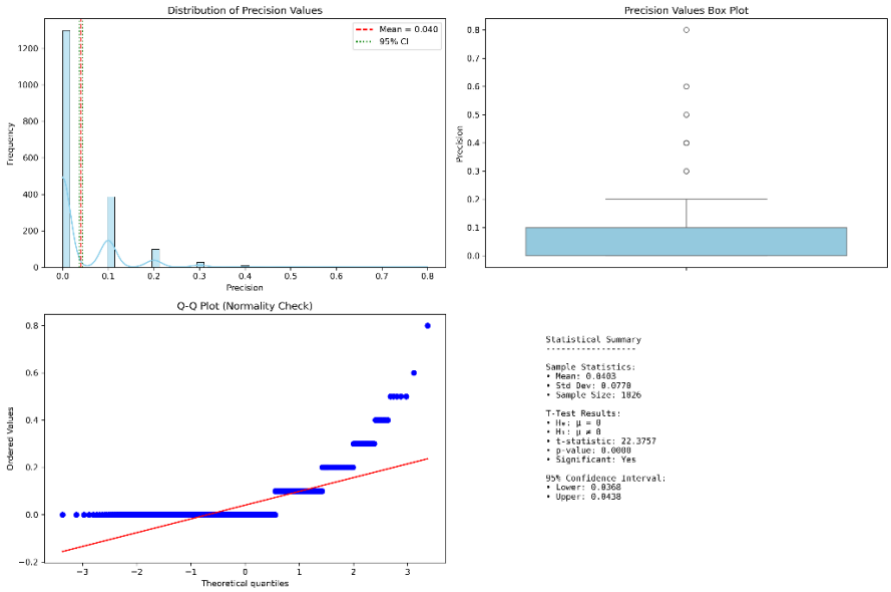
Figure 9. *Gini index value for YM20 (a) and YM10 (b)*

In the final stage, the statistical analysis results for the proposed method indicate that the precision values associated with the DUTR method demonstrate statistically significant performance for YM20 and YM10 datasets. The t-test conducted on a total of 428 samples yielded a mean precision of 0.1161 with a standard deviation of 0.1350 (see Figure 10(a)). The obtained t-statistic was 17.7989, with a corresponding p-value of 0.0000. Since the p-value is well below the significance level of $\alpha = 0.05$, the null hypothesis (H_0: $\mu = 0$) is rejected, confirming that the mean precision is significantly greater than zero. Moreover, the 95% confidence interval [0.1033, 0.1289] indicates that the true population mean lies entirely above zero, further supporting the conclusion that the DUTR method provides significantly better precision compared to random guessing. These findings statistically validate that the DUTR method achieves a precision level that is meaningfully different from zero, demonstrating statistically significant performance. However, despite this significance, the relatively low average precision suggests that the method's absolute performance remains limited. Therefore, it can be concluded that while the DUTR method yields statistically significant results, its overall precision performance is modest.

The statistical analysis presented in the Figure 10(b) provides a comprehensive evaluation of the precision performance of the DUTR method on the YM10 dataset. The histogram reveals that the majority of precision values are concentrated below 0.1, with a reported mean precision of approximately 0.040, indicating that the method consistently produces low yet frequent precision scores. The boxplot further illustrates a notable presence of outliers, suggesting that while the method often yields modest precision, it occasionally achieves substantially higher values in certain cases. The Q-Q plot (Quantile-Quantile plot) demonstrates a clear deviation from the normal distribution, particularly at the distribution tails, indicating non-normality and a positively skewed distribution. This deviation suggests that parametric assumptions may not be entirely appropriate for inferential analysis. Additionally, the 95% confidence interval for the mean precision is reported as [0.0383, 0.0416], reinforcing the reliability and robustness of the method's performance. In summary, these findings indicate that the DUTR method demonstrates a statistically significant and consistent precision performance on the YM10 dataset, despite the presence of outliers and non-normality in the data distribution.



(a)



(b)

Figure 10. *Statistical analysis of DUTR for YM20 (a) and YM10 (b)*

VII. CONCLUSION AND FUTURE WORK

In this study, we analyzed and compared the performance of four user popularity calculation methods (APRI, BTA_{APRI}, BTA_{RPI}, and RPI) along with our proposed DUTR method, using the YM10 and YM20 datasets. Additionally, we incorporated four baseline popularity bias methods (DUoR, Paiw, Mul, Aug) to provide a comprehensive evaluation. Our experimental results demonstrate that the DUTR method effectively balances accuracy and novelty across both datasets.

In terms of Precision, DUTR consistently performs competitively with the APRI and BTA_{APRI} methods across both datasets, particularly in the DUTR position, showing enhanced user prioritization capabilities. For Recall, DUTR achieves superior results compared to other methods, especially in capturing relevant but less popular users. The F1-Measure results indicate that DUTR provides a balanced performance by optimizing the trade-off between precision and recall. While the Novelty metric is highest for BTA_{RPI}, the DUTR method maintains a favorable balance by recommending both popular and less popular users. Moreover, DUTR demonstrates stable performance across APLT and LTC, ensuring diverse and meaningful recommendations without compromising on novelty.

A comparative analysis between the YM10 and YM20 datasets reveals that DUTR consistently outperforms other methods in recall and offers a competitive balance in precision and novelty. This suggests that DUTR is particularly effective in scenarios where identifying underrepresented users is crucial. Additionally, the DUTR method is robust across different datasets, demonstrating its adaptability and scalability.

Overall, the DUTR method's capacity to balance user accuracy and novelty while addressing popularity bias provides a significant improvement over existing techniques. By focusing on user-specific preferences and mitigating the dominance of popular users, DUTR enhances the fairness and diversity of recommendation systems. Briefly, the newly proposed DUTR method enhances recommendation accuracy by precisely capturing users' individual criteria priorities. Unlike other methods, it balances both user- and item-based priorities, providing more personalized and meaningful recommendations. The key advantages of the DUTR method include:

- *Higher Precision and Recall:* By accurately identifying users' priorities, DUTR produces both more accurate and comprehensive recommendations.

- *Balanced Novelty and Diversity:* It covers not only popular items but also less-known items that may interest users.

- *User-Tendency Approach:* Leveraging SHAP analysis to understand user priorities increases user satisfaction by offering more personalized recommendations.

Although the DUTR method demonstrates promising results, several avenues for future research remain open. One potential direction is to extend the method's applicability to other large-scale, real-world datasets to further validate its robustness and generalizability. Additionally, exploring the integration of explainability techniques with the DUTR framework could enhance user trust and transparency by providing interpretable recommendation rationales.

Another important direction involves dynamic user modeling, where user preferences evolve over time. Incorporating temporal information could improve the model's ability to capture changes in user behavior and adapt recommendations accordingly. Furthermore, applying DUTR to multi-objective optimization problems, where multiple conflicting criteria must be balanced, represents an intriguing area for future exploration.

Finally, extending the DUTR framework to address other forms of bias, such as long-tail item bias or demographic bias, could further enhance its fairness and applicability across diverse recommendation environments. Collaborative approaches that combine DUTR with other cutting-edge techniques, such as reinforcement learning or graph-based models, also offer promising possibilities for advancing the field of fair and explainable recommendation systems. Also, while SHAP (SHapley Additive exPlanations) values provide a snapshot of feature importance at a given point in time, user preferences in real-world scenarios are rarely static. In this study, we address this temporal aspect by allowing SHAP values to be recalculated periodically based on updated user-item interactions. As users continue to interact with the system and rate items, their underlying preferences may shift—placing more emphasis on certain criteria (e.g., storytelling over visuals) than before. These preference drifts are captured by retraining the model at regular intervals and re-computing SHAP values accordingly. This dynamic recalibration ensures that the system remains sensitive to evolving user priorities, thereby preserving the accuracy, fairness, and explainability of recommendations over time.

In future extensions, modeling these changes explicitly using time-aware or session-based SHAP analysis (e.g., through window-based evaluation or recurrent user profiles) could further enhance the system’s responsiveness to temporal preference evolution.

Article Information

Acknowledgments: Not applicable.

Author’s Contributions: The author carried out the conceptualization, methodology, analysis, writing—original draft, and writing—review and editing of the manuscript. The author has read and approved the final version of the manuscript.

Artificial Intelligence Statement: During the preparation of this work the author used Gemini and ChatGPT in order to refine written text for grammar checking, readability, and fluency. After using this

tool, the author reviewed and edited the content as needed and takes full responsibility for the content of the publication.

Conflict of Interest Disclosure: The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Plagiarism Statement: This article was checked for plagiarism using Turnitin, and the similarity index is within the acceptable range.

V. REFERENCES

[1] F. O. Isinkaye, Y. Folajimi and B. A. Ojokoh, "Recommendation systems: Principles, methods and evaluation," *Egyptian Informatics Journal*, vol. 16, pp. 261–273, 2015.

[2] H. Abdollahpouri, M. Mansoury, R. Burke and B. Mobasher, "The unfairness of popularity bias in recommendation," *RMSE Workshop 13th ACM Conferenceon Recommender Systems (RecSys)*, Copenhagen, Denmark, 2019.

[3] L. Boratto, G. Fenu and M. Marras, "Connecting user and item perspectives in popularity debiasing for collaborative recommendation," *Information Processing and Management*, vol. 58, 2021, Art. no. 102387.

[4] I. Covert and S.-I. Lee, "Improving KernelSHAP: Practical shapley value estimation using linear regression," in *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, San Diego, California, USA, 2021, pp. 3457–3465.

[5] C. Anderson, *The Long Tail: Why the Future of Business is Selling More For Less*. New York, USA: Hyperion, 2006.

[6] E. Brynjolfsson, Y. J. Hu and M. D. Smith, "From niches to riches: Anatomy of the long tail," *Sloan Management Review*, vol. 47, no. 4, pp. 67–71, 2006.

[7] Ò. Celma and P. Cano, "From hits to niches?: Or how popular artists can bias music recommendation and discovery," in *NETFLIX '08: Proceedings of the 2nd KDD Workshop on Large-Scale Recommender Systems and the Netflix Prize Competition*, Las Vegas, Nevada, USA, 2008, pp. 1-8.

[8] Y. J. Park and A. Tuzhilin, "The long tail of recommender systems and how to leverage it," in *RecSys'08: Proceedings of the 2008 ACM Conference on Recommender Systems*, Lausanne, Switzerland, 2008, pp. 11-18.

[9] C. Chen, M. Zhang, Y. Liu and S. Ma, "Missing data modeling with user activity and item popularity in recommendation," in *Information Retrieval Technology: 14th Asia Information Retrieval Societies Conference, AIRS 2018*, Y.H. Tseng et al., Eds., Taipei, Taiwan, 2018, pp. 113-125.

[10] T. Kamishima, S. Akaho, H. Asoh and J. Sakuma, "Correcting popularity bias by enhancing recommendation neutrality," *Proceedings of the 8th ACM Conference on Recommender Systems (RecSys 2014)*, Foster City, Silicon Valley, USA, 2014.

[11] H. Abdollahpouri, R. Burke and B. Mobasher, "Controlling popularity bias in learning-to-rank recommendation," in *RecSys 2017 - Proceedings of the 11th ACM Conference on Recommender Systems*, Como, Italy, 2017, pp. 42–46, 2017.

[12] H. Abdollahpouri, R. Burke and B. Mobasher, "Popularity-aware ıtem weighting for long-tail recommendation," *arXiv preprint arXiv:1802.05382*, 2018.

[13] H. Abdollahpouri, R. Burke and B. Mobasher, "Managing popularity bias in recommender systems with personalized re-ranking," in *Proceedings of the 32nd International Florida Artificial Intelligence Research Society Conference, FLAIRS 2019*, Florida, USA, 2019.

[14] H. Abdollahpouri, M. Mansoury, R. Burke, B. Mobasher and E. Malthouse, "User-centered evaluation of popularity bias in recommender systems," in *UMAP'21: Proceedings of the 29th ACM Conference on User Modeling, Adaptation and Personalization*, Utrecht, Netherlands, 2021, pp. 119-129.

[15] E. Yalcin and A. Bilge, "Evaluating unfairness of popularity bias in recommender systems: A comprehensive user-centric analysis," *Information Processing & Management*, vol. 59, no. 6, 2022, Art. no. 103100.

[16] E. Yalcin and A. Bilge, "Blockbuster: A new perspective on popularity-bias in recommender systems," in *2021 6th International Conference on Computer Science and Engineering (UBMK)*, Ankara, Türkiye, 2021, pp. 107–112.

[17] E. Yalcin and A. Bilge, "Popularity bias in personality perspective: An analysis of how personality traits expose individuals to the unfair recommendation," *Concurrency and Computation: Practice and Experience*, vol. 35, no. 9, 2023, Art. no. e7647.

[18] Y. Tacli, E. Yalcin and A. Bilge, "Novel approaches to measuring the popularity inclination of users for the popularity bias problem," in *2022 International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT)*, Ankara, Türkiye, 2022, pp. 555–560.

[19] R. Sanders, "The Pareto principle: its use and abuse," *Journal of Services Marketing*, vol. 1, no. 2, pp. 37–40, 1987.

[20] M. Gulsoy, E. Yalcin, Y. Tacli and A. Bilge, "DUoR: dynamic user-oriented re-ranking calibration strategy for popularity bias treatment of recommendation algorithms," *International Journal of Human-Computer Studies*, vol. 203, 2025, Art. no. 103578.

[21] K. Lakiotaki, N. F. Matsatsinis and A. Tsoukias, "Multicriteria user modeling in recommender systems," *IEEE Intelligent Systems*, vol. 26, pp. 64–76, 2011.

[22] H. Wang, Y. Lu and C. Zhai, "Latent aspect rating analysis without aspect keyword supervision," in *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, San Diego, California, USA, 2011, pp. 618–626.

[23] A. M. Turk and A. Bilge, "Robustness analysis of multi-criteria collaborative filtering algorithms against shilling attacks," *Expert Systems with Applications*, vol. 115, pp. 386–402, 2019.

[24] J. L. Herlocker, J. A. Konstan, A. Borchers and J. Riedl, "An algorithmic framework for performing collaborative filtering," in *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 1999*, Bberkeley, CA, USA, 1999, pp. 230-237.

[25] L. Zhang, "The definition of novelty in recommendation system," *Journal of Engineering Science & Technology Review*, vol. 6, pp. 141–145, 2013.