SPADE: Escaping the Popularity-Similarity Frontier to Measure Serendipitous Recommendations

Tobias Vente[1], Maarten Peirsman[1], Noah Daniëls[1], Hannu Toivonen[2] and Bart Goethals[1,3]

[1]ADREM Data Lab, University of Antwerp, Middelheimlaan 1, 2020 Antwerpen, Belgium
[2]Department of Computer Science, PO Box 68, FI-00014 University of Helsinki, Finland
[3]Froomle, Post X, Borsbeeksebrug 34, 2600 Antwerpen, Belgium

Abstract

Recommender systems engineer serendipity to foster active exploration and break predictable consumption cycles. The problem with existing offline beyond-accuracy metrics is that they often either isolate historical similarity or global popularity. We aim to design an evaluation metric that examines similarity, popularity, and actual user relevance. To achieve this, we introduce SPADE (Serendipitous Pareto Distance Evaluation). SPADE maps all items into a two-dimensional space to directly calculate a user-specific Pareto frontier of maximally popular and historically similar items. The final serendipity score is then computed by averaging the minimum Euclidean distance from this boundary strictly for the correctly recommended test-set items. Evaluating SPADE across five datasets and five baseline algorithms confirms its effectiveness; our results show that the metric successfully prevents algorithms from exploiting beyond-accuracy measures with irrelevant or non-personalized recommendations, reliably isolating serendipitous discoveries.

1. Introduction

Recommender systems excel at predicting user preferences, yet they often struggle to deliver what users never knew they needed [1, 2]. When systems strictly optimize for predictive accuracy, they risk trapping users in isolated information silos and continuously narrowing their exposure to diverse content [3]. To overcome this filter bubble effect and ensure long-term user retention, researchers engineer serendipity into these systems [4]. Serendipity acts as a counterweight, fostering active exploration and breaking predictable consumption cycles [1].

While dictionary definitions of serendipity emphasize pure chance and luck, in the context of recommender systems, it is treated instead as an explicit engineering objective. Serendipity is defined as the intersection of surprise, relevance, and novelty in recommendations [5, 6, 7]. Introducing serendipity does not mean serving random items; rather, it involves delivering suggestions that appear as fortunate, random discoveries to the user, while in fact being purposefully selected by the algorithm [5, 6, 7]. For example, recommending the next obvious blockbuster or a direct sequel almost guarantees high acceptance, but it fails to deliver a surprising discovery [2]. In contrast, recommending an item from an unexpected yet relevant genre achieves serendipity [8].

The degree of serendipity in a recommendation relies on two distinct dimensions: its similarity to the target user's historical interactions and its global popularity across the system [6, 9]. While suggesting highly similar or globally dominant items is a safe strategy to ensure baseline relevance, it fundamentally compromises both unexpectedness and novelty [6, 9]. A recommendation that too closely mirrors a user's past interactions fails to generate surprise, just as a universally popular item is highly unlikely to provide a novel discovery. Consequently, achieving serendipity requires navigating

FRAME'26: Methodology First - Rethinking Research Assessment in RecSys Workshop, September 28, 2026, Minneapolis, Minnesota, USA

tobias.vente@uantwerpen.be (T. Vente); maarten.peirsman@uantwerpen.be (M. Peirsman); noah.daniels@uantwerpen.be (N. Daniëls); hannu.toivonen@helsinki.fi (H. Toivonen); bart.goethals@uantwerpen.be (B. Goethals)
https://tobiasvente.github.io/ (T. Vente); http://www.cs.helsinki.fi/hannu.toivonen/ (H. Toivonen); https://www.bartgoethals.com/ (B. Goethals)
0009-0003-8881-2379 (T. Vente); 0009-0004-1394-354X (M. Peirsman); 0009-0001-7778-4769 (N. Daniëls); 0000-0003-1339-8022 (H. Toivonen); 0000-0001-9327-9554 (B. Goethals)

© 2026 Copyright © 2026 for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

CEUR Workshop Proceedings ceur-ws.org ISSN 1613-0073
published 2026-11-26

CEUR-WS.org/Vol-4341/paper7.pdf

away from these high-probability zones to find items that are relevant, yet neither popular nor similar to a user's past interactions.

Unfortunately, established serendipity metrics struggle to capture this dynamic due to their narrow focus. They tend to evaluate serendipity by isolating just one dimension of the recommendation space. For example, some approaches measure unexpectedness purely as the content-based distance between recommendations and a user's history [10, 11, 12]. Other metrics focus entirely on global patterns. These include evaluating serendipity as a deviation from primitive popularity baselines [13, 1], relying on item co-occurrence probabilities [14, 15], or measuring the time saved in discovering an item [16].

While existing serendipity metrics capture distinct facets of serendipity, they share a flaw: they do not assess both similarity and popularity simultaneously. By ignoring either similarity or popularity, isolated metrics often reward mainstream items simply because they are new to the user, or reward highly predictable items just because they are globally rare, or inadvertently reward obscure discoveries that hold no actual relevance to the user. Failing to penalize these false positives can inflate the measured serendipity, exposing the need for a metric that strictly optimizes for both dimensions.

To address this gap, we introduce SPADE (Serendipitous Pareto Distance Evaluation), a novel offline beyond-accuracy metric that quantifies serendipity by jointly evaluating historical similarity and global popularity. Within this two-dimensional space, we construct a user-specific Pareto frontier of items that maximize both global reach and alignment with a user's history. SPADE measures serendipity as the Euclidean distance from an algorithm's correctly predicted items to the closest point on the Pareto front. By defining SPADE and evaluating it against established beyond accuracy metrics, we investigate whether this structural approach can successfully measure serendipity. Ultimately, SPADE provides researchers with a standard, quantifiable method to determine whether an algorithm is merely retreating to predictable and popular content or if it is actively delivering serendipitous discoveries.

To validate SPADE, we execute three research tasks: formalization, offline evaluation, and comparative analysis. First, we define SPADE's mathematical framework, detailing the construction of the Pareto frontier and the calculation of distances. Then, by comparing SPADE against commonly used beyond-accuracy metrics (Novelty, Co-Occurrence, and Primitivity) across five baseline algorithms on five widely used datasets, we demonstrate how it isolates serendipitous items that traditional metrics misclassify.

The code used to execute the experiments is publicly available in our GitHub[1] repository and further contains documentation to ensure the reproducibility of our experiments

2. Related Work

While the importance of serendipity in recommender systems is widely recognized, existing evaluation metrics largely approach the concept through isolated dimensions. A review of the literature reveals three primary methodologies for quantifying serendipity and unexpectedness offline. However, because these approaches rely on a narrow focus (prioritizing either user history, primitive baselines, or temporal distance), they struggle to capture the necessary dual dynamic of historical similarity and global popularity, leaving them vulnerable to false positives.

The most frequent approach to measuring serendipity calculates the distance between a recommended item and the user's historical profile, utilizing either content features or collaborative filtering (CF) metrics. Initially outlined by Shani and Gunawardana [10], content-based distance measurements have been widely adapted to evaluate how far a new recommendation deviates from a user's known preferences [11, 17, 12]. Variations of this metric, which calculate the difference between recommendations and user profiles through slightly different formulas, have been used to evaluate user exploration [18] and multi-objective diversity [19].

Similarly, other researchers measure unexpectedness by calculating CF-based distances, frequently employing Point-wise Mutual Information (PMI) or Normalized PMI (NPMI) to evaluate the statistical co-occurrence between a candidate item and the specific items in a user's historical profile [14, 15, 20]. While effective at measuring personalization, these profile-centric metrics are entirely blind to global

[1] https://github.com/Maarten11/SPADE

item popularity. By focusing solely on the user's history, they risk rewarding highly popular items simply because they are absent from a specific user's profile, falsely categorizing a highly predictable global trend as a serendipitous discovery.

A second prominent approach operationalizes serendipity by comparing a proposed algorithm's outputs against those of a "primitive" recommender system. Originally introduced by Murakami et al. [13] and famously formalized by Ge et al. [1], this metric conceptualizes serendipity as the proportion of relevant items that are unexpected. Unexpectedness is defined by the items the primitive baseline fails to recommend. This method has been widely adopted across various domains, including news recommendations [21], serendipity-oriented greedy algorithms [22], fuzzy inference systems [23], research paper recommendations [24, 25], offline data processing [26], and deep neural networks [27, 28]. The fundamental flaw of this approach is its heavy reliance on the chosen baseline. Because the primitive system can be based on either global popularity or historical similarity (recommending the same content), the resulting serendipity score is ambiguous. It cannot guarantee that an item is both a long-tail discovery and a taste-broadening deviation.

A more specialized metric defines serendipity through the lens of time. Introduced by Kawamae [29], this approach measures the difference between the time an algorithm recommends an item and the time it would theoretically take a user to find that item on their own without the system's help. An item with a large time difference is considered highly serendipitous because it implies the user would not have naturally discovered it in the near future. This metric has been primarily used to evaluate real-time recommendations [16, 12]. While conceptually interesting, this temporal approach requires extensive, time-stamped historical logs of organic user discovery, which are rarely available. Furthermore, it measures the symptom of serendipity (delayed discovery) rather than the underlying structural causes—the item's relationship to the user's taste and the global market.

While our work introduces the Pareto front as an evaluation tool, the concept itself is well-established in recommender systems literature, primarily within the domain of multi-objective optimization [30, 31]. Because recommender systems often face competing goals—such as maximizing accuracy while maintaining catalog coverage, diversity, or fairness—researchers frequently employ Pareto frontiers to find optimal trade-offs during model training [32, 31]. For instance, algorithms are often designed to push recommendations toward a Pareto-optimal boundary where accuracy cannot be further improved without sacrificing diversity [33, 34].

Critically, the existing literature uses Pareto frontiers almost exclusively as an optimization objective for generating recommendations rather than as a post-hoc evaluation metric.

Ultimately, the current literature lacks a unified metric that evaluates serendipity as a function of both user-level similarity and global popularity. Because existing metrics ignore this crucial trade-off, they lack a reliable mechanism to penalize the false positives that occur at the extremes of either dimension.

3. SPADE (Serendipitous Pareto Distance Evaluation)

To evaluate serendipity, every candidate item i for a user profile U is mapped into a two-dimensional space defined by its global popularity and its similarity to the user's profile. We calculate a user-specific Pareto front of items that maximize both of these dimensions. Each item on this front is either the most popular item among items that are at least as similar or the most similar item among items that are at least as popular. We posit that these Pareto-optimal items are the least serendipitous because they dominate either popularity or similarity. Conversely, items farther from the Pareto front are considered more serendipitous, as they are simultaneously less popular and less similar to items on the front (section 5.1). To compute the final SPADE score, we calculate the average distance from the Pareto front only for the correctly recommended items (i.e., those in the user's test set).

3.1. Defining the Dimensions

For this Pareto formulation to be well-posed, the two dimensions must represent a genuine trade-off. It is therefore critical to choose a similarity measure that is not positively correlated with popularity.

If a positively correlated measure such as cosine similarity is used [35], the most popular items will inherently also score as the most similar. Geometrically, this causes the candidate items to cluster along a single diagonal, and the Pareto front collapses into a single point (or a very narrow band) in the extreme upper-right of the space. When the front collapses in this way, there is no longer a boundary of trade-offs, and measuring distance to the front becomes meaningless. We therefore require opposing measures to scatter items across the space, forcing a wide, distinct curve that represents the maximum possible trade-off between popularity and similarity.

Popularity (x-axis): We define the popularity of a candidate item i by its interaction frequency, normalized to a range of $[0, 1]$. During training, the occurrence n_i of each item is counted and divided by the maximum item count across the entire item set I:

$$pop(i) = \frac{n_i}{\max_{k \in I} n_k}$$

Similarity (y-axis): To satisfy the independent requirement, we utilize *unnormalized* pointwise mutual information (PMI). The PMI of two items is calculated based on the fraction of users who interacted with either and both items as follows:

$$\text{PMI}(i, j) = \log \frac{\frac{n_{ij}}{n}}{\frac{n_i}{n}\frac{n_j}{n}} = \log \frac{n_{ij}}{\frac{1}{n} n_i n_j}$$

where n_i and n_j indicate the number of users who have interacted with items i and j respectively, n_{ij} represents the number of users who interacted with both items, and n indicates the total number of users. It is well known that PMI is negatively correlated with popularity [36], making it ideal for our use case. We intentionally avoid Normalized PMI [14, 15], as the normalization process reintroduces positive correlation with popularity.

To counteract PMI's sensitivity to noise for rare items, we apply empirical Bayes smoothing [37]. This dampens the effect of low counts and shifts PMI towards 0 when evidence for the item pair is small. We also clip the similarities to be strictly non-negative, as negative similarities predominantly represent noise. The formula for our smoothed Positive PMI (PPMI) is:

$$\text{PPMI}_\alpha(i, j) = \max \left\{ 0, \log \frac{n_{ij} + \alpha}{\frac{1}{n} n_i n_j + \alpha} \right\}$$

We fix the smoothing strength at $\alpha = 1$ in our experiments. To compute the overall similarity between a candidate item i and the user profile U, we aggregate the pairwise similarities by taking the sum over all items j in the user's history:

$$sim(i, U) = \sum_{j \in U} \text{PPMI}_\alpha(i, j)$$

3.2. Space Normalization and Distance Calculation

Because popularity and profile similarity naturally operate on different scales, calculating a distance metric would result in the larger-scaled axis dominating the space. To ensure both dimensions contribute equally, we apply Min-Max scaling to the profile similarities for each user, projecting the similarity axis into the same $[0, 1]$ range as the popularity axis. We denote this scaled similarity as $\widetilde{sim}(i, U)$.

With both axes bounded in $[0, 1]$, we formally define the user-specific Pareto front $\mathcal{P}_U$. An item p belongs to $\mathcal{P}_U$ if there exists no other item k such that both $pop(k) > pop(p)$ and $\widetilde{sim}(k, U) > \widetilde{sim}(p, U)$ hold true.

Finally, the SPADE score of a candidate item i is calculated as the minimum Euclidean distance i to the Pareto front $\mathcal{P}_U$:

$$\text{SPADE}(i, U) = \min_{p \in \mathcal{P}_U} \sqrt{(pop(i) - pop(p))^2 + (\widetilde{sim}(i, U) - \widetilde{sim}(p, U))^2}$$

A higher SPADE score indicates that an item is located further away from the Pareto-optimal boundaries of popularity and similarity, effectively quantifying its serendipitous value.

The score for a user's recommendation R_U is calculated as the average SPADE score over all correctly recommended items (i.e., the items in user U's test set). To calculate the SPADE score for an algorithm, we average the scores over all the users in the test set.

4. Experimental Setup

Our experiments demonstrate the effectiveness of SPADE in evaluating recommendation serendipity relative to popular serendipity metrics. We evaluate SPADE using baseline recommendation algorithms and common dataset configurations reported in the recommender systems literature, ensuring that SPADE's performance can be applied to general recommender system research. Our analysis focuses on the top-N recommendation task, specifically examining how accurately the metric isolates genuinely unexpected items from easily predictable false positives based entirely on offline data.

Datasets: We evaluate SPADE across five datasets from distinct domains: CiteULike [38], MIND [39], MovieLens-1M [40], MovieLens-20M [40], and Netflix[2]. Table 1 shows statistical information on the datasets.

Dataset	# Interactions	# Users	# Items	Avg.#Int. per user	Avg.#Int. per item	Sparsity
CiteULike	200,251	5,550	15,439	36.08	12.97	99.77%
MIND*	1,086,939	10,000	33,220	108.69	32.72	99.67%
ML-1M	574,385	6,038	3,125	95.13	183.80	96.96%
ML-20M*	730,034	10,000	11,361	73.00	64.26	99.36%
Netflix*	1,234,880	10,000	15,093	123.49	81.82	99.18%

Table 1
Statistical information on the datasets used in our experiments. Datasets marked with (*) have been subsampled to 10,000 users to keep within hardware limitations.

Algorithms: We compute SPADE against recommendations predicted by five baseline algorithms: EASE, SLIM, Item-based k-Nearest Neighbors (ItemKNN), Popularity, and Random. We use the algorithm implementations from the Recpack library [41] and perform a grid search to tune the hyperparameters[3].

Preprocessing Following common practice, we perform 5-core filtering across all datasets, pruning the data to ensure that each item contains at least 5 interactions [42, 43]. For the most intensive data sets, MIND, MovieLens-20M and Netflix, we subsample the user populations due to hardware memory limitations, capping them at 10,000 users.

Beyond-Accuracy Metrics: To evaluate the effectiveness of SPADE against existing standards, our experimental setup uses three established baseline metrics that align directly with the methodologies discussed in section 2. First, to represent the profile-centric evaluation approaches, we measure **Co-Occurrence**. This metric operationalizes collaborative filtering (CF) distance between a candidate item and a user's historical profile, using Point-wise Mutual Information (PMI) to assess statistical unexpectedness. Second, to capture the comparative baseline methodology, we evaluate **Primitivity**. Following the framework established by Ge et al., this metric calculates serendipity as the proportion of recommendations that deviate from the outputs of a "primitive" system (which, for our experiments, is defined as a standard popularity-based recommender). Finally, we include **Novelty**, calculated as the inverse global popularity of the recommended items. Because profile-distance metrics (like Co-Occurrence) are often blind to global trends, Novelty serves as the traditional proxy for uncovering long-tail items, allowing us to independently isolate the global popularity dimension that SPADE structurally integrates.

[2] https://www.kaggle.com/datasets/netflix-inc/netflix-prize-data
[3] More details, like the search spaces, can be found in our GitHub repository: https://github.com/Maarten11/spade

Pipeline: For our evaluation, we rely on the strong generalization data-split scenario[4] utilizing an 80/20 train-test split, whereas 20% of the train set is used as validation data. We evaluate both standard recommendation performance and all included metrics at cut-offs $K \in \{1, 2, 3, 5, 10, 20, 30\}$.

Hardware: All experiments and evaluations were executed on an Apple Silicon M5 Pro with 64 GB of RAM.

5. Results & Discussion

In this section, we empirically evaluate the efficacy of the SPADE metric at both the user and dataset levels. We first provide a qualitative validation of its underlying mechanics through specific user case studies, followed by a comprehensive cross-domain quantitative analysis comparing SPADE against popular beyond-accuracy metrics.

5.1. Qualitative Validation: The Mechanics of SPADE

To illustrate the functionality and practical value of the SPADE metric, we examine the recommendation space for two distinct users who show serendipitous item interactions (User 1874 and User 2186). For both users, we visualize the item space by mapping normalized similarity to the user's historical profile on the y-axis and global item popularity on the x-axis. In these visualizations, candidate items are represented by blue dots, predictions generated by the EASE baseline are marked in green, and ground truth interactions from the test set are denoted by red crosses. The Pareto front—representing the optimal trade-off between similarity and popularity—is delineated by a solid line.

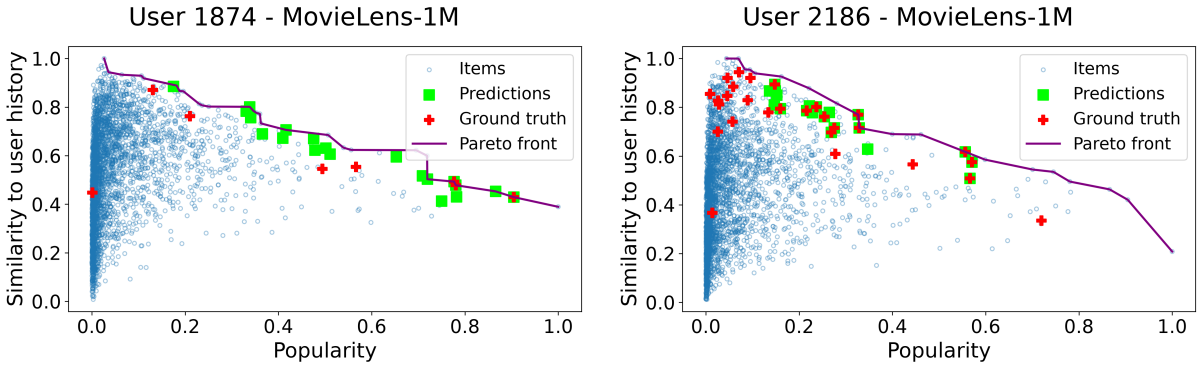

Figure 1: The y-axis denotes normalized item similarity to the user's historical profile, and the x-axis denotes global item popularity. Candidate items are represented by blue dots, EASE baseline predictions in green, and ground truth interactions (test set) by red crosses. The solid line indicates the Pareto front maximizing similarity and popularity. More serendipitous items are in the bottom-left quadrant. These relevant but unexpected items lie furthest from the Pareto front and therefore receive the highest serendipity scores under the SPADE metric.

In both cases, we observe a ground truth interaction that deviates significantly from the user's typical behavior, falling far outside the high-similarity, high-popularity cluster near the Pareto front.

Case Study 1 - User 1874: For User 1874, one test-set item stands out in the bottom-left quadrant of the plot in section 5.1: the 1997 Sci-Fi movie Retroactive. This user's historical interactions are dominated by Drama, Crime, and Thriller films, with Sci-Fi movies accounting for only 9% of their viewing history. Correspondingly, the item exhibits a low similarity score of 0.46 and an exceptionally low popularity score of 0.005. Despite being neither popular globally nor aligned with the user's established preferences, the user still interacted with it, proving its underlying relevance. Because this item lies the furthest distance from the Pareto front, SPADE identifies it as serendipitous and

[4] https://recpack.froomle.ai/generated/recpack.scenarios.StrongGeneralization.html#recpack.scenarios.StrongGeneralization

assigns it the highest serendipity score. This demonstrates SPADE's ability to reward the successful recommendation of deeply serendipitous, relevant items.

Case Study 2 - User 2186: User 2186 provides a contrasting example of serendipity that is even more extreme but equally valid. The standout ground truth item here is the 2000 crime movie The Way of the Gun. While this user primarily consumes Action and Adventure, Crime movies constitute 11.6 % of their interaction history. This item yields a similarity score of 0.38 and a popularity score of 0.05. If a recommender system were to successfully predict it, SPADE would award it the highest serendipity score for this user, acknowledging that the recommendation pushes the boundaries of their typical consumption habits while remaining relevant.

Ultimately, these examples validate the core premise of SPADE: by measuring distance from the Pareto front, the metric successfully identifies and rewards items that break away from obvious popularity and historical-similarity biases, capturing serendipity.

5.2. Cross-Domain Empirical Evaluation

Having established the foundational mechanics of the SPADE metric at the user level, we extend our evaluation to a cross-domain quantitative analysis to assess its robustness compared to traditional beyond-accuracy metrics.

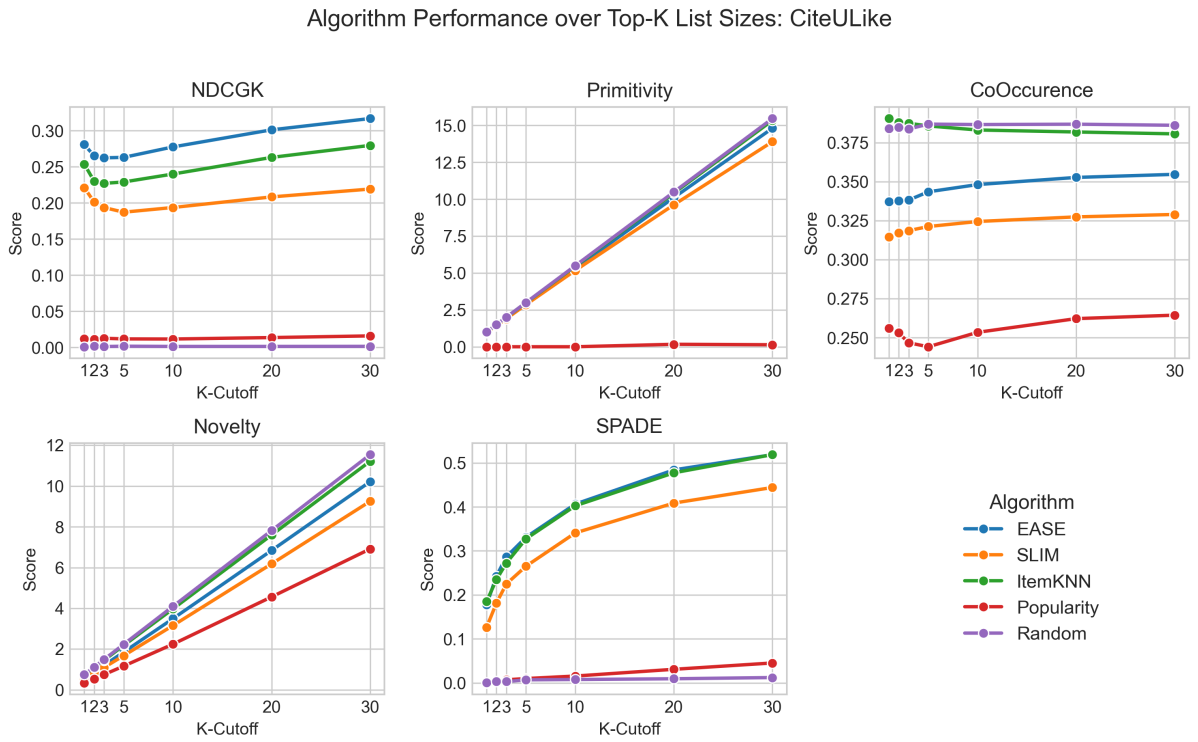

Figure 2: Performance comparison of baseline recommendation algorithms (EASE, SLIM, ItemKNN, and Popularity) across varying recommendation list sizes ($K \in \{1, 2, 3, 5, 10, 20, 30\}$). The subplots depict the evaluation trajectories for accuracy (NDCG) and beyond-accuracy metrics (Primitivity, Co-Occurrence Surprise, Novelty, and SPADE) as the cutoff threshold increases.

Traditional metrics, such as Novelty, Primitivity, and Co-Occurrence, evaluate recommended items irrespective of their actual relevance to the user. Consequently, an algorithm can mathematically maximize traditional beyond-accuracy metric scores by simply recommending random, irrelevant items. The empirical evaluation clearly illustrates this vulnerability, e.g., on the MIND dataset, the Random algorithm yields a near-zero ranking accuracy (NDCG of 0.0006), yet achieves very high Primitivity (5.4989) and Co-Occurrence (0.281789) scores (table 2, fig. 2). In contrast, SPADE ensures relevance by calculating serendipity scores only for predicted items included in the test set. By accurately identifying

Dataset	Algorithm	NDCGK	Primitivity	CoOccurence	Novelty	SPADE
Netflix	EASE	0.315685	4.844650	0.103907	1.168139	0.127368
Netflix	SLIM	0.280544	4.544350	0.100814	1.110615	0.117351
Netflix	ItemKNN	0.190461	3.705450	0.085582	0.993255	0.061379
Netflix	Popularity	0.109781	0.000000	0.076752	0.783255	0.029467
Netflix	Random	0.001734	5.498050	0.214283	inf	0.004001
ML-1M	EASE	0.334812	4.396189	0.095179	1.094719	0.101459
ML-1M	SLIM	0.299857	3.864540	0.089278	0.971579	0.085457
ML-1M	ItemKNN	0.258572	3.752361	0.085493	1.013478	0.068455
ML-1M	Popularity	0.149624	0.000000	0.068181	0.687975	0.030897
ML-1M	Random	0.007858	5.476885	0.179450	2.941511	0.014116
MIND	EASE	0.528621	5.281100	0.160480	1.738990	0.234498
MIND	SLIM	0.468111	5.185100	0.156472	1.632805	0.210349
MIND	ItemKNN	0.360481	5.016600	0.151883	1.917122	0.194649
MIND	Popularity	0.063749	0.000000	0.116312	1.038346	0.036538
MIND	Random	0.000617	5.498997	0.281789	inf	0.003692
ML-20M	EASE	0.313071	4.349293	0.116474	1.230299	0.131990
ML-20M	SLIM	0.277540	4.074596	0.114782	1.169247	0.115773
ML-20M	ItemKNN	0.238583	4.194949	0.108970	1.134467	0.101111
ML-20M	Popularity	0.127909	0.000000	0.087457	0.783185	0.031791
ML-20M	Random	0.001366	5.496224	0.221319	inf	0.002831
CiteULike	EASE	0.277414	5.330776	0.348068	3.494691	0.406407
CiteULike	SLIM	0.193589	5.174007	0.324404	3.157389	0.340890
CiteULike	ItemKNN	0.239903	5.459838	0.383043	3.975786	0.402518
CiteULike	Popularity	0.011624	0.000903	0.253401	2.244291	0.015532
CiteULike	Random	0.001361	5.496300	0.386524	4.097694	0.007649

Table 2
Cross-domain evaluation of baseline recommendation models (EASE, SLIM, ItemKNN, and Popularity) at a top-K cutoff of 10. The table details ranking accuracy alongside beyond-accuracy metrics to illustrate the trade-off between relevance and serendipitous discovery across the different datasets.

random recommendations as unconstrained noise, SPADE successfully penalizes this lack of relevance and assigns a proportionally low serendipity score of 0.0036 (table 2, fig. 2).

This strict relevance constraint also resolves a critical blind spot when evaluating high-performing models. While traditional metrics, e.g., show that EASE on ML-1M achieves both high ranking accuracy (NDCG of 0.3348) and high novelty (1.09, table 2), looking at these metrics in isolation cannot verify whether the novel recommendations are actually the ones the user found relevant.

Finally, SPADE successfully passes sanity checks by confirming that the non-personalized Random and Popularity baselines yield minimal serendipitous value. True serendipity inherently requires recommending relevant items that avoid both high similarity to the user's history and high global popularity. This expectation is perfectly mirrored by the Popularity and Random baselines, which consistently yield the lowest SPADE scores across all evaluated datasets (e.g., 0.0036 for Random on MIND, and 0.0294 for Popularity on Netflix).

Conclusion In conclusion, SPADE provides a robust, empirical framework for algorithm evaluation in recommender systems. By mathematically grounding serendipity in test-set relevance, SPADE prioritizes relevance and penalizes highly personalized or popular recommendations. Consequently, SPADE enables researchers to reliably distinguish genuinely serendipitous recommendations from those that simply exploit isolated beyond-accuracy metrics.

Declaration on Generative AI

During the preparation of this work, the author(s) used Gemini in order to: Grammar and spelling check, Paraphrase and reword. After using this tool/service, the author(s) reviewed and edited the content as needed and take(s) full responsibility for the publication's content.

References

[1] M. Ge, C. Delgado-Battenfeld, D. Jannach, Beyond accuracy: evaluating recommender systems by coverage and serendipity, in: Proceedings of the Fourth ACM Conference on Recommender Systems, RecSys '10, Association for Computing Machinery, New York, NY, USA, 2010, p. 257–260. URL: https://doi.org/10.1145/1864708.1864761. doi:10.1145/1864708.1864761.

[2] R. J. Ziarani, R. Ravanmehr, Serendipity in recommender systems: a systematic literature review, Journal of Computer Science and Technology 36 (2021) 375–396.

[3] T. T. Nguyen, P.-M. Hui, F. M. Harper, L. Terveen, J. A. Konstan, Exploring the filter bubble: the effect of using recommender systems on content diversity, in: Proceedings of the 23rd International Conference on World Wide Web, WWW '14, Association for Computing Machinery, New York, NY, USA, 2014, p. 677–686. URL: https://doi.org/10.1145/2566486.2568012. doi:10.1145/2566486.2568012.

[4] Y. Xi, M. Weng, W. Chen, C. Yi, D. Chen, G. Guo, M. Zhang, J. Wu, Y. Jiang, Q. Liu, Y. Yu, W. Zhang, Bursting filter bubble: Enhancing serendipity recommendations with aligned large language models, in: Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2, KDD '25, Association for Computing Machinery, New York, NY, USA, 2025, p. 5059–5070. URL: https://doi.org/10.1145/3711896.3737199. doi:10.1145/3711896.3737199.

[5] A. Maksai, F. Garcin, B. Faltings, Predicting online performance of news recommender systems through richer evaluation metrics, in: Proceedings of the 9th ACM Conference on Recommender Systems, RecSys '15, Association for Computing Machinery, New York, NY, USA, 2015, p. 179–186. URL: https://doi.org/10.1145/2792838.2800184. doi:10.1145/2792838.2800184.

[6] M. Kaminskas, D. Bridge, Diversity, serendipity, novelty, and coverage: A survey and empirical analysis of beyond-accuracy objectives in recommender systems, ACM Trans. Interact. Intell. Syst. 7 (2016). URL: https://doi.org/10.1145/2926720. doi:10.1145/2926720.

[7] D. Kotkov, J. A. Konstan, Q. Zhao, J. Veijalainen, Investigating serendipity in recommender systems based on real user feedback, in: Proceedings of the 33rd Annual ACM Symposium on Applied Computing, SAC '18, Association for Computing Machinery, New York, NY, USA, 2018, p. 1341–1350. URL: https://doi.org/10.1145/3167132.3167276. doi:10.1145/3167132.3167276.

[8] M. Jenders, T. Lindhauer, G. Kasneci, R. Krestel, F. Naumann, A serendipity model for news recommendation, in: Joint German/Austrian Conference on Artificial Intelligence (Künstliche Intelligenz), Springer, 2015, pp. 111–123.

[9] Y. C. Zhang, D. O. Séaghdha, D. Quercia, T. Jambor, Auralist: introducing serendipity into music recommendation, in: Proceedings of the Fifth ACM International Conference on Web Search and Data Mining, WSDM '12, Association for Computing Machinery, New York, NY, USA, 2012, p. 13–22. URL: https://doi.org/10.1145/2124295.2124300. doi:10.1145/2124295.2124300.

[10] G. Shani, A. Gunawardana, Evaluating recommendation systems, in: Recommender systems handbook, Springer, 2010, pp. 257–297.

[11] T. De Pessemier, S. Dooms, L. Martens, Comparison of group recommendation algorithms, Multimedia tools and applications 72 (2014) 2497–2541.

[12] C.-D. Wang, Z.-H. Deng, J.-H. Lai, P. S. Yu, Serendipitous recommendation in e-commerce using innovator-based collaborative filtering, IEEE transactions on cybernetics 49 (2018) 2678–2692.

[13] T. Murakami, K. Mori, R. Orihara, Metrics for evaluating the serendipity of recommendation lists, in: Annual conference of the Japanese society for artificial intelligence, Springer, 2007, pp. 40–46.

[14] M. Kaminskas, D. Bridge, Measuring surprise in recommender systems, in: Proceedings of the

workshop on recommender systems evaluation: dimensions and design (Workshop programme of the 8th ACM conference on recommender systems), Citeseer, 2014.
[15] M. Kaminskas, D. Bridge, Diversity, serendipity, novelty, and coverage: a survey and empirical analysis of beyond-accuracy objectives in recommender systems, ACM Transactions on Interactive Intelligent Systems (TiiS) 7 (2016) 1–42.
[16] N. Kawamae, Real time recommendations from connoisseurs, in: Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining, 2015, pp. 537–546.
[17] M. N. Jelassi, S. Ben Yahia, E. Mephu Nguifo, Towards more targeted recommendations in folksonomies, Social Network Analysis and Mining 5 (2015) 68.
[18] M. Chen, Y. Wang, C. Xu, Y. Le, M. Sharma, L. Richardson, S.-L. Wu, E. Chi, Values of user exploration in recommender systems, in: Proceedings of the 15th acm Conference on recommender systems, 2021, pp. 85–95.
[19] J. Sá, V. Queiroz Marinho, A. R. Magalhães, T. Lacerda, D. Goncalves, Diversity vs relevance: A practical multi-objective study in luxury fashion recommendations, in: Proceedings of the 45th International ACM SIGIR Conference on research and development in information retrieval, 2022, pp. 2405–2409.
[20] T. Kaya, C. Kaleli, A novel top-n recommendation method for multi-criteria collaborative filtering, Expert Systems with Applications 198 (2022) 116695.
[21] A. Maksai, F. Garcin, B. Faltings, Predicting online performance of news recommender systems through richer evaluation metrics, in: Proceedings of the 9th ACM Conference on Recommender Systems, 2015, pp. 179–186.
[22] D. Kotkov, J. Veijalainen, S. Wang, T. A. Majchrzak, P. Traverso, K.-H. Krempels, V. Monfort, A serendipity-oriented greedy algorithm for recommendations (2017).
[23] L. D. Babu, E. D. Raj, et al., Evaluation of similarity based recommender using fuzzy inference system, in: 2015 International Conference on Computing and Network Communications (CoCoNet), IEEE, 2015, pp. 576–581.
[24] C. Nishioka, J. Hauk, A. Scherp, Towards serendipitous research paper recommender using tweets and diversification, in: International Conference on Theory and Practice of Digital Libraries, Springer, 2019, pp. 339–343.
[25] C. Nishioka, J. Hauke, A. Scherp, Influence of tweets and diversification on serendipitous research paper recommender systems, PeerJ Computer Science 6 (2020) e273.
[26] Y. Ren, G. Li, W. Zhou, A survey of recommendation techniques based on offline data processing, Concurrency and Computation: Practice and Experience 27 (2015) 3915–3942.
[27] P. Adamopoulos, A. Tuzhilin, On unexpectedness in recommender systems: Or how to better expect the unexpected, ACM Transactions on Intelligent Systems and Technology (TIST) 5 (2014) 1–32.
[28] R. J. Ziarani, R. Ravanmehr, Deep neural network approach for a serendipity-oriented recommendation system, Expert Systems with Applications 185 (2021) 115660.
[29] N. Kawamae, Serendipitous recommendations via innovators, in: Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval, 2010, pp. 218–225.
[30] X. Lin, H. Chen, C. Pei, F. Sun, X. Xiao, H. Sun, Y. Zhang, W. Ou, P. Jiang, A pareto-efficient algorithm for multiple objective optimization in e-commerce recommendation, in: Proceedings of the 13th ACM Conference on recommender systems, 2019, pp. 20–28.
[31] M. T. Ribeiro, N. Ziviani, E. S. D. Moura, I. Hata, A. Lacerda, A. Veloso, Multiobjective pareto-efficient approaches for recommender systems, ACM Transactions on Intelligent Systems and Technology (TIST) 5 (2014) 1–20.
[32] P. Li, A. Tuzhilin, Deep pareto reinforcement learning for multi-objective recommender systems, arXiv preprint arXiv:2407.03580 (2024).
[33] L. Xiao, Z. Min, Z. Yongfeng, G. Zhaoquan, L. Yiqun, M. Shaoping, Fairness-aware group recommendation with pareto-efficiency, in: Proceedings of the eleventh ACM conference on recommender systems, 2017, pp. 107–115.

[34] K. Peng, M. Raghavan, E. Pierson, J. Kleinberg, N. Garg, Reconciling the accuracy-diversity trade-off in recommendations, in: Proceedings of the ACM Web Conference 2024, 2024, pp. 1318–1329.
[35] S. Lin, C. Gao, J. Chen, S. Zhou, B. Hu, Y. Feng, C. Chen, C. Wang, How do recommendation models amplify popularity bias? an analysis from the spectral perspective, arXiv preprint arXiv:2404.12008 (2024).
[36] G. J. Bouma, Normalized (pointwise) mutual information in collocation extraction, 2009. URL: https://api.semanticscholar.org/CorpusID:2762657.
[37] H. E. Robbins, An empirical bayes approach to statistics, in: Breakthroughs in Statistics: Foundations and basic theory, Springer, 1992, pp. 388–394.
[38] H. Wang, B. Chen, W.-J. Li, Collaborative topic regression with social regularization for tag recommendation, in: IJCAI, 2013.
[39] F. Wu, Y. Qiao, J.-H. Chen, C. Wu, T. Qi, J. Lian, D. Liu, X. Xie, J. Gao, W. Wu, et al., Mind: A large-scale dataset for news recommendation, in: Proceedings of the 58th annual meeting of the association for computational linguistics, 2020, pp. 3597–3606.
[40] F. M. Harper, J. A. Konstan, The movielens datasets: History and context, ACM Trans. Interact. Intell. Syst. 5 (2015). URL: https://doi.org/10.1145/2827872. doi:10.1145/2827872.
[41] L. Michiels, R. Verachtert, B. Goethals, Recpack: An (other) experimentation toolkit for top-n recommendation using implicit feedback data, in: Proceedings of the 16th ACM Conference on Recommender Systems, 2022, pp. 648–651.
[42] O. Barkan, R. Hirsch, O. Katz, A. Caciularu, N. Koenigstein, Anchor-based collaborative filtering, in: Proceedings of the 30th ACM International Conference on Information & Knowledge Management, 2021, pp. 2877–2881.
[43] A. B. Melchiorre, N. Rekabsaz, C. Ganhör, M. Schedl, Protomf: Prototype-based matrix factorization for effective and explainable recommendations, in: Proceedings of the 16th ACM Conference on Recommender Systems, 2022, pp. 246–256.