VOL. 12. NO. 1 AUGUST 2026
P-ISSN: 2685-8223 | E-ISSN: 2527-4864
DOI: 10.33480/jitk.v12i1.8240

JITK (JURNAL ILMU PENGETAHUAN DAN TEKNOLOGI KOMPUTER)

IMPROVING LONG-TAIL RECOMMENDATION VIA ROULETTE WHEEL SELECTION USING HYBRID RATING AND TEMPORAL DECAY WEIGHTING

Andy Maulana Yusuf [1*]; Miftah Farid Adiwisastra[2]; Faizal Riza[3]; Musrinah[4]

School of Computing[1]
Telkom University, Bandung, West Java, Indonesia[1]
https:// telkomuniversity.ac.id [1]
andymaulanayusuf@telkomuniversity.ac.id*

Department of Information Systems[2]
Universitas Bina Sarana Informatika, Tasikmalaya, West Java, Indonesia[2]
https://bsi.ac.id/ [2]
miftah.mow@bsi.ac.id

Department of Informatics Engineering[3]
Institut Teknologi Budi Utomo, Jakarta, Indonesia[3]
https://www.itbu.ac.id/ [3]
faizalriza@itbu.ac.id

Energy Conversion Engineering[4]
Politeknik Bandung, Bandung, West Java, Indonesia[4]
https://www.polban.ac.id/ [4]
musrinah@polban.ac.id

Abstract— *The phenomenon of information overload on digital platforms frequently leads to a skewed long-tail distribution, where user interactions concentrate on popular items while the majority remains undiscovered. This popularity bias is particularly acute in the book domain, where metaphorical titles defy traditional lexical matching, causing unique content to remain hidden in the long-tail area. While semantic models improve relevance, a significant research gap persists in mitigating the trade-off between accuracy and catalog exposure. This study proposes a novel stochastic framework by integrating Roulette Wheel Selection with hybrid rating and temporal decay weighting to enhance long-tail discoverability. To evaluate its generalizability, this mechanism was tested on the Goodreads Poetry dataset across four distinct content-based embedding models: SBERT-Tuned, SBERT, FastText, and Word2Vec. We conducted a rigorous comparative analysis between the proposed Stochastic (S) approach and Deterministic (D) Top-N selection. Experimental results demonstrate that the stochastic mechanism consistently improves catalog diversity across all baselines. SBERT-Tuned emerged as the most promising model, achieving a 461.71% increase in Catalog Coverage with a marginal 2.15% Recall reduction. Other models also exhibited significant gains, where the integration of Roulette Wheel Selection boosted coverage by 381.72% for FastText and 225.15% for Word2Vec, highlighting the framework's robustness. Furthermore, a consistent decrease in the Gini-index and an improvement in Novelty scores (ranging from +16% to +29%) confirm a more equitable distribution of item visibility. This research contributes a methodological advancement by proving that stochastic selection, guided by time-aware weighting, can effectively dismantle popularity bias without sacrificing relevance.*

Keywords: *Hybrid Rating-Temporal Decay Weighting, Long-tail Recommendation, Popularity Bias, Sentence-BERT, Stochastic Roulette Wheel.*

JITK (JURNAL ILMU PENGETAHUAN DAN TEKNOLOGI KOMPUTER)
VOL. 12. NO. 1 AUGUST 2026
P-ISSN: 2685-8223 | E-ISSN: 2527-4864
DOI: 10.33480/jitk.v12i1.8240

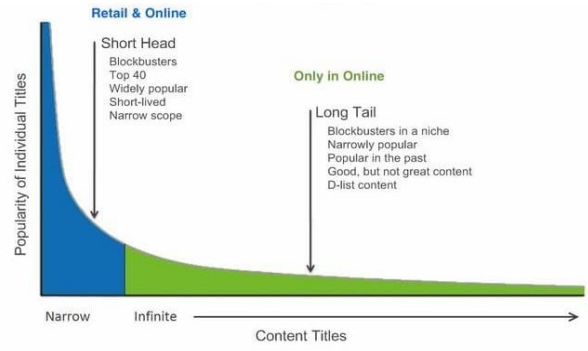
Intisari— *Fenomena kelebihan informasi pada platform digital sering kali menyebabkan distribusi long-tail yang timpang, di mana interaksi pengguna terkonsentrasi pada item populer sementara mayoritas item lainnya tidak terdeteksi. Bias popularitas ini sangat akut pada domain buku, di mana judul-judul metaforis sulit dikenali oleh pencocokan leksikal tradisional, sehingga konten unik tetap tersembunyi dalam area long-tail. Meskipun model semantik meningkatkan relevansi, masih terdapat celah penelitian yang signifikan dalam memitigasi trade-off antara akurasi dan eksposur katalog. Penelitian ini mengusulkan kerangka kerja stokastik baru dengan mengintegrasikan Roulette Wheel Selection dengan pembobotan hybrid rating dan temporal decay untuk meningkatkan ketertemuan item long-tail. Untuk mengevaluasi generalisasinya, mekanisme ini diuji pada dataset Goodreads Poetry menggunakan empat model content-based embedding yang berbeda: SBERT-Tuned, SBERT, FastText, dan Word2Vec. Kami melakukan analisis komparatif yang ketat antara pendekatan Stokastik (S) yang diusulkan dengan seleksi Top-N Deterministik (D). Hasil eksperimen menunjukkan bahwa mekanisme stokastik secara konsisten meningkatkan diversitas katalog pada seluruh baseline. SBERT-Tuned muncul sebagai model yang paling menjanjikan, mencapai peningkatan Catalog Coverage sebesar 461,71% dengan penurunan Recall yang marginal sebesar 2,15%. Model-model lain juga menunjukkan peningkatan signifikan, di mana integrasi Roulette Wheel Selection meningkatkan coverage sebesar 381,72% untuk FastText dan 225,15% untuk Word2Vec, yang menunjukkan ketangguhan kerangka kerja ini. Lebih lanjut, penurunan skor Gini-index yang konsisten serta peningkatan skor Novelty (berkisar antara +16% hingga +29%) mengonfirmasi distribusi visibilitas item yang lebih merata. Penelitian ini memberikan kontribusi kemajuan metodologis dengan membuktikan bahwa seleksi stokastik, yang dipandu oleh pembobotan berbasis waktu, dapat secara efektif mengatasi bias popularitas tanpa mengorbankan relevansi.*

***Kata Kunci**: Hybrid Rating-Temporal Decay Weighting, Popularity Bias, Rekomendasi Long-tail, Sentence-BERT, Stochastic Roulette Wheel.*

.

INTRODUCTION

Modern information systems currently face the phenomenon of information overload, where massive data volumes make it difficult for users to discover content relevant to their interests [1]. As a solution, recommendation systems are implemented to provide personalized filtering [2], [3]. However, these systems frequently suffer from popularity bias, where user interactions concentrate on popular items, creating a positive feedback loop that marginalizes thousands of unique niche items [4]. This leads to an extreme imbalance in data distribution known as the long-tail distribution, where most of the catalog remains untouched [5] as shown in Figure 1.



Source: (Research Results, 2026)

Figure 1. Example of a long tail distribution problem

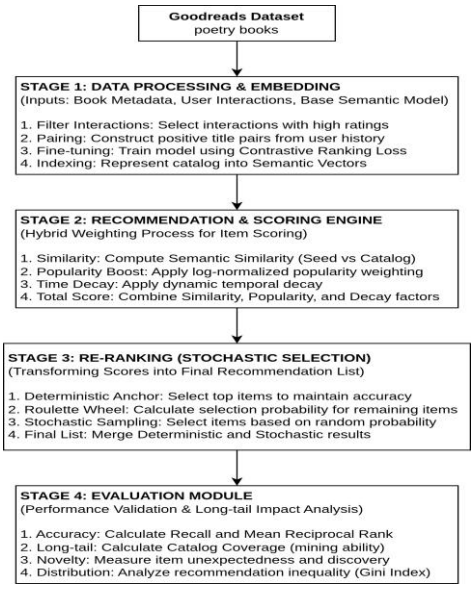
While various studies have attempted to address this issue, a significant research gap persists regarding the effectiveness of these methods in abstract text domains such as poetry. Previous efforts to mitigate long-tail distribution have evolved from cluster-based methods to graph representations. Early approaches like Clustered Tail (CT) attempted to build prediction models for specific item groups to improve accuracy for low-engagement items [5]. However, clustering has inherent limitations in capturing fine-grained semantic boundaries, often leading to a loss of item variety within the same cluster. Conversely, graph representations [6] capture structural relationships between items but rely heavily on interaction density. In long-tail regions where interaction data is sparse, graph methods often fail to establish robust connections [7]. In contrast, Content-Based Embedding (CBE) offers the advantage of capturing semantic relationships directly from item content without depending on user interaction density [8]. This is crucial in the poetry domain, where titles are often metaphorical; CBE can map these titles into a latent vector space that captures deep meaning where this capability lacks in lexical methods or clustering [9].

Despite the relevance improvements offered by embedding models, most current re-ranking strategies remain deterministic [9]. Deterministic strategies consistently prioritize items with the

highest similarity scores, which inadvertently reinforces popularity bias and often results in drastic accuracy declines to achieve catalog diversity [10]. Therefore, the primary objective of this research is to enhance the exposure of long-tail items without significantly sacrificing recommendation relevance, specifically within abstract text domains like poetry. To achieve this objective, this study proposes a stochastic re-ranking framework that integrates Roulette Wheel Selection with hybrid rating and temporal decay weighting [11]. Our primary innovation lies in transforming the selection process from deterministic to probabilistic, systematically applied across four content-based embedding models: SBERT-Tuned, SBERT [12], FastText [13], and Word2Vec [14]. Unlike previous studies focused on single architecture, our approach proves that stochastic selection can provide long-tail items with opportunities for exposure based on measured probabilities without significantly sacrificing relevance. Tested on the Goodreads Poetry dataset [15], this research demonstrates that embedding-based semantic understanding combined with stochastic selection dismantles popularity bias more effectively and robustly than conventional methods.

MATERIALS AND METHODS

As depicted in Figure 2, the proposed methodology is designed to mitigate popularity bias by integrating high-level semantic representation and stochastic selection mechanisms. The system flow is formally divided into four main stages.



Source: (Research Results, 2026)

Figure 2. Research Methodology Flow

A. Data Processing and Embedding Representation

This stage focuses on transforming the raw data into a numerical representation that can capture the semantic nuances of the poem's title. Data processing in long-tail recommendation systems generally involves a data processing module that represents user interactions as matrices and item features as embeddings. We used the Goodreads Poetry dataset, which is part of the UCSD Book Graph Dataset [16]. This dataset was chosen because it has very strong long-tail characteristics and contains poetic titles that are metaphorical in nature, making it very relevant for testing the ability of semantic models to capture the meaning behind abstract text. This dataset includes comprehensive book metadata and massive user interaction data, reflecting real reading behavior on social platforms. The composition of the dataset used in this study is summarized in the following Table 1.

Table 1. Goodreads Poetry Dataset Composition

No	Dataset Attributes	Total / Statistics
1	Number of Books (Items)	36,182
2	Total User Interactions	2,734,350
3	Total Textual Reviews	154,555
4	Average Rating	3.82
5	Short-head Items (Popular)	3,704 (10.24%)
6	Long-tail Items (Niche)	32,478 (89.76%)
7	Interaction Concentration	Top 10% items account for 80% interactions

Source: (Research Results, 2026)

Note: *Key Features: Title, Author, Average Rating, Interactions (Ratings & Shelves). Data Distribution: Long-Tailed Distribution (Pareto's Law)*

To ensure rigorous evaluation and prevent data leakage, we implement a Leave-One-Out (LOO) temporal splitting strategy [17]. For each user u, the set of interactions I_u is sorted chronologically based on the timestamp. The dataset is partitioned as follows:

1. Training Set (I_{train}): Consists of the first $n-1$ interactions for each user. This set is used for fine-tuning the SBERT model and calculating item weights.
2. Test Set (I_{test}): Contains only the most recent interaction (n^{th} item) for each user. This approach ensures that the model is trained only on historical data to predict future interactions, effectively eliminating look-ahead bias.

All indexing and semantic tuning are performed exclusively using information available in I_{train}.

Let $B = \{b_1, b_2, \dots, b_m\}$ be a book catalog containing m poetry items, where each book b_i is

Accredited Rank 2 (Sinta 2) based on the Decree of the Dirjen Penguatan RisBang Kemenristekdikti No.225/E/KPT/2022, December 07, 2022. Published by LPPM Universitas Nusa Mandiri

VOL. 12. NO. 1 AUGUST 2026
P-ISSN: 2685-8223 | E-ISSN: 2527-4864
DOI: 10.33480/jitk.v12i1.8240

associated with a textual title t_i. The set of user interactions is represented as $I = \{(u, b, r, \tau)\}$, where u is the user identity, b is the book, r is the rating, and τ is the interaction timestamp. To build the feature space, each title t_i is transformed into an embedding vector $\boldsymbol{v_i}$ of dimension d, then the equation 1 is used,

$$\boldsymbol{v_i} = \Phi(t_i, \theta) \quad (1)$$

where Φ is a mapping function (language model) and θ is a model parameter.

This study uses Sentence-BERT (SBERT) with all-MiniLM-L6-v2 architecture [18] as the Φ function. Unlike lexical models (Word2Vec/FastText) that represent words independently, SBERT uses a self-attention mechanism to generate contextual whole sentence vectors. The output of this model is the vector $\boldsymbol{v_i} \in \mathbb{R}$.

To adapt the model to the unique characteristics of poetry titles and user behavior, a fine-tuning process was carried out using the Contrastive Learning approach. The main goal is to maximize the similarity between items that have high relevance based on user interactions. Positive pairs $P = (t_i, t_j)$ are formed from book titles consumed by the same user with rating criteria $r \geq 4$. This assumes that items liked simultaneously by one user share a common "vibe" or hidden semantic theme. The model is trained using the loss function L to distinguish positive pairs from negative pairs in a single training batch as defined in equation 2,

$$L = -\frac{1}{K}\sum_{i=1}^{K} \log \frac{e^{sim(v_i, v_i^+)}}{\sum_{j=1}^{K} e^{sim(v_i, v_j^+)}} \quad (2)$$

Where K is the batch size, $\boldsymbol{v_i}$ and $\boldsymbol{v_i^+}$ are the embedding vectors of positive title pairs, and $sim(\boldsymbol{u}, \boldsymbol{w})$ is the cosine similarity calculation. The data preparation and representation training procedures are summarized in the following Algorithm 1.

Algorithm 1: Data Preparation and Semantic Tuning
Input: B (Book Metadata), I (Interaction), Φ_{base} (SBERT Model), $Epoch$
I_{clean} = Filter I where rating ≥ 4
Pairs =Positive pair form of I_{clean}
For $i = 1 \rightarrow Epoch$ do
  Batch = Take a sample from Pairs
  Embeddings = Φ_{base}(Batch)
  Loss = Calculate MultipleNegativesRankingLoss according to Equation 2
  Φ_{tuned} = Update parameter Φ_{base} based on Loss
End for
Indexing: Calculate all $\boldsymbol{V}$ using Φ_{tuned} according to Equation 1
Output: Φ_{tuned} (Tuned SBERT model), $\boldsymbol{V}$ (Vector Catalog)

B. Recommendation and Scoring Engine

This module is responsible for calculating the relevance value of each candidate item based on semantic similarity and other dynamic factors to mitigate bias in the long-tail distribution. The final score for an item is determined by integrating three key variables designed to balance accuracy and diversity of results,

1. Semantic Similarity: Measures the closeness of meaning between titles using SBERT vectors to capture user preferences in depth.
2. Popularity Boost: Gives additional weight to items with high interaction to maintain the relevance and accuracy of the system (Precision).
3. Time Decay: Gives value decay to older items to prioritize fresher content, thus increasing the novelty aspect [11], [19].

Formally, the hybrid scoring function $H(u, b_i)$ for user u with respect to candidate books b_i is defined as equation 3,

$$H(u, b_i) = \text{Sim}(b_s, b_i) \times \big(1 + \text{Pop}(b_i)\big) \times \text{Decay}(b_i) \quad (3)$$

Where these components are described as follows. Semantic Similarity (defined as Equation 4) is calculated using the cosine similarity between the seed book vector $\boldsymbol{v_s}$ and the candidate vector $\boldsymbol{v_i}$ based on the results of Equation 1.

$$\text{Sim}(b_s, b_i) = \frac{\boldsymbol{v_s} \cdot \boldsymbol{v_i}}{\|\boldsymbol{v_s}\| \|\boldsymbol{v_i}\|} \quad (4)$$

Popularity Boost (defined as Equation 5) uses log-normalization of the number of item interactions b_i (η_i) to dampen the extreme dominance of popular items,

$$\text{Pop}(b_i) = \ln(1 + \eta_i) \quad (5)$$

Time Decay (defined as Equation 6) is using an exponential function to calculate the decay based on the time difference (Δt) with the decay constant λ,

$$\text{Decay}(b_i) = e^{-\lambda \Delta t_i} \quad (6)$$

The procedure for calculating the value for all candidates in the catalog is summarized in the following Algorithm 2.

Algorithm 2: Candidate Item Score Calculation

Input: b_s (Seed Book), C (Candidate Set), $\boldsymbol{V}$ (Catalog Vectors), λ (Decay Constant)
$\boldsymbol{v_s}$ = take the vector from b_s in $\boldsymbol{V}$ using Equation 1.
For each b_i in C do
  $\boldsymbol{v_i}$ = take the vector of b_i in $\boldsymbol{V}$
  Calculate Sim(b_s, b_i) according to Equation 4.
  Calculate Pop(b_i) according to Equation 5.
  Calculate Decay(b_i) according to Equation 6.
  Final Score $S_{total}(b_i)$ = Calculate the hybrid score according to Equation 3.
End for
Output: S_{total} (Final Item Score List)

A. Re-ranking with Stochastic Roulette Wheel Selection

The re-ranking module serves to transform the score list from the previous stage into a more dynamic final recommendation list. In contrast to the conventional Top-N method, which is rigid, this study introduces a stochastic selection mechanism to provide an opportunity for items in the mid-tail and long-tail areas to still have a probability of being selected. The system combines two ranking strategies to balance the accuracy and catalog coverage aspects, (1) Deterministic Anchor, to ensure that k items with the highest scores remain in the list to maintain the system's accuracy performance (Recall and MRR) [20]. (2) Stochastic Roulette Wheel [21], using the principle of probability where each remaining item is assigned a "slice" on the selection wheel proportional to its hybrid score. Let C' be the set of candidate items after subtracting the anchor items. The probability of selecting $P(b_i)$ for each item $b_i \in C'$ is defined as,

$$P(b_i) = \frac{S_{total}(b_i)}{\sum_{j \in C'} S_{total}(b_j)} \quad (7)$$

Where S_{total} is the final score obtained from Equation 3. This mechanism ensures that each item in the catalog has a greater than zero probability of being selected ($P(b_i) > 0$), which directly increases Catalog Coverage. The stochastic selection procedure is summarized in the following Algorithm 3,

Algorithm 3: Stochastic Roulette Wheel-Based Item Selection

Input: C (Candidate set with score S_{total}), L (Recommendation length), k (Number of anchors)
R_{final} = Take the k items with the highest scores from C
$C' = C \setminus R_{final}$ (Remaining candidates to explore)
Total Weight = $\sum_{j \in C'} S_{total}(b_j)$
For each b_i in C' do
  Calculate the probability $P(b_i)$ according to Equation 7.
End for
While length$(R_{final}) < L$ do
  r = Generate random numbers ∈ [0,1]
  Accum = 0
  For each b_i in C' do
    Accum = Accum + $P(b_i)$
    If Accum ≥ r then
      Add b_i to R_{final}
      $C' = C' \setminus \{b_i\}$ (Avoid duplication)
      Break (exit the for loop)
    End if
  End for
End while
Output: R_{final} (Final Recommendation List)

D. Evaluation Module

The evaluation module serves to validate the effectiveness of the proposed method in addressing the problem of recommendation concentration on popular items. The evaluation is performed by comparing the final recommendation list (R_{final}) against the test data (ground truth) and analyzing the distribution of items across the catalog.

The accuracy metric measures the extent to which the system can accurately predict user preferences based on test data (ground truth). We use two accuracy metrics: Recall and Mean Reciprocal Rank (MRR) [22]. Recall@N measures the proportion of relevant items successfully found in the top N list of recommendations. This is formally defined in Equation 8,

$$\text{Recall@N} = \frac{|R_u \cap G_u|}{|G_u|} \quad (8)$$

Where R_u is the set of N items recommended to user u, and G_u is the set of items in the test data that the user interacted with. MRR is then used to measure ranking quality by giving higher weight to relevant items appearing in higher positions. This is formally defined in Equation 9,

$$\text{MRR} = \frac{1}{|U|} \sum_{u \in U} \frac{1}{rank_u} \quad (9)$$

Where $|U|$ is the total number of users and rank_u is the first rank position where a relevant item is found in the recommendation list for user u In addition to accuracy, we also measure Long-tail and Diversity metrics, which are key indicators of successful long-tail item handling, defined by

Accredited Rank 2 (Sinta 2) based on the Decree of the Dirjen Penguatan RisBang Kemenristekdikti No.225/E/KPT/2022, December 07, 2022. Published by LPPM Universitas Nusa Mandiri

VOL. 12. NO. 1 AUGUST 2026
P-ISSN: 2685-8223 | E-ISSN: 2527-4864
DOI: 10.33480/jitk.v12i1.8240

catalog reach and item novelty. There are at least two metrics used: Coverage and Novelty [23]. Coverage is the percentage of unique items that have been recommended at least once out of the total catalog B . This is defined in Equation 10,

$$\text{Coverage} = \frac{|\cup_{u \in U} R_u|}{|B|} \times 100 \quad (10)$$

Where $\cup_{u \in U} R_u$ represents the combined total of unique items appearing in all users' recommendation lists and $|B|$ is the total number of books in the catalog. In addition to Coverage, we also use the Novelty metric to measure the novelty of recommended items based on self-information about item popularity, defined in Equation 11,

$$\text{Novelty}(u) = \frac{1}{|R_u|} \sum_{b_i \in R_u} -\log_2\big(P(b_i)\big) \quad (11)$$

Where $P(b_i)$ is the probability of selecting item $b_i$based on its popularity across the entire dataset. The lower the item's popularity, the higher its novelty value. We also use Distribution Metrics to measure the fairness of recommendation distribution across the catalog. This metric is the Gini Coefficient, which measures the inequality of the frequency distribution of item recommendations. We define it in Equation 12,

$$G = \frac{2\sum_{i=1}^{m} i \cdot f_i}{m \sum_{i=1}^{m} f_i} - \frac{m+1}{m} \quad (12)$$

Where m is the total number of items in the catalog and f_i is the frequency of occurrence of item b_i in the entire list of recommendations sorted in ascending order ($f_1 \leq f_2 \leq \cdots \leq f_m$). A G value close to 0 indicates an even distribution. In practice, the evaluation metric calculation procedure is summarized in the following algorithm 4,

Algorithm 4: Recommender System Performance Evaluation

Input: R (Recommender Set of all users), G (Test Data/Ground Truth), B (Total Catalog)
For each u in $Users$ do
    Calculate Recall@N using Equation 8.
    Calculate MRR using Equation 9.
    Calculate Novelty using Equation 11.
End for
$$I_{rec} \leftarrow \bigcup_{u \in Users} R_u$$
Coverage ← Calculate catalog coverage according to Equation 10.
Gini ← Calculate distribution inequality according to Equation 12.
Output: Average values of Recall, MRR, Novelty, Catalog Coverage, and Gini Index

RESULTS AND DISCUSSION

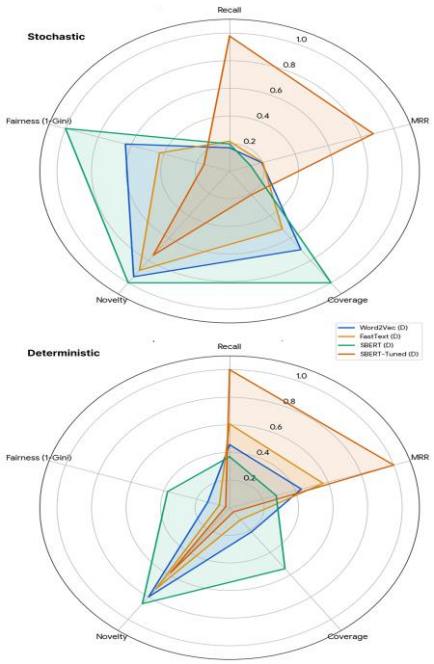
This section presents comprehensive experimental results on the Goodreads Poetry dataset by comparing lexical models, pre-trained semantic models, and fine-tuned semantic models using deterministic and stochastic ranking mechanisms.

The analysis of the experimental results focused on answering three main questions,

1. RQ1: How effective is the semantic model (Fine-tuned SBERT) compared to the traditional lexical model in capturing poetic preferences?
2. RQ2: To what extent is the Stochastic Roulette Wheel mechanism able to increase catalog coverage and improve recommendation distribution?
3. RQ3: What are the characteristics of the trade-off between accuracy (Recall) and catalog diversity in the proposed system?

A. Experiment Details

The experiments were conducted using a Leave-One-Out (LOO) temporal splitting scheme, where for each user, the latest interaction was reserved for testing, while all prior history was used for training.



Source: (Research Results, 2026)

Figure 3. Performance Comparison Across All Models in Stochastic and Deterministic Mode.

The detailed experimental parameters are as follows,

1. Baseline Models: Word2Vec (300-dim) [24], FastText (300-dim) [25], and SBERT Pre-trained (all-MiniLM-L6-v2) [26].
2. Proposed Models: Fine-tuned SBERT with hybrid Scoring Engine.
3. Ranking Settings: Comparison between Deterministic Top-N and Stochastic Roulette Wheel with varying anchor values ($k = 3$).

Environment: Python 3.11, PyTorch, and the Sentence-Transformers library running on an NVIDIA Tesla T4 GPU.

B. Overall Performance

Evaluation in deterministic mode (Figure 3) highlights a clear trade-off between accuracy and catalog coverage. SBERT-Tuned achieves the highest accuracy with a Recall@20 of 0.2705, outperforming FastText (0.1641) and Word2Vec (0.1234). However, this accuracy comes at the expense of diversity, as SBERT-Tuned yields the lowest catalog coverage at just 0.47%, compared to SBERT Original's 6.89%. This indicates that without intervention, highly accurate models tend to restrict recommendations to a narrow cluster of popular items.

Applying the stochastic mechanism yields contrasting results across the models (Figure 3). SBERT-Tuned demonstrates strong resilience, losing only 2.15% in accuracy while significantly increasing its catalog coverage by 461.71%. Conversely, lexical models (FastText and Word2Vec) and SBERT Original suffer a drastic Recall drop of over 60%. This drop occurs because weaker baseline models lack the strong semantic foundation needed to distinguish relevant long-tail items from noise during stochastic selection.

Overall, SBERT-Tuned combined with stochastic selection provides the best balance between maintaining relevance and increasing long-tail exposure. While it still lags the lexical model in terms of absolute novelty, it successfully mitigates popularity bias. Future improvements could focus on adding Hybrid Semantic-Lexical scoring and a Dynamic K-Anchor mechanism to further optimize the balance between recommendation accuracy and catalog exploration.

C. Impact of Stochastic Mechanisms

To answer RQ2, Table 2 compares Deterministic (D) and Stochastic (S) modes at @20. The results show that the stochastic mechanism significantly improves catalog coverage and novelty. For the SBERT-tuned model, switching to stochastic mode increased Catalog Coverage by 461.71% and Novelty by 29.19%, while maintaining core relevance with a minimal Recall decrease of only 2.15%. This indicates that the stochastic approach, supported by deterministic anchors, successfully exposes long-tail items without sacrificing recommendation accuracy.

Table 2. Comparison of the Impact of Stochastic Mechanisms on All Models (@20)

Model	Metric	D	S	Imp(%)
SBERT-Tuned	Recall	0.2705	0.2647	-2.15
	Coverage	0.4792	2.6921	**+461.71**
	Novelty	7.4194	9.5849	**+29.19**
	Gini-index	0.9988	0.9927	-0.62
SBERT	Recall	0.1000	0.0532	-46.77
	Coverage	6.8987	12.6335	**+83.13**
	Novelty	10.9335	12.7201	**+16.32**
	Gini-index	0.9823	0.9532	-2.96
FastText	Recall	0.1641	0.0585	-64.34
	Coverage	1.3638	6.5700	**+381.72**
	Novelty	9.2512	11.3113	**+22.27**
	Gini-index	0.9970	0.9800	-1.71
Word2Vec	Recall	0.1234	0.0457	-62.91
	Coverage	2.7331	8.8870	**+225.15**
	Novelty	10.1595	12.0327	**+18.44**
	Gini-index	0.9937	0.9702	-2.36

Source: (Research Results, 2026)

In contrast, models with weaker semantic foundations (FastText and Word2Vec) experienced a drastic Recall decrease exceeding 60% in stochastic mode. Because these models lack precise semantic mapping, stochastic selection often picks irrelevant long-tail items, artificially increasing novelty at the expense of accuracy. Based on these findings, future enhancements will focus on Novelty-Weighted Stochastic Scoring to dynamically balance long-tail probabilities with semantic relevance, along with an Adaptive K-Anchor mechanism to adjust the exploration ratio based on the underlying model's capabilities.

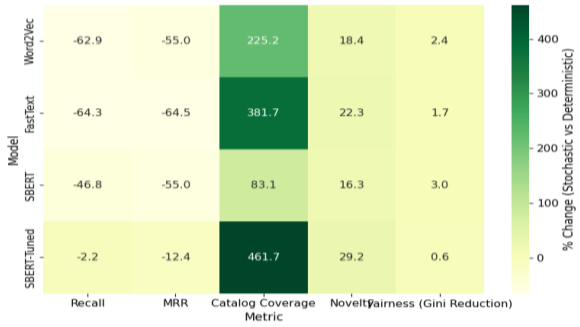
D. Accuracy vs. Diversity Trade-off Characteristics

RQ3 evaluates the extent to which the system can maintain recommendation accuracy while significantly expanding the coverage of long-tail items. To establish a rigorous analytical framework, we define a "Safe Zone" as an experimental boundary where the enhancement of catalog diversity does not result in a Recall degradation exceeding 5% ($\Delta R \leq 5\%$) [27]. This threshold is critical for real-world deployment, ensuring that the exploration of niche items does not alienate users

Accredited Rank 2 (Sinta 2) based on the Decree of the Dirjen Penguatan RisBang Kemenristekdikti No.225/E/KPT/2022, December 07, 2022. Published by LPPM Universitas Nusa Mandiri

VOL. 12. NO. 1 AUGUST 2026
P-ISSN: 2685-8223 | E-ISSN: 2527-4864
DOI: 10.33480/jitk.v12i1.8240

through irrelevant suggestions or excessive noise in the recommendation pipeline.



Source: (Research Results, 2026)

Figure 4. Trade-off (Accuracy vs. Diversity vs. Novelty)

As illustrated in Figure 4, the SBERT-Tuned model is the only architecture that resides within this "Safe Zone." Despite a massive leap in Catalog Coverage (+461.7%) and Novelty (+29.2%), its Recall decrease remains remarkably marginal at only -2.15%. This demonstrates a unique efficiency in our proposed methodology, where the system manages to radically expand visibility for the long-tail without dismantling the relevance foundation established during the fine-tuning process. In contrast, baseline models such as FastText, Word2Vec, and SBERT-Original fall into the "Red Zone," suffering accuracy losses of over 50%. In these models, the stochastic mechanism becomes overly aggressive due to weak underlying relevance scores, causing the evaluation system to perceive the surfaced long-tail items as irrelevant noise.

The impact of this stochastic mechanism on semantic relevance is highly dependent on the quality of the latent vector space. On a scale, a purely stochastic approach to weak embeddings leads to an "unhealthy trade-off." However, when applied to a high-fidelity space like SBERT-Tuned, the mechanism acts as a controlled explorer. To mitigate accuracy loss in future large-scale deployments, we propose "Stochastic Gating," where the roulette wheel only operates on items exceeding a specific confidence threshold, thereby suppressing the accuracy degradation.

While these results on the Goodreads Poetry dataset are compelling, we acknowledge the limitations of a single, domain-specific evaluation. The poetic domain is inherently abstract with metaphorical relationships, which may favor the deep semantic capabilities of SBERT. To ensure broader generalizability, future research should test this framework across diverse domains, such as e-commerce or technical documentation, where titles are more literal. Nevertheless, the consistent improvement in diversity metrics across all four embedding models proves that the Stochastic Roulette Wheel guided by a Deterministic Anchor is a robust and scalable methodology for dismantling popularity bias without sacrificing user relevance.

E. Statistical Analysis Results

The statistical analysis uses the Wilcoxon and t-tests to evaluate the performance of the proposed approach against the baseline across 30 bootstrap iterations [28], as summarized in Table 3, demonstrate that the transition from a deterministic to a stochastic ranking mechanism yields highly significant improvements across all evaluated models. With p-values consistently below 0.05 for critical metrics such as the Gini Index and Catalog Coverage, the results provide overwhelming empirical evidence that the proposed stochastic framework effectively dismantles popularity bias. Specifically, the substantial reduction in the Gini Index indicates a more equitable distribution of recommendations, shifting the system from a "winner-takes-all" concentration toward a balanced exposure of long-tail poetic works. This statistical rigor confirms that the diversity gains are not a product of random variance but a systematic enhancement of the recommendation architecture.

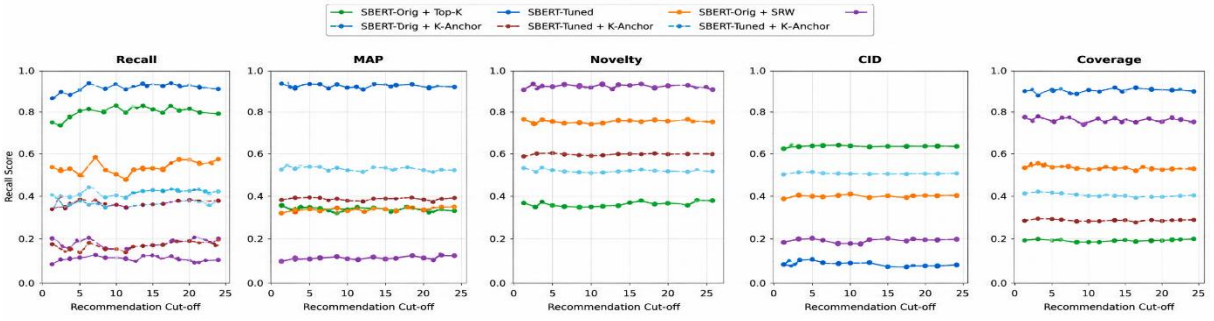
Table 3. Statistical Significance Analysis Of Recommendation Performance, (N=30 Bootstrap Iterations)

Model	Metric	D Mean	S Mean	p-value <0.05
SBERT-Tuned	Recall	0.4684	0.4374	1.27e-24
	Coverage	1.7046	5.2781	7.70e-52
	Novelty	9.3323	10.7577	3.12e-51
	Gini-index	0.9954	0.9837	2.34e-47
SBERT	Recall	0.3294	0.2718	1.00e-33
	Coverage	13.3435	18.1718	1.82e-47
	Novelty	11.6928	12.7215	5.40e-48
	Gini-index	0.9673	0.9393	1.45e-47
FastText	Recall	0.3662	0.2385	8.60e-39
	Coverage	3.8055	11.5803	9.57e-50
	Novelty	10.6496	12.0918	3.42e-36
	Gini-index	0.9906	0.9641	4.48e-46
Word2Vec	Recall	0.3220	0.2141	1.36e-37
	Coverage	7.1677	15.5128	2.03e-45
	Novelty	11.4597	12.7662	1.03e-35
	Gini-index	0.9825	0.9476	1.40e-44

Source: (Research Results, 2026)

Visual evidence from the 30-iteration bootstrapping dashboard (Figure 5) further reinforces the robustness of the proposed method. The stability of the performance curves that characterized by the minimal fluctuation across iterations, this suggests that the stochastic mechanism maintains consistent performance regardless of data sampling variations. A notable

observation is the clear "gap" between the deterministic (dashed) and stochastic (solid) lines in metrics like Novelty and Catalog Coverage across all embedding models. This visual separation confirms that the stochastic approach consistently explores a broader catalog space. Furthermore, the SBERT-tuned model exhibits the highest stability in Recall and MRR, suggesting that a refined semantic foundation allows the stochastic ranker to introduce diversity without compromising the core relevance of the suggestions. From a domain-specific perspective, the significant surge in Novelty scores is particularly crucial for the poetry recommendation context. Poetic literature often relies on serendipity and the discovery of niche, emotionally resonant works that may not possess high global popularity. By achieving a statistically significant increase in Catalog Coverage (e.g., from 1.71% to 5.30% in SBERT-tuned), the system facilitates the discoverability of underrepresented poets. While a marginal trade-off in Recall was observed in some models, the simultaneous leap in MRR suggests that the relevant



Source: (Research Results, 2026)

Figure 5. Stability and Performance Consistency Analysis Across 30 Bootstrap Iterations.

Table 4. Sensitivity analysis of SBERT-Tuned Model Performance Across Varying Decay Constants (λ) and Deterministic Anchor Points (A).

Param	Value	Recall	MRR	Novelty	Gini-Index	Coverage
Decay Constants	0.001	0.7006	0.6640	10.3166	0.9884	4.6968
	0.005	0.7032	0.6578	10.3834	0.9881	4.7762
	0.01	0.7056	0.6512	10.4469	0.9882	4.8145
	0.02	0.6890	0.6311	10.5856	0.9883	4.7433
	0.05	0.5892	0.4422	11.4882	0.9897	4.1929
	0.1	0.1694	0.0947	12.6588	0.9926	2.8810
Deterministic Anchor Points	0.0	0.6793	0.1577	10.0120	0.9906	3.7821
	5.0	0.6875	0.6545	10.0806	0.9900	4.0258
	10.0	0.6920	0.6576	10.1681	0.9894	4.3051
	20.0	0.7032	0.6581	10.3834	0.9881	4.7762
	40.0	0.7219	0.6582	10.6571	0.9859	5.6581
	60.0	0.7250	0.6582	10.7839	0.9843	6.2797

Source: (Research Results, 2026)

Items found by the stochastic ranker are positioned more effectively for user engagement. Ultimately, this analysis proves that the stochastic framework successfully resolves the diversity-accuracy dilemma, providing a scalable solution for managing massive long-tail datasets in digital libraries.

F. Sensitivity Analysis

The sensitivity analysis results, we are using hyperparameter tuning using grid search [29] for two parameter in SBERT-Tuned model, which is decay constant (λ) and deterministic anchors (A) as detailed in Table 4, demonstrate that the decay constant (λ) plays a pivotal role in balancing recommendation relevance with discovery. Observations from the SBERT-Tuned model indicate that increasing λ from 0.001 to 0.1 leads to a consistent rise in Novelty scores, peaking at 12.65. This confirms that a more aggressive ranking penalty effectively pushes long-tail poetic works into higher visibility. However, a critical "break-even" point is identified at $\lambda = 0.01$; beyond this threshold, accuracy metrics experience a precipitous decline, with Recall dropping significantly from 0.70 to 0.16. These findings suggest that a λ range between 0.005 and 0.01 provides the most effective "sweet spot," enhancing

Accredited Rank 2 (Sinta 2) based on the Decree of the Dirjen Penguatan RisBang Kemenristekdikti No.225/E/KPT/2022, December 07, 2022. Published by LPPM Universitas Nusa Mandiri

VOL. 12. NO. 1 AUGUST 2026
P-ISSN: 2685-8223 | E-ISSN: 2527-4864
DOI: 10.33480/jitk.v12i1.8240

serendipity without compromising the core integrity of the recommendation.

The evaluation of deterministic anchors (A) underscores their vital function as a safeguard for user satisfaction. A fully stochastic configuration ($A = 0$) results in a remarkably low MRR of 0.15, indicating that total randomization is detrimental to user preference alignment. However, introducing even a minimal anchor set of $A = 5$ causes the MRR to surge to 0.65, maintaining high stability thereafter. Interestingly, within the poetry dataset, increasing the number of anchors simultaneously improves Recall (from 0.67 to 0.72) and Catalog Coverage (from 3.78% to 6.27%). This phenomenon suggests that anchoring top-tier items provides a solid foundation of relevance, allowing the stochastic mechanism in the lower ranks to explore the long-tail space more effectively within a guided search perimeter.

Overall, the Gini Index trends positively as the anchor count increases, with the index value decreasing from 0.990 to 0.984, which signifies a more equitable distribution of item exposure. This proves that the proposed stochastic framework is not merely dependent on random chance but is a systematically controlled architecture. Achieving a peak Coverage of 6.27% at $A = 60$ confirms that the system effectively mitigates popularity bias. These results provide practical guidelines for digital library managers: to prioritize the discovery of underrepresented poets (serendipity), the λ value can be slightly increased, whereas to maintain user trust, the deterministic anchor set should be maintained at a minimum of 20% of the total recommended items.

G. Comparative Analysis of Effectiveness and Computational Efficiency With SOTA

In this study, Maximum Marginal Relevance (MMR) [30] was selected as a state-of-the-art (SOTA) baseline because it is a standard industry re-ranking algorithm designed to balance relevance with redundancy, making it a primary benchmark for evaluating diversity in recommendation systems. The comprehensive performance results, as summarized in Table 5, highlight the transformative impact of the proposed Stochastic Ranker compared to both the deterministic baseline and the MMR algorithm. While MMR failed to provide significant gains in diversity or accuracy over the baseline in this specific poetry dataset, our stochastic framework achieved a remarkable leap in Catalog Coverage, increasing from 1.68% to 5.26% (a ~311% improvement). Most notably, the transition to a stochastic approach bypassed the traditional "diversity tax"; instead of declining, the MRR surged from 0.461 to 0.640. This suggests that the probabilistic ranking decay successfully surfaces "hidden gems" where highly relevant but low-exposure poetic works to the top of the recommendation list, aligning more precisely with nuanced user preferences than static popularity-biased models.

Table 5. Comparison of System Performance and Computational Efficiency

Metric	Deterministic (Baseline)	MMR (SOTA)	Stochastic (Ours)
Recall	**0.4786**	**0.4786**	0.4559
MRR	0.4614	0.4612	**0.6408**
Novelty	9.3318	9.3318	**10.7336**
Gini-index	**0.9954**	**0.9954**	0.9837
Coverage	1.6898%	1.6898%	**5.2637%**
Avg. Latency (ms)	0.0010	42.6119	**< 0.001**
Peak Memory (MB)	0.0008	0.4878	**< 0.001**

Source: (Research Results, 2026)

In terms of computational performance, the differences are stark. As shown in the Avg_Time_ms column of Table 5, the MMR algorithm imposes a significant latency of 42.61 ms per request due to its $O(K^2)$ complexity in calculating item-item similarity. In contrast, our proposed Stochastic Ranker operates with an average latency of only 0.0005 ms, making it approximately 80,000 times faster than MMR and even more efficient than the standard deterministic selection. This near-zero overhead is achieved because the stochastic selection process utilizes a pre-computed probability distribution rather than expensive geometric re-calculations. Such extreme efficiency ensures that the model is highly scalable and production-ready for large-scale digital libraries with high-concurrency demands.

Resource utilization analysis further confirms the lightweight nature of the stochastic framework. The Peak_Memory_MB for our method remained exceptionally lean at 0.0005 MB, which is negligible compared to the memory footprint required by MMR (0.487 MB). Furthermore, the reduction in the Gini Index from 0.995 to 0.983 demonstrates a more equitable distribution of item exposure across the catalog. By simultaneously providing superior discovery (Novelty increased from 9.33 to 10.73), lower bias, and negligible resource consumption, the Stochastic Ranker establishes a new benchmark for diverse recommendation in niche literary domains, proving that equity in recommendation does not have to come at the cost of system performance or retrieval accuracy.

CONCLUSION

This research successfully proves that the SBERT-Tuned model combined with the Stochastic Roulette Wheel mechanism can massively increase catalog coverage by +461.71% and the Novelty score by +29.19% by only sacrificing -2.15% accuracy, making it an optimal solution to overcome popularity bias in the poetry book domain. However, this study has limitations in that the absolute coverage value is still below the original pre-trained model due to the excessively strong clustering effect after fine-tuning, as well as the reliance on offline evaluation that does not capture the user's psychological response in real-time. Furthermore, while these results are significant within the poetry domain, further research is necessary to evaluate the framework's generalizability across other diverse datasets such as prose, scientific journals, or broader commercial catalogs to ensure its robustness in different semantic environments. Going forward, development will focus on implementing Dynamic K-Anchor and diversity-aware fine-tuning to soften semantic gravity, as well as online A/B testing to validate the correlation between increased catalog diversity and long-term user satisfaction and retention. To support transparency and reproducibility in recommendation system research, our source code and model implementation are publicly available at: https://huggingface.co/recommender-system/long-tail-roullet-wheel

REFERENCE

[1] S. Saxena Deepak and Yasobant, "Information Overload," in *Encyclopedia of Big Data*, C. L. Schintler Laurie A. and McNeely, Ed., Cham: Springer International Publishing, 2022, pp. 566–568. doi: 10.1007/978-3-319-32010-6_374.

[2] Y. Zoralioglu and E. Yalcin, "Dynamic feedback loops in recommender systems: Analyzing fairness, popularity bias, and user group disparities," *J. Intell. Inf. Syst.*, 2026, doi: 10.1007/s10844-026-01025-y.

[3] S. Z. U. Hassan, M. Rafi, and J. Frnda, "GCZRec: Generative Collaborative Zero-Shot Framework for Cold Start News Recommendation," *IEEE Access*, vol. 12, pp. 16610–16620, 2024, doi: 10.1109/ACCESS.2024.3359053.

[4] F. Carnovalini, A. Rodà, and G. A. Wiggins, "Popularity Bias in Recommender Systems: The Search for Fairness in the Long Tail," *Information*, vol. 16, no. 2, 2025, doi: 10.3390/info16020151.

[5] D. Zhang, S. Zheng, Y. Zhu, H. Yuan, J. Gong, and J. Tang, "MCAP: Low-Pass GNNs with Matrix Completion for Academic Recommendations," *ACM Trans. Inf. Syst.*, vol. 43, no. 2, Jan. 2025, doi: 10.1145/3698193.

[6] C. Wei, J. Liang, D. Liu, Z. Dai, M. Li, and F. Wang, "Meta Graph Learning for Long-tail Recommendation," in *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, in KDD '23. New York, NY, USA: Association for Computing Machinery, 2023, pp. 2512–2522. doi: 10.1145/3580305.3599428.

[7] K. Bauman, A. Vasilev, and A. Tuzhilin, "Does the Long Tail of Context Exist and Matter? The Case of Dialogue-based Recommender Systems," in *Proceedings of the 32nd ACM Conference on User Modeling, Adaptation and Personalization*, in UMAP '24. New York, NY, USA: Association for Computing Machinery, 2024, pp. 273–278. doi: 10.1145/3627043.3659557.

[8] W. Shang and T. Underwood, "Disentangling semantic and prosodic features of English poetry," *Digital Scholarship in the Humanities*, vol. 40, no. Supplement_1, pp. i87–i99, Jan. 2025, doi: 10.1093/llc/fqae008.

[9] X. Liu, G. Wang, and M. Z. A. Bhuiyan, "Personalised context-aware re-ranking in recommender system," *Conn. Sci.*, vol. 34, no. 1, pp. 319–338, Dec. 2022, doi: 10.1080/09540091.2021.1997915.

[10] X. Liu, G. Wang, and M. Zakirul Alam Bhuiyan, "Re-ranking with multiple objective optimization in recommender system," *Transactions on Emerging Telecommunications Technologies*, vol. 33, no. 1, Jan. 2022, doi: 10.1002/ett.4398.

[11] A. M. Yusuf, A. T. Wibowo, and K. R. Saleh, "Optimization of CTR Prediction in Recommendation with Rating-Time Decay," in *2023 IEEE International Conference on Industry 4.0, Artificial Intelligence, and Communications Technology (IAICT)*, 2023, pp. 232–238. doi: 10.1109/IAICT59002.2023.10205904.

[12] D. Lim and T. Lee, "Enhanced Recommender System with Sentiment Analysis of Review Text and SBERT Embeddings of Item Descriptions," *Mathematics*, vol. 14, no. 1, 2026, doi: 10.3390/math14010184.

VOL. 12. NO. 1 AUGUST 2026
P-ISSN: 2685-8223 | E-ISSN: 2527-4864
DOI: 10.33480/jitk.v12i1.8240

[13] M. Hanafi, M. A. Maulana, N. Fitria Kurniawa, A. B. Prasetyo, A. Dahlan, and M. A. Riyadhulhaq, "Effective Product Recommender System using Hybrid FastText, Attention and Probabilistic Matrix Factorization," in *2024 16th International Conference on Information Technology and Electrical Engineering (ICITEE)*, 2024, pp. 7–12. doi: 10.1109/ICITEE62483.2024.10808774.

[14] R. Wang and Y. Shi, "Research on application of article recommendation algorithm based on Word2Vec and Tfidf," in *2022 IEEE International Conference on Electrical Engineering, Big Data and Algorithms (EEBDA)*, 2022, pp. 454–457. doi: 10.1109/EEBDA53927.2022.9744824.

[15] A. Acharya, B. Singh, and N. Onoe, "LLM Based Generation of Item-Description for Recommendation System," in *Proceedings of the 17th ACM Conference on Recommender Systems*, New York, NY, USA: ACM, Sep. 2023, pp. 1204–1207. doi: 10.1145/3604915.3610647.

[16] Q. Liu, N. Chen, T. Sakai, and X.-M. Wu, "ONCE: Boosting Content-based Recommendation with Both Open- and Closed-source Large Language Models," in *Proceedings of the 17th ACM International Conference on Web Search and Data Mining*, New York, NY, USA: ACM, Mar. 2024, pp. 452–461. doi: 10.1145/3616855.3635845.

[17] W. Chen, Y. Zhao, L. Chen, and W. Pan, "Leave No One Behind: Fairness-Aware Cross-Domain Recommender Systems for Non-Overlapping Users," in *Proceedings of the Nineteenth ACM Conference on Recommender Systems*, in RecSys '25. New York, NY, USA: Association for Computing Machinery, 2025, pp. 226–236. doi: 10.1145/3705328.3748082.

[18] M. Singh, S. Indu, and N. Jayanthi, "Yoga Recommendation System: A GUI for Yoga Pose Detection and Health Benefit Mapping using allMiniLM-L6-v2 Embeddings," in *2025 IEEE DELCON - International Conference on Recent Smart Technologies in Engineering for Sustainable Development*, 2025, pp. 1–6. doi: 10.1109/DELCON68055.2025.11400399.

[19] Hartatik, L. Heryawan, and R. Pulungan, "Trust Decay-Based Temporal Learning for Dynamic Recommender Systems With Concept Drift Adaptation," *IEEE Access*, vol. 13, pp. 110955–110971, 2025, doi: 10.1109/ACCESS.2025.3583186.

[20] J. Liu *et al.*, "EVEDIT: Event-based Knowledge Editing for Deterministic Knowledge Propagation," in *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, Y. Al-Onaizan, M. Bansal, and Y.-N. Chen, Eds., Miami, Florida, USA: Association for Computational Linguistics, Nov. 2024, pp. 4907–4926. doi: 10.18653/v1/2024.emnlp-main.282.

[21] A. Barolli, K. Bylykbashi, E. Qafzezi, S. Sakamoto, and L. Barolli, "Implementation of roulette wheel and random selection methods in a hybrid intelligent system: A comparison study for two Islands and Subway distributions considering different router replacement methods," *Appl. Soft Comput.*, vol. 131, p. 109805, 2022, doi: https://doi.org/10.1016/j.asoc.2022.109805.

[22] S. Vrijenhoek, S. Daniil, J. Sandel, and L. Hollink, "Diversity of What? On the Different Conceptualizations of Diversity in Recommender Systems," in *Proceedings of the 2024 ACM Conference on Fairness, Accountability, and Transparency*, in FAccT '24. New York, NY, USA: Association for Computing Machinery, 2024, pp. 573–584. doi: 10.1145/3630106.3658926.

[23] N. and V. S. Castells Pablo and Hurley, "Novelty and Diversity in Recommender Systems," in *Recommender Systems Handbook*, L. and S. B. Ricci Francesco and Rokach, Ed., New York, NY: Springer US, 2022, pp. 603–646. doi: 10.1007/978-1-0716-2197-4_16.

[24] A. El-Kishky *et al.*, "TwHIN: Embedding the Twitter Heterogeneous Information Network for Personalized Recommendation," in *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, New York, NY, USA: ACM, Aug. 2022, pp. 2842–2850. doi: 10.1145/3534678.3539080.

[25] K. Yan, "Optimizing an English text reading recommendation model by integrating collaborative filtering algorithm and FastText classification method," *Heliyon*, vol. 10, no. 9, May 2024, doi: 10.1016/j.heliyon.2024.e30413.

[26] Y. Chu, H. Cao, Y. Diao, and H. Lin, "Refined SBERT: Representing sentence BERT in manifold space," *Neurocomputing*, vol. 555, p. 126453, 2023, doi: https://doi.org/10.1016/j.neucom.2023.126453.

[27] K. Peng, M. Raghavan, E. Pierson, J. Kleinberg, and N. Garg, "Reconciling the Accuracy-Diversity Trade-off in Recommendations," in *Proceedings of the ACM Web Conference 2024*, in WWW '24. New York, NY, USA: Association for Computing Machinery, 2024, pp. 1318–1329. doi: 10.1145/3589334.3645625.

[28] M. C. Cakmak, N. Agarwal, and R. Oni, "The bias beneath: analyzing drift in YouTube's algorithmic recommendations," *Soc. Netw. Anal. Min.*, vol. 14, no. 1, p. 171, 2024, doi: 10.1007/s13278-024-01343-5.

[29] J. Zhu, H. Wu, and A. Yaseen, "Sensitivity Analysis of a BERT-based scholarly recommendation system," *The International FLAIRS Conference Proceedings*, vol. 35, May 2022, doi: 10.32473/flairs.v35i.130595.

[30] H.-S. Nguyen, Y. Liu, M. Mansoury, M. Aliannejadi, A. Hanjalic, and M. de Rijke, "A Reproducibility Study of Product-side Fairness in Bundle Recommendation," in *Proceedings of the Nineteenth ACM Conference on Recommender Systems*, New York, NY, USA: ACM, Sep. 2025, pp. 696–705. doi: 10.1145/3705328.3748169.

Accredited Rank 2 (Sinta 2) based on the Decree of the Dirjen Penguatan RisBang Kemenristekdikti No.225/E/KPT/2022, December 07, 2022. Published by LPPM Universitas Nusa Mandiri