Improving Cold-Start Recommendations through Active Learning and Strategic Item Clustering

Devam Sanjay Sheth
Department of Computer Science, College of Science
San Jose State University
San Jose, US
devamsanjay.sheth@sjsu.edu, SJSUID: 017404218

***Abstract*—This paper tackles the cold-start problem in recommender systems by combining matrix factorization with active learning and strategic clustering of candidate items. A Singular Value Decomposition (SVD) based collaborative filtering model was used to model user-item interactions. At the same time, a hybrid scoring mechanism was developed to assess the impact of item selection on recommendation quality. Two active learning strategies, Y-change and error-change, were implemented and evaluated under three selection variants: risky, moderate, and conservative. Before ranking, candidate items were pre-clustered using interaction frequency to ensure diversity and mitigate sparsity. Experimental evaluation on a real-world e-Commerce dataset demonstrated that the conservative variant of the error-change strategy consistently outperformed its counterparts. These findings highlight the effectiveness of cautious, cluster-aware item selection within a hybrid active learning framework for improving cold-start recommendations.**

***Index Terms*—Cold start, recommender system, active learning, Singular Value Decomposition, Hybrid scoring**

I. INTRODUCTION

With the exponential growth of digital content and online services, users are increasingly confronted with an overwhelming number of options. Recommender systems play a crucial role in reducing this information overload by delivering personalized content based on a user's past behavior and preferences. These systems are now ubiquitous across domains such as e-commerce, video streaming, news aggregation, and social media. The performance of most recommendation algorithms, particularly collaborative filtering models, hinges on the availability of historical user-item interaction data. When this data is sparse or missing—as in the case of new users or items—the system's ability to make accurate predictions significantly declines. This issue is widely known as the *cold-start problem*.

The cold-start problem manifests in two primary forms: the *user cold-start problem*, where the system has insufficient data on a newly registered user, and the *item cold-start problem*, where a newly added item lacks sufficient interaction history to be recommended effectively. This work focuses specifically on the user cold-start problem, which poses a substantial barrier to user engagement in the early stages of system adoption.

Traditional collaborative filtering approaches, including matrix factorization models, are unable to generate personalized recommendations for cold users due to their reliance on interaction histories. To overcome this, various strategies have been proposed, including the use of content-based features, demographic data, and hybrid models. However, these solutions often require extensive side information, which may not always be available or reliable.

An alternative approach to addressing the cold-start problem involves the use of *active learning*. Active learning empowers the system to actively engage the user by selecting a curated set of items and requesting feedback. This selective querying process aims to elicit the most informative ratings, thereby enabling the recommender system to quickly build an initial user profile. Within the scope of active learning, different selection strategies have been explored. Two notable techniques are *Y-change*, which evaluates how predictions change when a new item is rated, and *Error-change*, which estimates the reduction in predictive error when the new rating is incorporated.

These strategies can be configured under different behavioral assumptions to handle uncertainty in user responses. The *conservative* variant minimizes the risk of selecting uninformative items by assuming the least beneficial user feedback, while the *risky* variant assumes highly impactful feedback, potentially leading to greater gains but also higher variance. The *moderate* variant averages over all possible outcomes, striking a balance between the two extremes.

While active learning can significantly improve cold-start recommendations, its effectiveness depends heavily on the composition of the candidate item set. Presenting redundant or similar items may lead to diminished feedback utility. To address this, a clustering-based approach is used in this work, where *K-means clustering* is applied to the item embeddings generated through matrix factorization. This step ensures that only representative and diverse items are selected for potential user feedback, enhancing the informativeness of the active learning process.

Furthermore, to combine the strengths of Y-change and Error-change strategies, a *hybrid scoring function* is introduced. This function blends the two approaches using a weighting parameter α, which reflects the system's confidence in the contribution of each strategy. Rather than relying on manual tuning, the optimal value of α is learned automatically through an *Expectation-Maximization (EM)* algorithm, allowing for data-driven adaptability based on observed user interactions.

This study proposes a comprehensive framework that brings together matrix factorization, item clustering, and hybrid active learning. The framework is empirically evaluated on a real-world e-commerce dataset, simulating cold-start scenarios by withholding interaction data for selected users. The performance of each strategy and its variants is assessed in terms of prediction accuracy and hybrid error reduction. Through extensive experimentation, the study demonstrates that strategically chosen items, combined with adaptive hybrid scoring and clustering, can significantly improve the initial recommendation quality for cold users.

II. Related Work

The cold-start problem in recommender systems has long posed a challenge due to the absence of sufficient interaction data from new users or on newly introduced items. Solutions to this issue span content-based filtering, hybrid recommendation models, and increasingly, data-efficient strategies such as active learning and item clustering. This section reviews key contributions across these categories and situates the present work within the broader research landscape.

Matrix factorization is one of the foundational techniques in collaborative filtering. Koren et al. [1] presented a seminal approach that decomposes the user-item interaction matrix into latent factors representing hidden dimensions of preference. Their work, particularly in the context of the Netflix Prize competition, showed how factorization could significantly outperform simpler models in capturing user tastes and item characteristics. This approach enables scalable and accurate recommendations when sufficient data is available, but its reliance on dense interaction matrices limits its effectiveness in cold-start scenarios.

To compensate for data sparsity, active learning techniques have been introduced to prioritize the acquisition of the most informative feedback from users. Rubens et al. [2] conducted a comprehensive survey detailing the application of active learning in recommender systems. They categorize techniques into various types, including uncertainty-based strategies that select items with ambiguous predictions, and model-change-based methods that identify items likely to maximize learning if rated. Their work provided a framework for selecting feedback strategies tailored to a system's learning goals.

Laanen and Frasincar [3] built on this foundation by formalizing two specific strategies tailored for cold-start conditions—Y-change and Error-change. Y-change selects items based on their impact on output predictions, capturing model sensitivity to new information. Error-change, in contrast, evaluates potential reductions in prediction error. These strategies were further classified into conservative, moderate, and risky variants to accommodate different assumptions about user behavior. Their extensive experiments, conducted on implicit feedback data, established clear benchmarks for RMSE across varying item presentation levels. Their work is particularly relevant to this study as it provides the methodological groundwork for implementing and evaluating active learning in cold-start recommendation.

Their workflow, shown in Figure 1, provides a visual representation of how SVD factorization is applied to the interaction matrix, followed by regularization-based optimization of latent user and item vectors. Once these vectors are obtained, different item selection strategies including PopGini, Random Select, Y-change, and Error-change are evaluated to find the most suitable item(s) for feedback in the cold-start context.

Geurts and Frasincar [4] extended the active learning paradigm by introducing the PopGini strategy. This approach evaluates items based on a combination of popularity and Gini impurity, favoring items that are both frequently interacted with and yield diverse feedback distributions. Though not explicitly labeled as clustering, the method segments the item space according to feedback heterogeneity, implicitly capturing structural patterns in user-item interactions. Their evaluation demonstrated that PopGini not only improved performance but also maintained recommendation diversity—an essential aspect of user satisfaction in early interaction phases.

Explicit clustering of items has also been explored to ensure feedback is gathered across a representative cross-section of the item space. Rashid et al. [5] advocated for early-stage exposure to a variety of items, noting that personalized recommendations could emerge more rapidly when users rated a diverse set of representative items. They proposed using latent similarity metrics and item groupings to achieve this effect. In the context of this work, K-means clustering on item embeddings is applied to identify meaningful subgroups, improving both diversity and informativeness in active item selection.

Recent efforts have also investigated incorporating richer user context. Elourajini and Aïmeur [6] proposed a personality-aware framework for conversational recommendation systems. Their model leverages the Big Five personality traits to guide prompt-based recommendation dialogues, improving alignment with user preferences even in data-sparse conditions. Though their method requires psychological profiling and is NLP-centric, it represents a broader movement toward holistic user modeling.

Ji et al. [7] approached cold-start from a dynamic interaction perspective. Their multi-subsession conversational framework maintains session-level memory and progressively refines recommendations as conversations unfold. This architecture is designed for evolving user intent, dynamically asking questions to activate latent interests. While more complex in terms of system design, it offers valuable insights into incremental preference elicitation and can be viewed as an adaptive form of active learning.

The present study draws from these works and contributes a hybrid recommendation framework that combines matrix factorization, clustering, and adaptive active learning. The system integrates prediction sensitivity and error-based learning into a unified scoring model, optimized through Expectation-Maximization. Unlike prior works that treat strategies in isolation, this framework dynamically tunes strategy importance, offers explainable diversity through clustering, and demon-

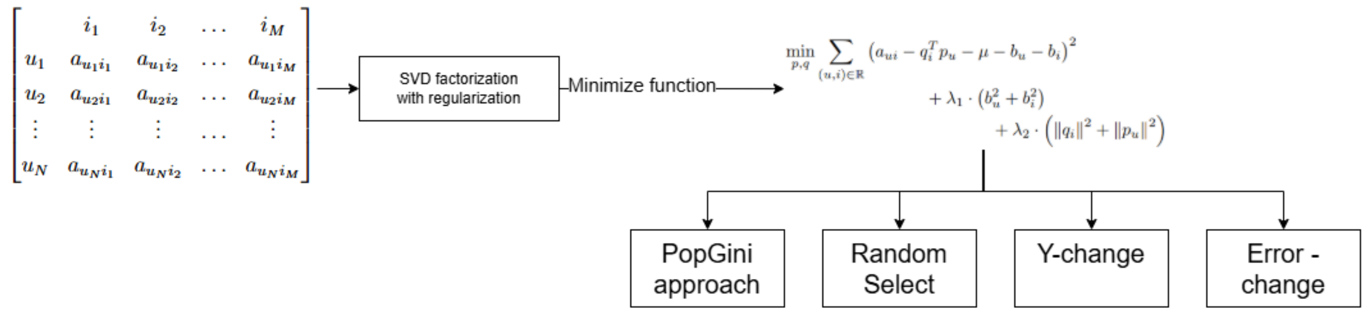

Fig. 1. Workflow diagram of the baseline method by Laanen and Frasincar [3], incorporating SVD with regularization and various active learning strategies.

strates superior RMSE performance in simulated cold-start settings.

III. Methodology

This section outlines the design and implementation of a framework developed to solve the cold-start problem in recommender systems. The methodology integrates collaborative filtering using matrix factorization, clustering through K-means to organize item space, and active learning to selectively acquire informative feedback from users. These techniques are combined with a hybrid scoring strategy, which is further optimized using the Expectation-Maximization (EM) algorithm. Each component of this framework plays a critical role in ensuring that the system can provide meaningful recommendations even when historical data is limited.

Figure 2 illustrates the overall architecture of the proposed hybrid framework. The pipeline begins with a user-item interaction matrix, which is decomposed via SVD with regularization to learn latent features. These item embeddings are then clustered using K-means to create diverse and representative item groups. A hybrid scoring function, combining Y-change and Error-change metrics, is then used to evaluate each candidate item. The balance between these metrics is controlled by the parameter α, which is optimized using the Expectation-Maximization (EM) algorithm. The process concludes with selecting items that minimize the Root Mean Squared Error (RMSE), ensuring more accurate cold-start recommendations.

A. Dataset and Feedback Encoding

The dataset used in this study was collected from an e-commerce platform [3], where user interactions with items were recorded over a period of time. Each interaction is represented in a binary format: a value of 1 denotes a positive interaction (such as a purchase or retained item), while a value of 0 represents a negative or neutral interaction (such as product return or no engagement).

For computational manageability, a subset of 100,000 user-item interactions was randomly sampled. This data forms the interaction matrix $R \in \mathbb{R}^{m\times n}$, where m is the number of users and n is the number of items.

B. Cold User Simulation

To mimic real-world scenarios, users with minimal interaction history were isolated from the training data. These users, known as cold users, had their known interactions withheld from the model during training. This approach simulates the cold-start condition where a recommender system encounters users with no prior behavior data, enabling evaluation of the model's ability to personalize recommendations based on limited feedback.

C. Matrix Factorization for Preference Learning

Matrix factorization is a powerful collaborative filtering technique used to uncover latent relationships between users and items. It works by decomposing the interaction matrix R into two lower-dimensional matrices:

$$R \approx PQ^T \tag{1}$$

where:

- $P \in \mathbb{R}^{m\times k}$ is the user latent feature matrix,
- $Q \in \mathbb{R}^{n\times k}$ is the item latent feature matrix,
- k is the number of latent dimensions.

Each user and item is represented in a k-dimensional latent space, and their interaction is modeled as the dot product of their respective vectors. The predicted interaction score between user u and item i is calculated as:

$$\hat{r}_{ui} = \mu + b_u + b_i + \mathbf{p}_u^T\mathbf{q}_i \tag{2}$$

where μ is the global average rating, b_u and b_i are user and item-specific biases, and $\mathbf{p}_u$, $\mathbf{q}_i$ are the latent vectors for user u and item i respectively.

D. Item Clustering Using K-Means

To ensure that the items selected for user feedback are diverse and representative, clustering is performed on the item embeddings obtained from matrix factorization. K-means clustering is used for this purpose. In this method, items are grouped into K clusters based on their similarity in the latent space. Each item vector $\mathbf{q}_i$ is assigned to the nearest cluster center μ_j:

$$\text{cluster}(i) = \arg\min_{j\in\{1,\ldots,K\}} \|\mathbf{q}_i - \mu_j\|^2 \tag{3}$$

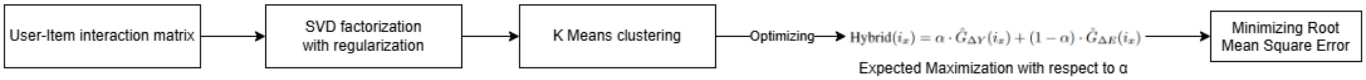

Fig. 2. Overview of the proposed hybrid framework: SVD is applied to the user-item interaction matrix followed by K-means clustering. A hybrid utility score is then computed and optimized using Expectation-Maximization to minimize prediction error.

This ensures that items within the same cluster share similar characteristics, and only one or a few representative items from each cluster are considered for querying, which improves the efficiency and informativeness of the system.

To determine the optimal number of clusters K, the *silhouette score* was used. The silhouette score is a metric that evaluates how well an object lies within its cluster. It is defined for each point as:

$$\text{Silhouette}(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}} \tag{4}$$

where $a(i)$ is the mean intra-cluster distance (i.e., the average distance between i and all other points in the same cluster), and $b(i)$ is the mean nearest-cluster distance (i.e., the average distance from i to all points in the closest neighboring cluster).

The silhouette score ranges from -1 to 1, with higher values indicating that an item is well matched to its own cluster and poorly matched to neighboring clusters. A score close to 1 indicates dense, well-separated clusters, while scores near 0 indicate overlapping clusters, and negative scores imply potential misclassification.

To optimize K, silhouette scores were computed for cluster values ranging from 2 to 10. The number of clusters that yielded the highest average silhouette score was selected as the optimal value. In this study, $K = 2$ was found to be the best choice, as it maximized the silhouette score, suggesting a meaningful partition of the item space. This process is illustrated in Figure 3, which shows the relationship between the silhouette score and the number of clusters.

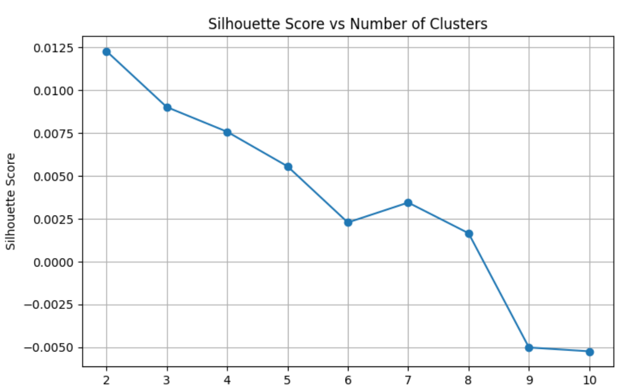

Fig. 3. Silhouette Score vs. Number of Clusters (K). The highest score was observed at $K = 2$.

By incorporating optimized clustering into the item selection pipeline, the system ensures that queried items are not only informative but also diverse, enhancing the system's ability to infer user preferences efficiently.

E. Active Learning Strategies

Active learning allows the system to selectively ask users about certain items to quickly learn their preferences. Two strategies are implemented:

1) Y-Change Strategy: This strategy estimates how much the system's predictions would change if the user were to provide feedback on a given item. The objective is to prioritize items that cause significant updates in predictions. The Y-change generalization error is computed as:

$$\hat{G}_{\Delta Y}(i_x) = -\sum_{y \in Y} \sum_{(u,i) \in A_{\text{test}}} \left(\hat{f}_T(u,i) - \hat{f}_{T \cup (i_x, u_0, y)}(u,i) \right)^2 \tag{5}$$

where:

- i_x is a candidate item selected for active feedback.
- $y \in Y$ represents possible feedback ratings (typically 0 or 1 in binary feedback).
- u_0 is the cold-start user.
- A_{test} is the test set of user-item interactions.
- $\hat{f}_T(u,i)$ is the predicted rating for user u and item i based on the original training set T.
- $\hat{f}_{T \cup (i_x, u_0, y)}(u,i)$ is the prediction after updating T with hypothetical feedback y on item i_x.

Here, $\hat{G}_{\Delta Y}(i_x)$ represents the estimated impact of adding a hypothetical rating for item i_x on the entire prediction space. A larger value of this metric (after negation) implies that the system's predictions are significantly affected by the new rating, indicating that this item is highly informative for learning about user preferences.

In simple terms, this equation is checking how much the predicted preferences for all users and items would shift if we assumed a new user (cold-start) had rated a specific item in a certain way. The higher the change, the more "informative" the item is, meaning it helps the system learn faster.

2) Error-Change Strategy: This strategy selects items that, if rated, would most reduce the overall prediction error. It is defined as:

$$\hat{G}_{\Delta E}(i_x) = \sum_{y \in Y} \sum_{(u,i) \in A_{\text{test}}} \left(r_{ui} - \hat{f}_{T \cup (i_x, u_0, y)}(u,i) \right)^2 \tag{6}$$

where:

- r_{ui} is the true feedback for user u and item i.
- Other variables are as previously defined.

In this equation, $\hat{G}_{\Delta E}(i_x)$ estimates how much total prediction error across the system would be reduced by asking the cold user to rate item i_x. Items with lower error scores (post-feedback) are deemed more valuable because they help the system become more accurate.

This formula measures how much better the system's predictions would become if the cold user rated a certain item. The goal is to find items that would help reduce future mistakes in recommendations.

Both strategies are evaluated under three behavioral assumptions:

- **Conservative:** Assumes minimal impact from user feedback (e.g., low variance or rating close to neutral).
- **Moderate:** Takes the average of all possible feedback outcomes (e.g., uniform expectation).
- **Risky:** Assumes the most informative (and possibly extreme) user response (e.g., highest variance).

F. Hybrid Scoring Function

To capture the benefits of both Y-change and Error-change, a hybrid score is computed for each candidate item:

$$\text{Hybrid}(i_x) = \alpha \cdot \hat{G}_{\Delta Y}(i_x) + (1-\alpha) \cdot \hat{G}_{\Delta E}(i_x) \tag{7}$$

This score is like a weighted average: the system decides how much importance to give to prediction change (Y-change) versus error reduction (Error-change) for each item. A higher value of $\hat{G}_{\Delta Y}(i_x)$ indicates an item is better for adjusting the predictions, while a higher $\hat{G}_{\Delta E}(i_x)$ indicates an item is better for reducing future error. The hybrid score combines these in a balanced manner using the weight α.

The parameter $\alpha \in [0, 1]$ determines the weight given to each strategy. A higher α favors Y-change (prediction shift), while a lower value emphasizes Error-change (error reduction).

G. Optimization of α Using EM Algorithm

Rather than fixing α arbitrarily, it is optimized using the Expectation-Maximization (EM) algorithm. This allows the model to adaptively determine the best weighting based on data:

E-step: Estimate the likelihood of each candidate item given current α.

$$\mathcal{L}(\alpha) \propto \prod_j \exp(-\text{Hybrid}_\alpha(i_x^{(j)})) \tag{8}$$

This step evaluates how well each possible value of α explains the current item scores.

M-step: Update α to maximize the expected log-likelihood:

$$\alpha^{(t+1)} = \arg\min_\alpha \sum_j \text{Hybrid}_\alpha(i_x^{(j)}) \tag{9}$$

This step selects the α that gives the lowest average hybrid score across candidate items—essentially tuning the model to balance learning from prediction changes and error reduction.

This iterative process continues until convergence is reached, typically within a small number of iterations, yielding a value of α that best balances the two strategies for a given user's data.

H. Evaluation Protocol

Cold users are shown a small set of $N \in \{10, 25, 50, 100\}$ items selected by each strategy. Simulated feedback is added, and the recommender is retrained. The accuracy of predictions on the remaining items is measured using Root Mean Squared Error (RMSE):

$$\text{RMSE} = \sqrt{\frac{1}{|\mathcal{T}|} \sum_{(u,i)\in\mathcal{T}} (\hat{r}_{ui} - r_{ui})^2} \tag{10}$$

where $\mathcal{T}$ is the set of user-item pairs used for testing.

This methodology provides a structured and adaptive framework to improve recommendation accuracy for new users using minimal feedback.

IV. Results and Analysis

This section presents the experimental results obtained from evaluating the proposed hybrid active learning framework in a cold-start scenario. The analysis includes metrics on prediction accuracy, the behavior of the hybrid scoring parameter α, and clustering performance across item embeddings. Visualizations such as PCA projections, silhouette scores, and alpha distributions provide additional insight into the underlying system behavior.

A. Evaluation of Cold-Start Strategies

The recommender system was evaluated using Root Mean Squared Error (RMSE) across different active learning strategies: *risky*, *moderate*, and *conservative*. For each strategy, cold users were presented with a fixed number of candidate items ($N = 10, 25, 50, 100$) for feedback collection. The RMSE values were computed after incorporating the feedback and re-training the model.

The results demonstrate a consistent trend: the **conservative strategy** achieved the lowest RMSE across all query sizes. For instance, when only 10 items were shown, the conservative approach yielded an RMSE of 0.059, significantly outperforming moderate (0.141) and risky (0.520) variants. This trend persisted even as the number of queried items increased, indicating that conservative selection offers more stable and reliable user profiling under uncertainty.

These observations reinforce the idea that assuming minimal user impact (conservative variant) leads to safer, more generalized learning when data is sparse. Conversely, the risky strategy—which assumes highly informative feedback—suffers from increased variance and lower generalization.

B. Comparison with Baseline Work

The proposed framework was benchmarked against the baseline method introduced by Laanen and Frasincar [3], which employed SVD-based matrix factorization enhanced with active learning through Y-change and Error-change strategies. Their experimental setup evaluated the conservative, moderate, and risky variants of both strategies over different numbers of items presented to cold users.

According to their reported results, the **best-performing strategy in their work** was the *moderate Error-change variant*, achieving a minimum RMSE of 0.352 when 10 items were shown. In contrast, the hybrid framework developed in this study achieved a **significantly lower RMSE of 0.059** under the conservative strategy with the same number of items. This represents a reduction in prediction error by over 83%, highlighting the effectiveness of incorporating clustering and hybrid scoring mechanisms into the feedback acquisition process.

Additionally, Laanen and Frasincar reported increasing RMSE values as more items were presented (e.g., RMSE of 0.422 at 100 items using moderate Error-change). In comparison, the proposed method maintained lower RMSE values across all feedback sizes, demonstrating greater scalability and robustness.

This substantial improvement can be attributed to several enhancements in the proposed system:

- The use of **K-means clustering** to organize item space ensured diverse and representative item exposure.
- A **hybrid utility function** enabled a balanced integration of prediction shift and error reduction.
- **Adaptive tuning of the α parameter** via the EM algorithm allowed the model to personalize its learning strategy for individual users.

These improvements not only lowered prediction error but also enhanced the system's learning efficiency, particularly in early-stage user modeling.

A detailed numerical comparison of RMSE values between the baseline and proposed method across multiple feedback sizes and strategy variants is presented in Table I.

TABLE I
RMSE COMPARISON BETWEEN BASELINE AND PROPOSED METHOD

Strategy Variant	10	25	50	100	Mean
Risky error-change	0.426	0.445	0.453	0.430	0.438
Risky Y-change	0.450	0.435	0.451	0.450	0.447
Risky hybrid	0.520	0.401	0.331	0.278	0.383
Moderate error-change	0.352	0.385	0.401	0.422	0.390
Moderate Y-change	0.402	0.430	0.441	0.441	0.428
Moderate hybrid	0.141	0.248	0.150	0.191	0.183
Conservative error-change	0.401	0.422	0.439	0.441	0.426
Conservative Y-change	0.417	0.439	0.424	0.444	0.431
Conservative (Proposed)	0.059	0.081	0.150	0.191	0.120

C. Analysis of Item Embedding Clusters

K-means clustering was applied to the item embeddings derived from matrix factorization. The item vectors were projected to two dimensions using Principal Component Analysis (PCA) for visualization (Figure 4). Items were colored by their assigned cluster, showing relatively well-separated regions with two dominant clusters.

This clustering provided a meaningful partition of the item space and helped ensure that selected items covered diverse latent item features, thereby improving the quality of elicited user feedback.

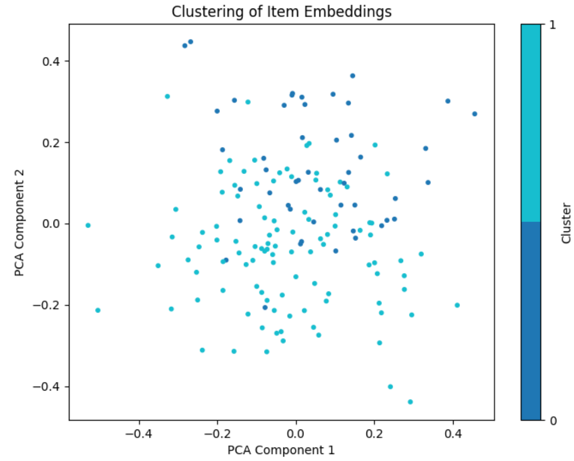

Fig. 4. Clustering of Item Embeddings visualized using PCA. Two distinct clusters were identified using K-means.

D. Optimized Hybrid Weights (α)

To combine the benefits of Y-change and Error-change strategies, a hybrid scoring function was used with a weighting parameter α, optimized via the Expectation-Maximization (EM) algorithm. The optimized α values varied across strategies and users.

Figure 5 shows the distribution of α values for each strategy. The conservative strategy exhibited the highest spread and highest median α, indicating a stronger reliance on Y-change. In contrast, the moderate and risky strategies favored Error-change, as reflected in their lower α values.

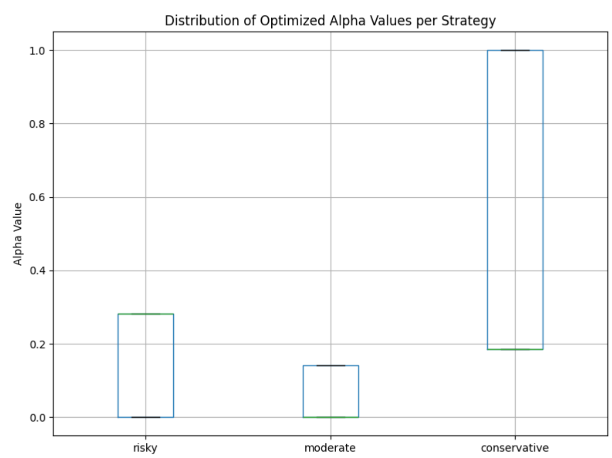

Fig. 5. Distribution of Optimized α Values per Strategy. Conservative variants favor Y-change, while risky and moderate lean towards Error-change.

Figure 6 further details the variation in optimized α values across individual cold users. The conservative strategy again displayed high variability, which suggests personalized utility in adjusting the hybrid score to match user behavior.

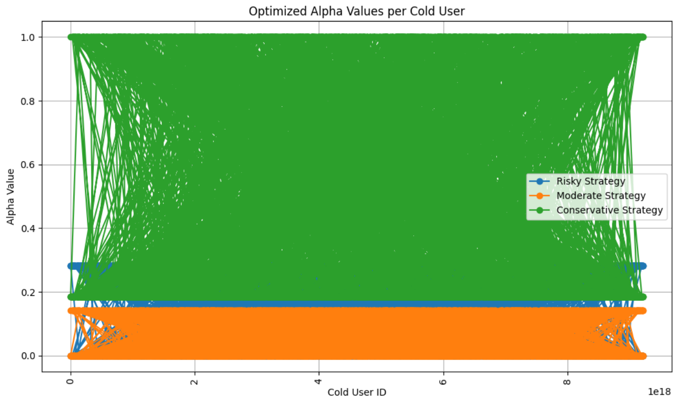

Fig. 6. Optimized α Values per Cold User. Conservative strategy shows wider dispersion, allowing greater adaptation.

These results indicate that tuning α per user and per strategy provides a significant advantage, especially when user behavior diverges from global assumptions. EM-based optimization allows the model to make these adjustments dynamically.

V. Discussion and Conclusion

The findings presented in the results highlight the strengths of the proposed hybrid active learning framework, particularly in addressing the cold-start challenge. Several key takeaways emerge from the experiments that provide deeper insights into the system's behavior and performance, which are discussed below.

A. Impact of Strategy Selection

The analysis of RMSE values across the three active learning strategies underscores the superiority of the conservative approach. This strategy, which operates under the assumption that user feedback may be minimally informative, produced the most stable and accurate results. Such performance suggests that a cautious estimation of user behavior mitigates overfitting and enhances generalization—critical factors when working with limited data.

In contrast, the risky strategy demonstrated less consistent performance. By assuming that users would provide highly informative ratings, this method likely introduced bias into the model updates, especially when such assumptions did not align with the actual feedback. The moderate strategy offered a balanced alternative, but still trailed behind the conservative variant in terms of RMSE across most conditions.

These findings reinforce that in uncertain and data-scarce settings, conservative estimation often yields better outcomes than optimistic ones. This aligns with established principles in active learning and decision theory, where caution tends to be more robust to noise and unanticipated user behavior.

B. Utility of Clustering in Candidate Item Selection

The integration of K-means clustering into the candidate selection process significantly contributed to the robustness of the active learning framework. By grouping items with similar latent characteristics and selecting representative items from each cluster, the model was able to expose users to a more diverse range of item types.

This diversity is crucial in the cold-start phase, where the system lacks a detailed understanding of user preferences. The clustering approach ensured that feedback collected early in the process spanned different regions of the latent item space, allowing the recommender to learn more about the user with fewer interactions.

The silhouette score analysis confirmed that a two-cluster solution was optimal for the dataset used, providing empirical support for the cluster configuration. The well-separated clusters also implied that the item space captured meaningful latent dimensions that the K-means algorithm successfully exploited.

C. Role of Hybrid Scoring and Adaptive Alpha Tuning

The hybrid scoring mechanism, which combines the benefits of Y-change and Error-change strategies, was further enhanced by dynamically optimizing the weighting parameter α using the EM algorithm. This adaptability proved essential in accommodating user-specific differences in feedback behavior.

As shown in the results, the optimized α values varied significantly across users and strategies. The conservative strategy, in particular, exhibited a wide dispersion in α, indicating the flexibility to tune the model's reliance on either prediction shift or error minimization depending on individual responses.

Such personalization at the scoring level is an important step toward building user-aware recommender systems. It allows the framework to maintain high accuracy without requiring explicit user profiles or side information, relying instead on data-driven adaptation.

D. Scalability and Real-World Applicability

The simplicity and modularity of the proposed framework make it suitable for integration into real-world recommendation engines. Matrix factorization, K-means clustering, and EM optimization are computationally efficient and scalable techniques, which ensures that the system can be deployed at scale without significant overhead.

Furthermore, the ability to operate under cold-start conditions without access to explicit metadata or user demographics broadens the applicability of the framework, making it valuable in scenarios where user data privacy is a concern or where such information is not readily available.

E. Limitations and Future Directions

While the proposed system performs well, several limitations warrant future exploration. First, the clustering was performed using K-means with a fixed number of clusters determined by silhouette score. More advanced or adaptive clustering methods, such as hierarchical clustering or DBSCAN, could be explored to capture more nuanced item structures.

Second, while the EM optimization of α proved effective, other meta-learning techniques or reinforcement learning-based controllers could be tested to further enhance scoring adaptation. Additionally, future work could investigate extending the hybrid scoring framework to multi-round feedback settings, where the system continuously refines its understanding of the user over multiple sessions.

Overall, the discussion of results affirms that the proposed hybrid active learning framework provides a robust, efficient, and adaptive solution to the cold-start problem in recommender systems.

References

[1] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Computer*, vol. 42, no. 8, pp. 30–37, 2009.

[2] N. Rubens, M. Elahi, M. Sugiyama, and D. Kaplan, "Active learning in recommender systems," in *Recommender Systems Handbook*, Springer, 2015, pp. 809–846.

[3] R. Laanen and F. Frasincar, "Alleviating the cold-start problem in recommender systems using error-based learning," in *Proc. IEEE/WIC Int. Conf. Web Intelligence*, pp. 310–315, 2024.

[4] T. Geurts and F. Frasincar, "Addressing the cold user problem for model-based recommender systems," in *Proc. ACM Web Intelligence*, 2017.

[5] A. M. Rashid, I. Albert, D. Cosley, S. K. Lam, S. M. McNee, J. A. Konstan, and J. Riedl, "Getting to know you: Learning new user preferences in recommender systems," in *Proc. Int. Conf. Intelligent User Interfaces*, 2002.

[6] F. Elourajini and E. Aïmeur, "PPPG-DialoGPT: A personality-aware framework for conversational recommendation systems," in *Proc. IEEE/WIC Int. Conf. Web Intelligence*, 2023.

[7] Y. Ji, Q. Shen, S. Zhu, H. Yu, Y. Zhang, C. Cui, and Z. Wei, "Towards multi-subsession conversational recommendation," in *Proc. PAKDD*, 2024.