End-to-end Gradient-Based Diversity Optimization for Top-k Recommendations

Abstract—**Recommender systems play an integral part in countless areas of modern society, including social media, the entertainment industry, and online shopping. Biases in the data are often responsible for homogenization of recommendations, thereby exacerbating echo chambers, social polarization, reduced novelty, content homogenization, and unfair representation. Diversity is therefore crucial to mitigate these socially relevant issues. Current state-of-the-art approaches diversify top-k recommendations by filtering the outputs of biased recommender systems, introducing inefficient and unnecessary post-processing steps. In this work, we improve this diversification process by promoting diversity through directly fine-tuning recommender systems models. Starting with training a non-diverse recommender system, we then learn to diversify by targeting a novel diversity-promoting objective, yielding diverse recommendations without post-processing. In a wide range of experiments, we consistently observe significant improvements in top-k diversity while maintaining model accuracy. Moreover, we demonstrate that our diversity objective converges efficiently with only a few steps.**

Index Terms—**Diversity, Recommendation**

I. Introduction

Recommender systems (RS) play a central role in shaping users' online experiences, influencing decisions across domains such as shopping, entertainment, social media, and news [1]. As they become increasingly pervasive, recommender systems do more than affecting individual choices — they also shape collective behavior, cultural trends, and political discourse [2].

Traditionally, the design of recommender systems has focused on maximizing accuracy, recommending items that users are most likely to engage with. However, focusing on a narrow engagement objective has led to several unintended consequences. Recommendations often become homogeneous, limiting exposure to novel content [3] and causing user fatigue. Moreover, recommender systems that do not account for diverse content tend to amplify popularity bias, while marginalizing niche ideas or smaller providers [4]. At a societal level, these systems may reinforce filter bubbles, trapping users in ideological echo chambers and contributing to polarization [5].

To address these concerns, *diversity* has emerged as a critical objective in the design of recommender systems [6]. Recent user studies and online experiments fount that add diversity does not simply trade accuracy for its own sake—when done well, it can enhance user satisfaction and engagement [7], [8], [9]. However, enhancing diversity often comes at the cost of accuracy, posing a fundamental trade-off at the core of recommender-systems research. Prior work tackles this trade-off in two main ways (1) *post-hoc re-ranking methods*, and (2) *learning-based methods*.

Recent research has shown that *post-hoc re-ranking methods* can substantially improve the diversity of recommendation lists while carefully quantifying their impact on traditional accuracy metrics [10], [11], [12]. These methods re-rank the output of a trained model to promote diversity, making them model-agnostic and easy to integrate into existing systems. Although they offer explicit control over the relevance–diversity trade-off, this trade-off remains a fundamental limitation in practice, as the re-ranking process can still degrade recommendation accuracy [13]. In contrast, *learning-based methods* incorporate diversity directly into the model training objective [14], [15], [16], [17], often achieving higher diversity while maintaining comparable relevance to models optimized solely for accuracy. However, these methods are typically model-specific, slower to converge, and sensitive to hyperparameters that govern the relevance–diversity balance. Moreover, they can obscure the underlying cause of diversity improvements, making them harder to interpret and tune. In addition to model-specific approaches, another line of learning-based methods adopts a data-centric perspective by incorporating beyond-accuracy objectives into techniques such as data augmentation, reweighting, and debiasing [18], [19], [20], [21], [22]. However, these efforts seldom address diversity explicitly.

To address the limitations of existing diversification approaches, we propose a unified framework that leverages differentiable ranking to optimize diversity in top-k recommendations in a scalable, interpretable, and model-agnostic manner. At the core of our framework is a simple yet effective diversity objective that can be seamlessly integrated into standard relevance-based training through gradient-based optimization—without requiring changes to model architecture or inference procedures. Building on this objective, we explore both explicit and implicit strategies for incorporating diversity into the training process. Specifically, we introduce two complementary algorithms:

- *Direct Diversity-guided Tuning* explicitly incorporates the diversity objective into model training via a joint relevance–diversity loss, using gradient signals to fine-tune model parameters for improved diversity with fast convergence.
- *Meta Diversity-guided Reweighting* adopts a data-centric approach, implicitly optimizing for diversity by adjusting instance-level weights based on their contribution to both relevance and diversity.

Together, these methods bridge the gap between post-hoc re-ranking and learning-based diversification—offering a flexible and principled alternative that preserves model generality and training efficiency.

We evaluate both methods on *matrix factorization* and *neural collaborative filtering* models across five real-world datasets. Our results show that our methods consistently improve diversity with neglectable accuracy loss, outperforming state-of-the-art baselines. Concretely, we make the following contributions: (1) We propose a unified, model-agnostic framework for diversity-aware recommendation that supports both end-to-end training and integration of explicit and implicit optimization via a joint relevance–diversity objective; (2) we demonstrate that both methods achieve substantial diversity gains with neglectable accuracy loss across diverse real-world datasets, outperforming strong baselines in relevance-diversity trade-off. (3) We show that our diversity-guided optimization not only consistently improves diversity within the top-k recommendations (as defined in the objective), but also yields benefits beyond the in-objective range; and (4) we highlight the potential of data-centric methods for improving diversity, as evidenced by the positive results from implicit diversity-guided optimization.

[ag]: What is the difference between contributions (2) and (4)? Both are about empirical evaluation. Contribution (3) is not concrete enough: it sounds quite vague. [tz]: Revised, in (4), I try to highlight that the data-centric method to improve diversity is promising, which is a (probably) blank area.

II. Related Work

[ag]: First few sentences of related work is repetition from intro.

Traditional recommender systems, such as matrix factorization and neural collaborative filtering, are primarily optimized for accuracy, aiming to predict user preferences based on historical interactions. While effective in surfacing relevant items, these models often suffer from popularity bias and homogeneity, repeatedly recommending popular or similar content. Such redundancy can limit users' exposure to novel or diverse items, reinforce existing preferences, and ultimately degrade user engagement and satisfaction.[3]. To address these shortcomings, recent research has emphasized the importance of diversity in recommendation lists. A diverse recommendation list covers a broader range of user interests, item attributes, or semantic categories, thereby mitigating filter bubbles, promoting serendipity, and increasing user satisfaction over time. We group the literature into three broad categories.

Post-hoc approach. Re-ranking approaches enhance diversity by adjusting the output of an existing recommendation model. These methods are model-agnostic and apply a secondary diversification step after prediction [10], [23], [24], [25], [26], [27], [12]. For example, Maximal Marginal Relevance (MMR) [10] greedily selects items that balance relevance and dissimilarity, while Diversity-Weighted Utility Maximization (DUM) [28] optimizes a modular utility under submodular diversity constraints. Determinantal Point Processes (DPP) [12] select diverse item sets based on kernel determinants. While widely adopted, these methods are limited by their dependence on the initial ranked list and lack of feedback loops for learning. They often suffer notable accuracy loss as diversity increases [13]. In contrast, our methods optimize diversity during model training via gradient-based objectives, enabling improved diversity without post-processing. This avoids performance degradation often observed in re-ranking and allows for end-to-end training.

[ag]: Is there a reference to backup the claim that "[these methods] often suffer notable accuracy loss as diversity increases?" [tz]: add 1 reference.

Learning approach. Learning-based methods incorporate diversity into model training. These include regularization strategies that penalize similarity among recommended items [15], [29], or listwise loss formulations that optimize relevance-diversity trade-offs jointly [16]. These methods generally outperform post-hoc re-ranking in balancing relevance and diversity, but are often model-specific, computationally intensive, and lack interpretability. In contrast, our proposed methods are model-agnostic. They allow for fine-tuning a relevance-only model using a differentiable diversity objective with minimal additional complexity. Recent work has also explored diversification through embedding learning. For example, DGCN [30] adjusts neighborhood sampling in GCNs to enhance long-tail exposure, and DGRec [31] promotes category coverage via GNN-based retrieval. Although effective, these methods often entail architectural modifications, auxiliary modules, or adversarial objectives. In contrast, our methods retain the original architecture and inference pipeline, modifying only the training signals. This ensures no added inference cost and broad compatibility with standard recommenders.

Data-centric approach. Advances in data-centric recommender systems focus on improving model performance by modifying the training data, rather than altering model architectures [32]. These methods aim to increase accuracy, alleviate noise, and reduce bias in user-item interactions, often through techniques such as attribute completion, denoising, and data distillation [18], [19], [20]. Several works address systemic biases — such as popularity bias and conformity effects [21], [22]. While most existing efforts primarily target accuracy and fairness, relatively few explicitly optimize for diversity. Our proposed implicit diversity-guided optimization method demonstrates that effectively adjusting the importance of training samples can lead to significant diversity gains, highlighting a promising direction for promoting diversity through data-centric learning.

III. Preliminaries

Our approach builds on the foundations of recommender systems and the concept of diversity, both of which we introduce next.

a) Recommender system: A recommender system aims to predict those items in the set $\mathcal{I} = \{i_1, i_2, ..., i_m\}$ that are relevant for users in the set $\mathcal{U} = \{u_1, u_2, ..., u_n\}$. It usually operates on a partially observed user-item rating matrix $\mathbf{R} \in \mathbb{R}^{n,m}$, where each entry $\mathbf{R}_{u,i} \in \mathbb{R}$ represents user u's rating of item i, typically restricted to a few possibilities such as $\{1, \ldots, 5, \}$. We denote $\mathbf{r}_u \in \mathbb{R}^m$ the uth row of $\mathbf{R}$, which contains user u's ratings over all items.

TABLE I
NOTATION

Variable	Meaning
n	Number of users
m	Number of items
k	Number of items in recommendation
$\mathcal{U}$	A set of n users
$\mathcal{I}$	A set of m items
$\mathcal{C}$	A set of categories
$\mathbf{R}$	Partially observed user rating matrix
$\mathbf{D}$	Distance matrix of items
$\mathcal{F}$	Embedding-based prediction model
$\mathbf{\Theta}$	Model parameter
$\mathbf{x}_u$	d dimensional embedding of user u
$\mathbf{y}_i$	d dimensional embedding of item i
$\mathbf{l}_u$	m dimensional binary vector

For example, user $u =$ *Ulysses* might rate the item $i =$ *Odyssey* with $\mathbf{R}_{u,i} = 5$, but would u also enjoy $i =$ *Iliad* even though u has not read it? To answer such questions, a recommender system predicts the ratings using a model $\mathcal{F}$ with parameters $\mathbf{\Theta}$, producing predicted ratings

$$\tilde{\mathbf{R}}_{u,i} = \mathcal{F}_{\mathbf{\Theta}}(u,i)$$

for all users $u \in \mathcal{U}$ and items $i \in \mathcal{I}$. In practice, most entries in $\mathbf{R}$ are unobserved, resulting in extremely sparse data. We represent the observed ratings as a set $\Omega = \{(u, i, \mathbf{R}_{u,i})\}^N$, where $N \ll nm$. This sparse dataset serves as the basis for training and evaluating the recommender model. We use $\Omega_{\mathrm{T}}, \Omega_{\mathrm{V}} \subset \Omega$ to denote the training and validation subsets of the observed data, respectively. There are many approaches to learn such a system. We consider two widely-used and highly-successful methods: *matrix factorization* (MF) [33] and *neural collaborative filtering* (NCF) [34], starting with non-negative matrix factorization.

Matrix factorization [33]: Matrix factorization is a classic approach in collaborative filtering, seeking to represent the rating matrix as a product of low-rank matrices. That is each user u and item i are represented by a d-dimensional vector $\mathbf{x}_u \in \mathbb{R}^d$ and $\mathbf{y}_i \in \mathbb{R}^d$ embedding in the form of parameters $\mathbf{\Theta}$. We compute the predicted rating as the inner product

$$\tilde{\mathbf{R}}_{u,i} = \mathbf{x}_u^\top \mathbf{y}_i \tag{1}$$

of the low-rank representations. While usually performing well, it cannot directly model complicated non-linear relationships. To address this, we also consider neural collaborative filtering.

Neural collaborative filtering [34]: Extending matrix-factorization methods, neural collaborative filtering aims to capture non-linear interactions between user and item embeddings by replacing the linear prediction model with a multi-layer perceptron (MLP)

$$\tilde{\mathbf{R}}_{u,i} = \mathrm{MLP}([\mathbf{x}_u; \mathbf{y}_i]) \ , \tag{2}$$

where $[\cdot;\cdot]$ denotes concatenation. Here, both the embedding vectors $\mathbf{x}_u, \mathbf{y}_i$ and the weights of the MLP are part of the model parameters $\mathbf{\Theta}$. To obtain a model $\mathbf{\Theta}$ for matrix factorization (respectively, neural collaborative filtering) which approximates ratings Ω_{T} well without overfitting, we minimize the regularized mean squared error (MSE)

$$\mathbf{\Theta}^* = \arg\min_{\mathbf{\Theta}} \mathcal{L}_{\mathrm{MSE}}(\mathcal{F}) + \lambda \|\mathbf{\Theta}\|_2^2 \ , \tag{3}$$

where $\mathcal{L}_{\mathrm{MSE}}(\mathcal{F})$ is defined as $\sum_{(u,i)\in\Omega_{\mathrm{T}}} (\mathbf{R}_{u,i} - \tilde{\mathbf{R}}_{u,i})^2$.

b) Top-k recommendations: Given a trained recommender system $\mathcal{F}_{\mathbf{\Theta}^*}$, we obtain for each user a predicted rating vector $\tilde{\mathbf{r}}_u \in \mathbb{R}^m$, where each entry $\tilde{\mathbf{r}}_u(i)$ represents the estimated relevance of item i out of m candidate items. We construct a binary top-k recommendation vector $\mathbf{l}_u \in \{0,1\}^m$, where $\mathbf{l}_u(i) = 1$ if item i is among the top-k, and 0 otherwise. We aim to learn a selection function $f_{\mathrm{top}_k} : \mathbb{R}^m \to \{0,1\}^m$ which maps a relevance score vector $\tilde{\mathbf{r}}_u \in \mathbb{R}^m$ to a binary indicator vector that identifies the top-k recommended items. Specifically, the selected set is given by

$$\mathbf{l}_u(i) = \begin{cases} 1 & \text{if } i \in Z_u(k) = \{z_1, \ldots, z_k\}, \\ 0 & \text{otherwise} \ , \end{cases}$$

where $Z_u(k)$ denotes the indices of the k highest-scoring items. To obtain $Z_u(k)$, we sort the items according to their predicted relevance scores in descending order, yielding a permutation $\mathbf{z}_u = (z_1, \ldots, z_m)$ such that $\tilde{\mathbf{r}}_u(z_i) \geq \tilde{\mathbf{r}}_u(z_j)$ for all $i < j$. The top-k recommendation set then corresponds to the first k elements of this ranked list. Next, we outline how to quantify the diversity of the resulting recommendation sets.

c) Diversity: A diverse list of recommendations refers to a list that covers a broad range of item characteristics, such as topics, categories, or genres. However, standard approaches like matrix factorization or neural collaborative filtering often fail to produce diverse recommendations, due to inherent biases in data [3]. To address this shortcoming, we first introduce measures used to evaluate diversity, then we discuss how augment the diversity of methods, before we move on to describing how we incorporate these diversity to promote diversity more efficiently.

To measure diversity, we first leverage meta information for each item, such as genres, categories, or topics, like {*Action*, *Drama*,...} in the context of movies. Coverage-based diversity seeks to measure the exposure of users to items from diverse categories. For this, it simply counts the number of categories covered in a recommendation list. Mapping each item to a set of categories $\mathcal{C}_i \subseteq \mathcal{C} = \{c_1, ..., c_C\}$ for each item $i \in \mathcal{I}$, we then define the *coverage-based diversity* of $Z_u(k)$ as the set

$$D_{\mathcal{C}}(Z_u(k)) = |\cup_{i \in Z_u(k)} \mathcal{C}_i| \ . \tag{4}$$

of unique categories covered by a subset $Z_u(k) \in \mathcal{I}$ of items [35]. However, just counting distinct categories is an inherently coarse diversity measure, which does not capture pairwise relationships between items.

To address this issue, we introduce our *distance-based diversity*, which leverages a pairwise item-item affinity matrix to quantify how different the recommended items are from another, offering a fine-grained and flexible diversity measure. Inspired by the *max dispersion problem* [36] and the *max-sum diversification problem* [14], we use distances to measure

Algorithm 1 Direct Diversity-Guided Tuning (DDT)

Require: Initialization model parameter $\mathbf{\Theta}$, training data Ω_{T}, item distance matrix $\mathbf{D}$, trade-off β, learning rate η

for each epoch from 1 to T **do**
  for each mini-batch $\mathcal{B} \in \Omega_{\mathrm{T}}$ **do**
    $\tilde{\mathbf{R}}_{u,i} \leftarrow \mathcal{F}_{\mathbf{\Theta}}(u, i)$, for all $(u, i, \mathbf{R}_{u,i}) \in \mathcal{B}$
    $\mathcal{L}_{\mathrm{MSE}} \leftarrow$ Compute $\mathcal{L}_{\mathrm{MSE}}(\tilde{\mathbf{R}}_{u,i}, \mathbf{R}_{u,i}; \mathcal{B})$ with (3)
    $\mathcal{L}_{\mathrm{DDRO}} \leftarrow$ compute DDRO with (9)
    $\mathcal{L}_{\mathrm{JOINT}} \leftarrow \beta \cdot \mathcal{L}_{\mathrm{MSE}} - (1-\beta) \cdot \mathcal{L}_{\mathrm{DDRO}}$
    $\mathbf{\Theta} \leftarrow \mathbf{\Theta} - \eta \cdot \nabla_{\mathbf{\Theta}} \mathcal{L}_{\mathrm{JOINT}}$
return $\mathbf{\Theta}$

diversity. Let $\mathbf{S} \in \mathbb{R}^{m \times m}$ be the pairwise affinity matrix of all items in $\mathcal{I}$. We define the *distance-based diversity* of set $Z_u(k) \in \mathcal{I}$ using a recommendation indicator of $\mathbf{l}_u$ as

$$D_{\mathbf{S}}(Z_u(k)) = \frac{2}{k(k-1)} \sum_{i=1}^{m} \sum_{j=1}^{m} \mathbf{l}_u(i)\mathbf{l}_u(j)(1 - \mathbf{S}_{i,j}) \ , \quad (5)$$

yielding a *differentiable* function, enabling to utilize an efficient diversity-guided gradient-based optimization, described next.

IV. Problem

Having introduced top-k recommender systems and diversity measures, we now bring them together to define our problem statement. In a nutshell, we aim to increase the diversity of the top-k items recommended by the system. Rather than indirectly obtain relevant yet diverse k recommendations via post-processing and selecting, we aim to change the recommender system directly to yield k diverse yet relevant items. To go beyond post-processing, we aim to learn a recommendation model that inherently balances diversity with relevance. This requires a differentiable top-k diversity objective, compatible with the learning objective of classical recommender systems, as detailed in the following.

a) Diversity reward objective (DRO): Given a user's predicted item relevance scores $\tilde{\mathbf{r}}_u \in \mathbb{R}^m$, we compute the top-$k$ recommendation vector $\mathbf{l}_u = \mathrm{top}_k(\tilde{\mathbf{r}}_u)$ as described earlier. Using the distance-based diversity measure $D_{\mathbf{S}}(\mathbf{l}_u)$ introduced in (5), we define the diversity reward objective as the average diversity across all users

$$\mathcal{L}_{\mathrm{DRO}}(k) = \frac{1}{n} \sum_{u=1}^{n} D_{\mathbf{S}}(Z_u(k)). \quad (6)$$

where $\mathbf{l}_u$ is the top-k binary vector for user u. While the diversity reward objective $\mathcal{L}_{\mathrm{DRO}}$ captures the diversity among top-k recommendations, it cannot be directly optimized using gradient-based methods due to the non-differentiability of the top_k operator. Specifically, computing top-k involves two discrete steps: sorting the predicted rating scores $\tilde{\mathbf{r}}_u \in \mathbb{R}^m$ to obtain a permutation of item indices $\mathbf{z}_u$, and applying hard thresholding to select the top-k items. Both steps prevent gradient backpropagation. We address this limitation by introducing a differentiable surrogate, as described next.

b) Differentiable diversity reward objective (DDRO): To overcome non-differentiability challenge in $\mathrm{top}_{\mathrm{k}}(\cdot)$, we adopt the framework of *differentiable ranking* [37], which provides a continuous relaxation of the sorting operation. The key idea is to replace the discrete permutation $\mathbf{z}_u$ with a soft ranking vector $\tilde{\mathbf{z}}_u^{(\varepsilon)} \in \mathbb{R}^m$, obtained by projecting the predicted scores $\tilde{\mathbf{r}}_u$ onto the permutahedron $\mathcal{P}_m$—the convex hull of all permutations of $(1, 2, \ldots, m)$. This projection is computed by solving the following entropy-regularized optimization problem

$$\tilde{\mathbf{z}}_u^{(\varepsilon)} = \mathrm{softrank}(\tilde{\mathbf{r}}_u) \coloneqq \arg\min_{r \in \mathcal{P}_m} \left\{ \frac{1}{\varepsilon} \langle \tilde{\mathbf{r}}_u, r \rangle + H(r) \right\}, \quad (7)$$

where $H(r)$ denotes an entropy regularizer and $\varepsilon > 0$ controls the smoothness of the approximation. This soft ranking enables a differentiable surrogate for top-k selection, allowing us to propagate gradients from the diversity objective back through the ranking step. In turn, this makes it possible to train the recommender model end-to-end using diversity-aware gradient updates. The obvious question now is, *does this ranking lead to sufficiently accurate top-k recommendations?* The better we approximate the discrete ordering, the more reliable are our top-k recommendations. Utilizing recent advancements of [37], we affirm this statement using Lemma 1 below.

Lemma 1 (Soft rank approximation [37])**.** *Given a rating vector $\tilde{\mathbf{r}}_u$, let $\tilde{\mathbf{z}}_u^{(\varepsilon)} \in \mathbb{R}^n$ be the soft rank vector obtained from optimizing* (7)*. Then, as $\varepsilon \to 0$, the soft ranks converge to the true ranks of $\tilde{\mathbf{r}}_u$ $\lim_{\varepsilon \to 0} \tilde{\mathbf{z}}_u^{(\varepsilon)} = \mathrm{rank}(\tilde{\mathbf{r}}_u)$, where $\mathrm{rank}(\tilde{\mathbf{r}}_u) \in \{1, \ldots, n\}^n$ denotes the integer-valued ranks (breaking ties arbitrarily).*

The softrank operation in (7) is differentiable and is even convex $\tilde{\mathbf{r}}_u$ whenever we let $\varepsilon \to \infty$ [37, Prop. 2]. This means the larger ε is, the 'easier' it is to optimize, but the further we depart from the true hard ranking. In practice, we choose a small ε that gives an exact approximation of hard rankings. Ensuring that soft ranking efficiently approximates hard ranking well, we now introduce the differentiable approximation as

$$\tilde{\mathbf{l}}_u(i) = \sigma_\tau(k - \tilde{\mathbf{z}}_u(i)) \quad (8)$$

for a scaled sigmoid function $\sigma_\tau(x) = [1 + \exp(-x/\tau)]^{-1}$ where the user-defined parameter τ regulates the smoothness-sharpness tradeoff. Replacing the soft-ranking-derived indicator yields the approximated top-k recommendations $\tilde{\mathbf{l}}_u$ in our *differentiable diversity reward objective*

$$\mathcal{L}_{\mathrm{DDRO}} = \frac{1}{n \cdot N} \sum_{u}^{n} \sum_{i=1}^{m} \sum_{j=1}^{m} \tilde{\mathbf{l}}_u(i)\tilde{\mathbf{l}}_u(j)\mathbf{D}_{i,j} \ . \quad (9)$$

c) DDPO-guided diversification in top-k recommendations: In practice, balancing both relevance and diversity can be challenging when their respective gradients

$$g_1 = \nabla_{\mathbf{\Theta}} \mathcal{L}_{\mathrm{MSE}} \quad \text{and} \quad g_2 = \nabla_{\mathbf{\Theta}} \mathcal{L}_{\mathrm{DDRO}}$$

point in different directions — formally, when the angle $\theta = \measuredangle g_1 \, g_2$ is large. In such cases, making progress on accuracy and

Algorithm 2 Meta Diversity Reweighting (MDR)

Require: Initialization model parameter $\mathbf{\Theta}$, training data Ω_{T}, item distance matrix $\mathbf{D}$, trade-off β, learning rate η

1: **for** each epoch from 1 to T **do**
2:   **for** mini-batches $\mathcal{B} \in \Omega_{\mathrm{T}}$ **do**
3:     $\tilde{\mathbf{R}}_{u,i} \leftarrow \mathcal{F}_{\mathbf{\Theta}}(u,i)$, for all $(u,i,\mathbf{R}_{u,i}) \in \mathcal{B}$
4:     $\mathbf{w} \leftarrow \vec{0}$
5:     $\mathcal{L}^{w}_{\mathrm{MSE}} \leftarrow \mathcal{L}^{w}_{\mathrm{MSE}}(\tilde{\mathbf{R}}_{u,i}, \mathbf{R}_{u,i}, \mathbf{w}; \mathcal{B})$ with (11)
6:     $\mathbf{\Theta}' \leftarrow \mathbf{\Theta} - \eta \cdot \nabla_{\mathbf{\Theta}} \mathcal{L}^{w}_{\mathrm{MSE}}$, update meta model.
7:     $\tilde{\mathbf{R}}'_{u,i} \leftarrow \mathcal{F}_{\mathbf{\Theta}'}(u,i)$, for all $(u,i,\mathbf{R}_{u,i}) \in \mathcal{B}$
8:     $\mathcal{L}_{\mathrm{JOINT}} \leftarrow \beta \cdot \mathcal{L}^{w}_{\mathrm{MSE}} - (1-\beta) \cdot \mathcal{L}_{\mathrm{DDRO}}$
9:     $\tilde{\mathbf{w}} \leftarrow \max(-\nabla_{\mathbf{w}} \mathcal{L}_{\mathrm{JOINT}}, 0)$;
10:     $\mathbf{w} \leftarrow \frac{\tilde{\mathbf{w}}}{\sum_j \tilde{w} + \epsilon)}$; normalization according to [38]
11:     $\mathbf{\Theta} \leftarrow \mathbf{\Theta} - \eta \cdot \nabla_{\mathbf{\Theta}} \mathcal{L}^{w}_{\mathrm{MSE}}(\tilde{\mathbf{R}}_{u,i}, \mathbf{R}_{u,i}, \mathbf{w}; \mathcal{B})$
12: **return** $\mathbf{\Theta}$

diversity is challenging. To address this conflict, we introduce a weight $\beta \in [0,1]$ and form the convex combination

$$\mathcal{L}_{\mathrm{JOINT}}(\mathbf{\Theta}) \;=\; \beta\,\mathcal{L}_{\mathrm{MSE}}(\mathbf{\Theta}) \;-\; (1-\beta)\,\mathcal{L}_{\mathrm{DDRO}}(\mathbf{\Theta})$$

whose gradient $\nabla_{\mathbf{\Theta}}\mathcal{L} \;=\; \beta\, g_1 \;-\; (1-\beta)\, g_2$ lies geometrically between g_1 and g_2, thus balancing relevancy with diversity while encouraging swift convergence. With this, we are ready to introduce our problem, Problem 2, as follows.

Problem 2. *Given a set of users $\mathcal{U}$ and items $\mathcal{I}$, a prediction model $\mathcal{F}$ parameterized by $\mathbf{\Theta}$ and a pair-wised item distance matrix $\mathbf{D}$. Our goal is to optimize the model parameter $\mathbf{\Theta}$ to achieve both high diversity and relevance in a top-k recommendation, guided by the jointly combined loss*

$$\mathcal{L}_{\mathrm{JOINT}} = \beta \cdot \mathcal{L}_{\mathrm{MSE}} - (1-\beta) \cdot \mathcal{L}_{\mathrm{DDRO}} \tag{10}$$

where $\beta \in [0,1]$ is a parameter that balances the trade-off between relevance and diversity.

While leading to an efficiently optimizable objective (10), joining them combines two diametrically opposed goals: diversity and relevance. Diverse recommendations are not necessarily the most 'relevant' ones, and vice versa. To deal with this balance both in the following.

V. Solution

To enable relevance-aware and diversity-sensitive top-k recommendation, we explore how to incorporate a joint loss $\mathcal{L}_{\mathrm{JOINT}}$ from (10) into standard end-to-end training. We propose two complementary strategies that differ in how they leverage this joint objective. The first method, *Direct Diversity-guided Tuning* (DDT), performs explicit optimization by applying gradient descent directly on the joint loss. This straightforward approach tightly couples relevance and diversity during each model update. The second method, *Meta Diversity-guided Reweighting* (MDR), adopts an implicit approach: it treats the joint objective as a meta-loss to guide per-example reweighting. This strategy enables the model to prioritize training instances that indirectly promote diversity while optimizing a relevance-based loss. Together, these methods offer two ways to operationalize the relevance-diversity trade-off—one through direct loss shaping, the other through adaptive data selection.

a) Direct diversity-guided tuning: To obtain top-k relevant and diverse recommendations — without relying on post-hoc re-ranking or heuristic interventions — we propose an integrated approach. Our approach DDT integrates a β-balanced relevance-diversity loss into a standard end-to-end training pipeline. Specifically, we use gradient-based joint optimization, which promotes relevance and diversity. The overall procedure is outlined in Algorithm 1. This method preserves the simplicity and efficiency of conventional end-to-end training while explicitly aligning the optimization process with our dual objective. By embedding both goals within a single differentiable framework, DDT eliminates the need for separate re-ranking stages.

b) Meta diversity-guided reweighting: Directly optimizing a joint loss often suffers from conflicting gradient directions between relevance (g_1) and diversity (g_2). To avoid this, we adopt a meta-learning-inspired approach in which we treat the joint loss as a *meta-objective* [38] — i.e., an objective for finding the problem definition rather than its model. Here we ask *"Which training examples, if included in the update, will reduce both training loss and improve diversity?"* To this end, we estimate and assign importance weights $w_{u,i} \in [0,1]$ to training samples (u,i,r) for each batch $\mathcal{B} \subseteq \Omega_{\mathrm{T}}$, yielding

$$\mathcal{L}^{w}_{\mathrm{MSE}} = \sum_{(u,i)\in\mathcal{B}} w_{u,i} \left(\mathbf{R}_{u,i} - \tilde{\mathbf{R}}_{u,i}\right)^2 . \tag{11}$$

As shown in Algorithm 2, we begin each mini-batch update (line 4) by initializing the per-example weights $w_{u,i} = 0$, effectively ignoring all samples. We then perform a one-step inner update to obtain a temporary model Θ' using the weighted training loss $\mathcal{L}^{w}_{\mathrm{MSE}}$ (line 5–6), where w is currently all zero. Next, we re-evaluate predictions using the updated parameters (line 7) and compute the joint meta-loss $\mathcal{L}_{\mathrm{JOINT}}$ (line 8), combining relevance and diversity. For this, we compute the gradient of $\mathcal{L}_{\mathrm{JOINT}}$ with respect to w, producing a utility score for each sample. We rectify it via $\tilde{\mathbf{w}} = \max(-\nabla_{\mathbf{w}}\mathcal{L}_{\mathrm{JOINT}}, 0)$ (line 9), and normalize (line 10) to obtain the final per-example weights $\mathbf{w}$, enforcing $\sum_{(u,i)\in\mathcal{B}} w_{u,i} = 1$. Normalization ensures that the overall gradient magnitude — and thus the effective learning rate — remains consistent across training steps, similar to standard Stochastic Gradient Descent (SGD), which averages over the batch [38]. The final model update is then performed using this reweighted loss (line 11). This implicit diversity-guide optimization in MDR allows us to explore whether modifying the data sampling distribution alone can help achieve a decent relevance-diversity trade-off. We provide an empirical answer in Section VI, demonstrating that such implicit reweighting can be a practical and effective alternative to explicit multi-objective optimization.

c) Optimization strategy: In our experiments, we investigate two optimization strategies for applying DDT and MDR. The first strategy, *fine-tuning*, starts from a model pre-trained solely on the relevance objective and then applies DDT or

TABLE II
STATISTICS OF DATASETS AND DIVERSITY METRICS.

Dataset	$|\mathcal{U}|$	$|\mathcal{I}|$	$|\Omega|$	DRO($\mathcal{I}$)	$|\mathcal{C}|$
Coat	290	300	6 960	0.73	33
KuaiRec	1 411	3 327	4 676 570	0.91	31
Netflix	4 999	1 112	557 176	0.83	27
Yahoo-R2	4 050	5 000	684 782	0.26	58
MovieLens	6 040	3 706	1 000 208	0.83	18

MDR to adjust the model toward greater diversity. This setting reflects practical scenarios where relevance-optimized models are already deployed and diversity needs to be introduced with post-training for adjustment. The second strategy, *training from scratch*, initializes model parameters randomly and trains them end-to-end using either DDT or MDR. This setting allows us to assess the full capacity of each method to balance relevance and diversity from the outset. By comparing these two strategies, we examine the flexibility and robustness of our proposed approaches across different stages of the training lifecycle.

VI. EXPERIMENTS

In this section, we describe the experimental setup to evaluate the effectiveness of our proposed solutions, introducing the datasets, evaluation criteria, and implementation details.

a) Datasets: To evaluate our methods across different recommendation scenarios, we consider datasets from three domains: entertainment, product, and social recommendations, as detailed in Tab. II. To cover the *entertainment* domain, we use Netflix[1] [39] and MovieLens[2] [40] for user–movie recommendations, as well as Yahoo-R2[3] [41] for user–music recommendations. These datasets contain user ratings on a 5-point scale $[1, 5]$, along with genre or category annotations. For Netflix (respectively, Yahoo-R2), we randomly sample 3 000 items (respectively, 5 000) and retain users with at least 20 ratings (respectively, 100+ ratings). In the *product recommendation* setting, we use the Coat[4] [42], which captures user-coat interactions in e-commerce. It contains $[1, 5]$ ratings and item 'meta' attributes. Finally, for the *social recommendation* scenario, we consider the KuaiRec[5] [43], which is collected from a mobile video-sharing platform, which includes play duration, video length, and 'watch ratios' from 0 (never watched) to 2 (twice watched), which we linearly interpolate to 5-star ratings for consistency.

b) Baselines: We compare our algorithm against a broad set of state-of-the-art recommender-systems methods, as well as diversification techniques covering greedy, probabilistic, and graph-based strategies. To study the impact of diversification, we employ classical *Non-negative MF* (NMF) [44] as a relevance-only baseline that does not use any diversity mechanisms. We also include two baselines from the post-processing family: *Maximal Marginal Relevance* (MMR) [10] and *Diversity-weighted Utility Maximization* (DUM) [28], both greedy diversification techniques applied on top of NMF as the underlying model.

MMR greedily selects top-k items that maximizes a weighted combination of relevance and dissimilarity with previously selected items. DUM, on the other hand, uses a submodular combination of relevance and category-based diversity reward. *Determinantal Point Processes* (DPP) [12] estimates the likelihood of item sets to be diverse and relevant as the determinant of an item-item similarity kernel matrix, from which we select the top-k using a greedy selection. Finally, we include a recent embedding-based method, *Diverse GNN Recommender (DGRec)* [31], which introduces a diversity-aware aggregation mechanism into graph neural networks by selecting neighbors that maximize coverage over item categories.

c) Evaluation criteria: We evaluate model accuracy using metrics suitable for top-k recommendations, *hit rate*, *precision*, and *recall* metrics. For each user u, let $\mathcal{R}_u$ denote the top-k recommended items and $\mathcal{T}_u$ the set of ground-truth relevant items (i.e., rated above 4 in the test set). *Hit rate* measures whether at least one relevant item appears in $\mathcal{R}_u$; *precision* is the fraction of items in $\mathcal{R}_u$ that are in $\mathcal{T}_u$; and *recall* is the fraction of relevant items in $\mathcal{T}_u$ that are retrieved in $\mathcal{R}_u$. However, the above requires known ground-truth. To evaluate the relevance of unseen items, we compute the potential user satisfaction as the *relevance score*

$$\text{Relevance}(u) = \frac{1}{k} \sum_{i \in \mathcal{R}_u} \mathbb{I}[p(\tilde{\mathbf{R}}_{u,i}) > \tau], \tag{12}$$

using preference likelihoods $p(\tilde{\mathbf{R}}_{u,i}) \in [0, 1]$. We set $\mu = 3$, $\alpha = 1.5$ for the sigmoid, and a threshold $\tau = 0.8$.

A. Questions

Having introduced out setup, we now introduce our research questions.

Q1 How does the model's accuracy change when applying explicit and implicit diversity-guided optimization under different settings?

Q2 Do both explicit and implicit diversity-guided optimization methods achieve better relevance–diversity trade-offs compared to existing baselines?

Q3 Is the diversity gain confined to the optimized top-k set, or does it generalize beyond the objective?

Q4 Does diversity-guided optimization consistently converge? How does fine-tuning a relevance-only pre-trained model compare to training from scratch?

In the following, we experimentally answer all these questions in detail. We conduct all our experiments on a single Linux server with 2 AMD Epyc 7742 CPUs, 1 TB of RAM and 1 NVIDIA DGX-A100 GPU. Our code is written in Python v3.11.7. All results are averaged over 10 independent runs. We use the authors' publicly-available implementation, consider top-k recommendations for each method, and ensure a fair parametrization. We provide all information needed to reproduce our results, such as code, data, and hyperparameters.[6]

[1] https://www.kaggle.com/datasets/rishitjavia/netflix-movie-rating-dataset
[2] https://grouplens.org/datasets/movielens/1m/
[3] https://webscope.sandbox.yahoo.com
[4] https://www.cs.cornell.edu/~schnabts/mnar/
[5] https://kuairec.com/

[6] https://anonymous.4open.science/r/DDRO-B975

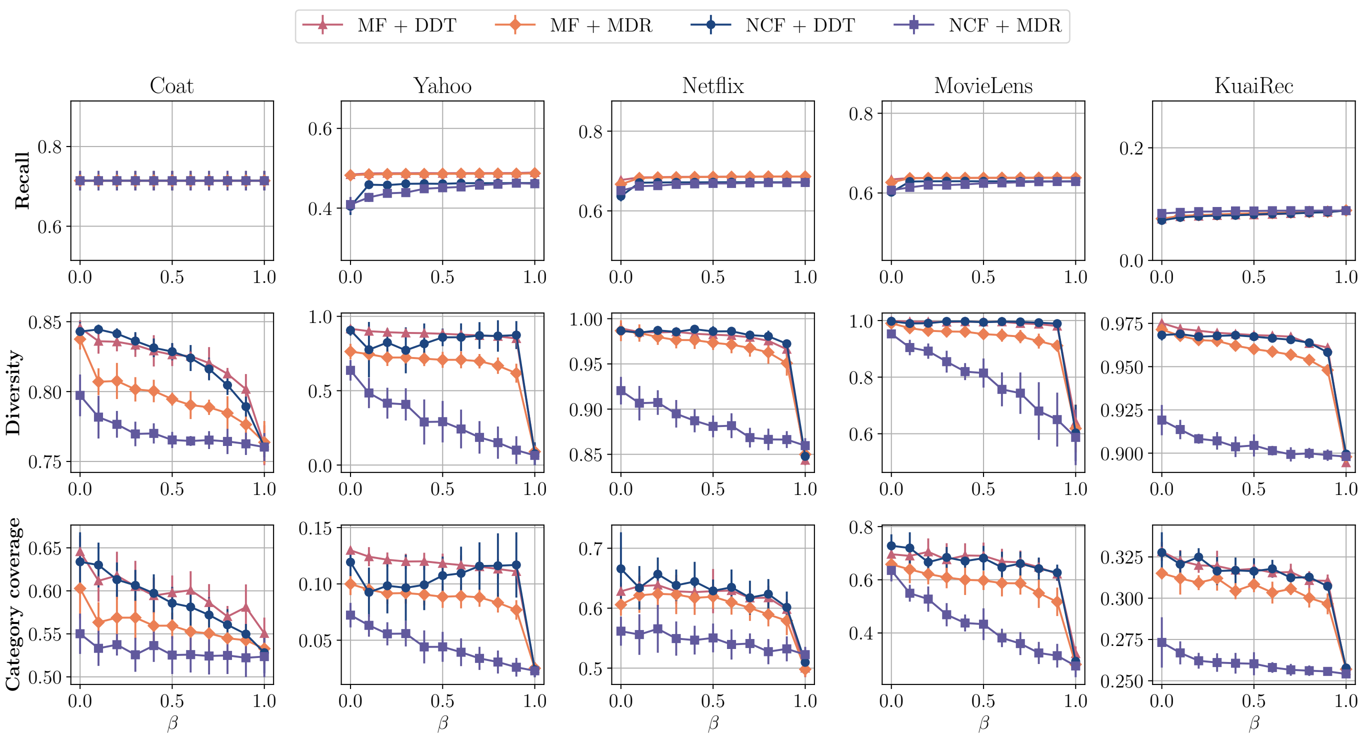

Fig. 1. Performance comparison of DDT and MDR applied to two recommender models (NMF and NCF) across five datasets: Coat, Yahoo-R2, Netflix, MovieLens, and KuaiRec. We vary the parameter $\beta \in [0, 1]$ to control the trade-off between relevance and diversity, where $\beta = 1$ corresponds to optimizing only the relevance loss $\mathcal{L}_{\mathrm{MSE}}$ and $\beta = 0$ corresponds to optimizing only the diversity loss $\mathcal{L}_{\mathrm{DDRO}}$. The x-axis indicates the value of β. The y-axis shows recall (top row), diversity reword objective (DRO) score with $k = 10$ (middle row), and category coverage (bottom row). In each setting, we initialize from a pre-trained model (using $\mathcal{L}_{\mathrm{MSE}}$ only), then fine-tune with the joint loss $\mathcal{L}_{\mathrm{JOINT}}$ for 10 epochs, selecting the best result by diversity score. All experiments are repeated 10 times, and we report the mean and standard deviation.

Q1: Impact on accuracy-diversity trade-off under varying optimization settings.

First, we study how increased diversity (lower β) influences the relevance. We show the diversity-accuracy trade-off of all combinations of tuning (DDT, MDR) and base models (MF, NCF) across all datasets. We vary β and report diversity and accuracy after 10 fine-tuning steps in Fig. 3.

Accuracy. Here, we observe a stable low-variance recall (top row) for decreasing β. This suggests that our diversity-guided objective does not compromise accuracy even under a moderate diversity fraction β. This is especially evident in Coat and KuaiRec, where the recall curves remain nearly flat. Furthermore, a slight drop in recall at $\beta < 0.3$ in Yahoo-R2, Netflix, and MovieLens can be seen, due to an exclusive focus on diversity in that parameter range, cf. Eq. (10). We see a more pronounced drop in recall for NCF, due to its higher model power. We can obtain similar trends in MSE loss, hit rate, and precision, with detailed results provided in the appendix.[7]

Diversity. Both our approaches show a considerable top-k diversity gain (DRO, middle row), highlighting the effectiveness of end-to-end optimization to increase diversity. Notably, the diversity of NMF with DDT or MDR increases sharply when β is reduced from 1.0 to 0.9 and then plateaus. While this trend is consistent in Netflix, MovieLens, and KuaiRec, Yahoo-R2 exhibits fluctuations for $\beta < 0.5$. Although DDT consistently and considerably outperforms MDR in diversity using NCF, these two perform similarly in the context of NMF, suggesting that reweighting may be beneficial for low-rank models.

Category coverage. We observe similar trends in category coverage as those seen with direct diversity. Both DDT and MDR significantly improve coverage measured across various datasets and β without severely compromising accuracy.

Q2: Diversity-relevance performance of all approaches

Next, we compare the proposed methods against alternative diversification approaches across all datasets. In Tab. III we report the mean and standard deviation of diversity and relevance scores for all methods. We highlight the best score in each column in bold and underline the second-best.

We see that our approach, DDT, demonstrates strong performance in both relevance and diversity, achieving either the best or second-best results across nearly all metrics and datasets, showing the effectiveness of jointly optimizing. Similarly, we see that MDR shows highly competitive performance, consistently ranking among the top two. Its performance is particularly notable on Netflix, MovieLens and KuaiRec,

[7] todotodotodotodo

TABLE III

Performance comparison of DDT and MDR against five alternative approaches in terms of diversity and relevance across five datasets: Coat, Yahoo-R2, Netflix, MovieLens, and KuaiRec. All experiments are repeated 10 times, and we report the mean and standard deviation of diversity and relevance scores. The diversity reward parameter is set to $k = 10$, and the trade-off parameter in the joint loss $\mathcal{L}_{\text{JOINT}}$ is fixed at $\beta = 0.2$. Each experiment is initialized from a pre-trained NMF model (optimized with $\mathcal{L}_{\text{MSE}}$ only), followed by fine-tuning with $\mathcal{L}_{\text{JOINT}}$ for 100 epochs. The best result is selected based on diversity score. The best results are marked in **bold**, and the second-best results are underlined.

Algorithm	Coat		Yahoo-R2		Netflix		MovieLens		KuaiRec	
	Diversity	Relevance	Diversity	Relevance	Diversity	Relevance	Diversity	Relevance	Diversity	Relevance
NMF	0.77 (0.02)	0.41 (0.04)	0.09 (0.05)	0.76 (0.02)	0.84 (0.01)	0.88 (0.03)	0.62 (0.08)	0.98 (0.00)	0.89 (0.01)	0.84 (0.01)
MMR	0.80 (0.01)	0.40 (0.04)	0.80 (0.06)	0.68 (0.03)	0.93 (0.01)	0.85 (0.04)	0.94 (0.03)	0.95 (0.01)	0.99 (0.00)	0.76 (0.01)
DUM	0.81 (0.01)	0.31 (0.04)	0.98 (0.02)	0.60 (0.02)	0.93 (0.00)	0.71 (0.03)	0.93 (0.01)	0.91 (0.01)	0.98 (0.00)	0.42 (0.01)
DPP	0.81 (0.01)	0.39 (0.04)	**1.00 (0.00)**	0.58 (0.01)	0.96 (0.00)	0.84 (0.04)	0.98 (0.01)	0.95 (0.01)	**1.00 (0.00)**	0.75 (0.01)
DGRec	0.71 (0.01)	**0.69 (0.02)**	0.33 (0.01)	0.83 (0.02)	0.76 (0.00)	0.83 (0.01)	0.73 (0.01)	0.47 (0.02)	0.91 (0.02)	0.18 (0.04)
DDT	**0.83 (0.01)**	0.50 (0.05)	0.98 (0.01)	**0.85 (0.01)**	**0.98 (0.00)**	**0.97 (0.01)**	**1.00 (0.00)**	**1.00 (0.00)**	0.98 (0.02)	**0.95 (0.00)**
MDR	0.82 (0.01)	0.47 (0.03)	0.86 (0.09)	0.82 (0.02)	**0.98 (0.01)**	0.93 (0.02)	0.97 (0.02)	0.99 (0.00)	0.97 (0.01)	0.85 (0.01)

TABLE IV

Diversity gain achieved by Direct Diversity Tuning (DDT) on top-$1 \sim k$ and $k+1 \sim 2k$ recommendations across five datasets: Coat, Yahoo-R2, Netflix, MovieLens, and KuaiRec. We vary the diversity reward parameter $k \in 5, 10, 20, 30, 40$ and apply DDT to two recommender models: NMF and NCF. In each setting, models are initialized from a relevance-pretrained checkpoint (using $\mathcal{L}_{\text{MSE}}$ only), then fine-tuned with the joint objective $\mathcal{L}_{\text{JOINT}}$ for 10 epochs. The best result is selected based on diversity score. Diversity gain is computed as the difference in DRO score between the fine-tuned and pre-trained models, normalized by the maximum achievable score (i.e., 1), and reported as a percentage. All experiments are repeated 10 times, and we report the mean and standard deviation.

	Dataset	**Diversity gain (%)**									
		$1 \sim k$					k+1 $\sim 2k$				
		k=5	k=10	k=20	k=30	k=40	k=5	k=10	k=20	k=30	k=40
NMF	Coat	10.9 (4.3)	5.4 (1.8)	3.5 (1.7)	2.9 (1.1)	2.6 (0.8)	-0.5 (4.7)	-2.5 (4.4)	-1.8 (1.3)	-1.5 (1.7)	-1.9 (0.8)
	Yahoo-R2	75.0 (14.3)	77.4 (8.7)	78.5 (3.8)	78.4 (2.4)	77.2 (1.6)	54.9 (11.2)	50.8 (9.9)	47.1 (2.7)	45.2 (2.1)	41.9 (1.3)
	Netflix	14.7 (2.3)	13.8 (1.0)	11.3 (0.7)	10.2 (0.6)	9.6 (0.5)	7.8 (2.6)	4.2 (1.7)	3.1 (2.1)	2.4 (2.2)	2.2 (1.8)
	MovieLens	41.4 (13.1)	37.2 (8.3)	28.4 (4.6)	23.8 (3.2)	21.6 (2.6)	25.6 (10.4)	16.2 (5.1)	10.1 (2.8)	8.3 (2.3)	6.8 (1.6)
	KuaiRec	9.2 (1.7)	7.2 (0.5)	9.4 (0.8)	10.6 (0.7)	10.9 (0.8)	5.5 (1.5)	10.3 (2.1)	6.0 (2.0)	3.3 (1.0)	1.4 (1.4)
NCF	Coat	9.7 (2.4)	1.2 (1.8)	6.9 (1.3)	0.3 (0.8)	4.8 (0.5)	-0.9 (0.6)	3.5 (0.4)	-1.1 (0.7)	2.7 (0.5)	-1.0 (0.5)
	Yahoo-R2	79.5 (12.8)	9.4 (17.5)	79.3 (7.6)	4.6 (6.0)	78.2 (8.7)	4.9 (13.7)	77.5 (4.6)	0.6 (9.0)	77.0 (6.3)	1.7 (7.3)
	Netflix	13.7 (2.0)	8.2 (1.9)	13.3 (0.9)	4.5 (1.4)	10.8 (0.7)	3.9 (1.2)	9.5 (0.5)	3.6 (1.1)	9.0 (0.5)	3.2 (1.1)
	MovieLens	46.8 (13.0)	19.6 (6.6)	36.2 (7.8)	12.7 (3.0)	26.9 (4.5)	10.9 (1.9)	22.9 (3.0)	9.6 (1.5)	20.9 (2.5)	9.0 (1.5)
	KuaiRec	8.8 (1.3)	5.4 (1.8)	6.5 (0.3)	10.7 (2.6)	8.7 (0.7)	8.0 (1.2)	10.6 (0.7)	4.2 (0.9)	11.1 (0.5)	2.3 (0.9)

where it approaches or matches the performance of DDT. This supports our earlier observation that implicit, data-driven reweighting can offer strong benefits. On the other hand, while post-hoc diversification of NMF using MMR, DUM, and DPP, considerably improve top-k diversity, they often notably reduce relevance, aligning with the results reported in previous studies [13]. For example, DPP occasionally achieves the highest diversity — particularly on Yahoo-R2 and KuaiRec— but at a high cost of low relevance. Greedy methods like MMR and DUM yield moderate diversity improvements but underperform in relevance. DGRec excels in diversity on certain datasets but suffers from severe relevance degradation, especially on KuaiRec, with a diversity of 0.91 and a relevance of 0.18. In contrast, DDT and MDR maintain a significantly better trade-off. In summary, fine-tuning and training for top-k diversity with our DDT and MDR achieve superior relevance-diversity trade-offs across the board, outperforming the competitors.

Q3: Diversity for varying k

Here, we examine how diversity changes across the growing number of recommended items. For this, we fine-tune pre-trained NMF and NCF models with DDT for 10 epochs with a fixed $\beta = 0.5$, and vary $k \in \{5, 10, 20, 30, 40\}$. The larger the top-$k$ set, the closer we are to covering all items. Thus, it gets harder to identify a truly diverse set. By increasing k, we ultimately converge to the average diversity. Therefore, we report relative diversity gains over the base model, normalized by the maximum possible diversity score. We distinguish between *in-objective* diversity gain — measured on the top-k recommendations — and *out-of-objective* diversity gain, assessed on the subsequent $k + 1 \sim 2k$ items. The former captures the direct effect of optimizing for diversity within the target set, while the latter evaluates whether this optimization yields benefits beyond the explicitly optimized range.

[ag]: Is it OK that table captions are fully in small caps? *[tz]: Small caps is not provided by the IEEE templates, it would be safe to not use small caps.*

In Tab. IV, we observe positive *in-objective* diversity gains across all datasets. In Coat, Netflix, and MovieLens, we see a decreasing gain as k increases. The improvement is especially pronounced on Yahoo-R2, with diversity gains exceeding $78.5\pm3.8\%$ in NMF and 79.3 ± 7.6 in NCF at $k = 20$, due to its low initial diversity (e.g., 0.09 in the base model). Substantial gains are also observed on MovieLens (e.g., $41.4 \pm 13.1\%$ and $46.8 \pm 13\%$ at $k = 5$) and Netflix, demonstrating the effectiveness of our method across both sparse and dense recommendation scenarios.

We see similar trends for the *out-of-objective* diversity gains, where diversity improvements generally diminish with increasing k. This effect is especially strong in Yahoo-R2 and MovieLens, meaning that our fine-tuning reshapes the item ranking in a way that benefits not only the target top-k set but also subsequent recommendations. However, the out-of-objective gains fluctuate around zero in the case of Coat—probably due to the dataset's small scale and limited item pool, where item diversity might sharply drop.

Overall, we demonstrate that our diversity-guided optimization not only consistently improves top-k in-objective diversity but also often enhances the out-of-objective diversity.

Q4: Convergence and dynamics of diversity-guided optimization

Finally, we study the empirical convergence behaviors of DDT and MDR. For this, we compare two optimization strategies: (1) training from scratch and (2) fine-tuning a relevance-pretrained model, for DDT and MDR. For each setting, we run 100 epochs and report the exact diversity (DRO) and its approximation (DDRO), depicting results in Fig. 2.

It can be seen that the approximation DDRO, closely tracks the true diversity values DDRO, confirming that the differentiable surrogate is a reliable optimization proxy. Furthermore, we see that DDT demonstrates stable and effective convergence in both training and fine-tuning modes, even consistently reaching maximum diversity. Notably, fine-tuning converges significantly faster, suggesting that, a few epochs of fine-tuning may suffice to achieve substantial diversity. For MDR, we find that performance is stable and comparable across training and fine-tuning when using the NMF model. However, optimization becomes erratic when considering the NCF, echoing earlier findings that implicit, data-driven reweighting methods may struggle to perform as good under increased model complexity. Overall, these results confirm that both training strategies are viable under DDT, while MDR may require additional care or adaptation in deep models.

B. Case study

Io better understand the capability of the diversity-guided optimization to enhance diversity while preserving recommendation relevance, we examine the recommendations generated for a representative user by two algorithms: direct diversity tuning (DDT) and non-negative matrix factorization (NMF). As shown in Tab. V, the list produced by NMF is notably homogeneous: six out of ten items are labeled as *drama*, with remaining entries only marginally extending into *romance*, *documentary*, *comedy* or *War*. In contrast, the DDT-generated

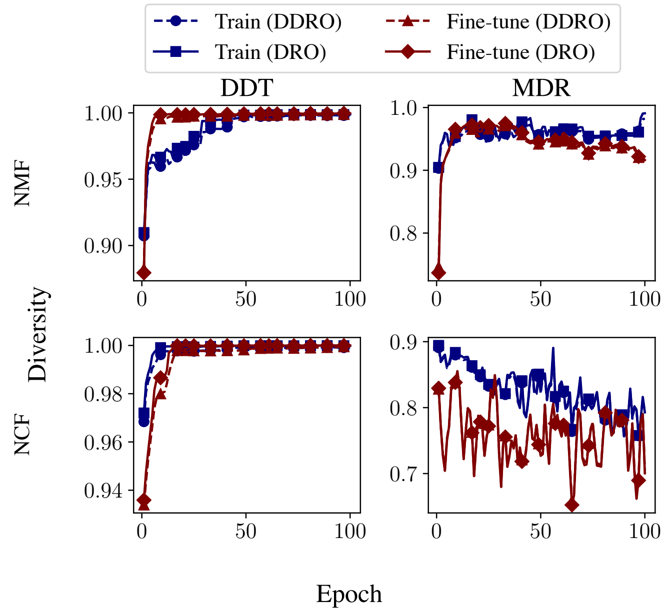

Fig. 2. Comparison of Direct Diversity Tuning (DDT) and Meta Diversity Reweighting (MDR) on the MovieLens dataset using two optimization strategies: fine-tuning and training from scratch. Each method is applied to both NMF and NCF models. In the fine-tuning setting, models are initialized from a pre-trained NMF or NCF model optimized with $\mathcal{L}_{\text{MSE}}$ only. In the training-from-scratch setting, models are initialized randomly. Results for NMF are shown in the top row, and for NCF in the bottom row. The y-axis indicates the diversity reward objective score with $k = 10$, and the x-axis denotes the number of training epochs.

list spans a much broader range of genres, including *horror*, *thriller*, *animation*, *crime*, and *adventure*, in addition to the categories already present in the NMF list.

Despite this increased diversity, DDT retains three of the top items (Mamma Roma, Smashing Time, and Gate of Heavenly Peace), which collectively represent the core genre themes of the NMF list (*drama*, *comedy*, and *documentary*, respectively). This suggests that DDT successfully preserves highly relevant content across key thematic dimensions while expanding the list to introduce more varied perspectives and experiences. Ultimately, DDT demonstrates its ability to balance relevance with diversity, offering users a richer and more engaging set of recommendations.

VII. Conclusion

In this work, we addressed the issue of limited top-k diversity of modern recommender systems, which contributes to echo chambers, reduced novelty, and social polarization. We propose an approach integrating diversity into a two-stage training process. We presented a unified framework for diversity-aware recommendation by introducing a differentiable diversity objective that enables end-to-end optimization of both relevance and diversity. We proposed two complementary, model-agnostic algorithms to support explicit and implicit integration of diversity into standard recommender systems. Extensive experiments on real-world datasets demonstrate that

TABLE V

COMPARISON OF RECOMMENDATION LISTS FOR A USER FROM TWO DIFFERENT ALGORITHMS: DDT AND NMF. EACH ROW REPORTS THE GENRE(S) AND MOVIE NAME OF THE RECOMMENDED ITEM. MOVIES APPEARING IN BOTH RECOMMENDATION LISTS ARE SHOWN IN **BOLD**.

	Rank	Genres	Movie Title
DDT	1	Horror	Vampyros Lesbos (Las Vampiras)
	2	Thriller	The Spiral Staircase
	3	War	Prisoner of the Mountains
	4	Animation, Musical	Melody Time
	5	Documentary	**The Gate of Heavenly Peace**
	6	Crime	Lured
	7	Drama	**Mamma Roma**
	8	Comedy	**Smashing Time**
	9	Adventure	Ulysses (Ulisse)
	10	Romance	Persuasion
NMF	1	Drama	**Mamma Roma**
	2	Drama	Foreign Student
	3	Drama	The Apple
	4	Drama, Romance	Leather Jacket Love Story
	5	Comedy	**Smashing Time**
	6	Documentary	**The Gate of Heavenly Peace**
	7	Documentary	Modulations
	8	Comedy, Romance, War	Forrest Gump
	9	Drama	Schlafes Bruder (Brother of Sleep)
	10	Drama, War	Schindler's List

our methods consistently improve diversity, converge efficiently, and introduce minimal computational overhead.

Limitations and future work: While our framework effectively promotes both relevance and diversity in top-k recommendations, several limitations remain. First, our findings suggest that the implicit approach (MDR) achieves performance comparable to explicit optimization, highlighting the potential of data-centric strategies for promoting diversity. This indicates that a significant portion of diversity bias may stem from the training data distribution itself, and future work could explore methods such as counterfactual data augmentation or diversity-aware sampling. Second, the trade-off between relevance and diversity is controlled by a scalar weight β in the joint loss, which currently requires empirical tuning. In some cases, diversity gains may come at a cost to accuracy, depending on the dataset characteristics. Developing adaptive or multi-objective formulations may offer a more principled way to balance these competing goals. Finally, our evaluation primarily relies on genre-based similarity to quantify diversity. While this provides interpretability, it does not capture more nuanced relationships available in rich metadata or learned semantic embeddings. Future work may extend the evaluation to more expressive and context-sensitive diversity metrics.

REFERENCES

[1] F. Ricci, L. Rokach, and B. Shapira, "Introduction to recommender systems handbook," in *Recommender systems handbook*. Springer, 2010, pp. 1–35.

[2] E. Pariser, *The filter bubble: What the Internet is hiding from you*. penguin UK, 2011.

[3] T. T. Nguyen, P.-M. Hui, F. M. Harper, L. Terveen, and 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*, 2014, pp. 677–686.

[4] A. Lambrecht and C. Tucker, "Algorithmic bias? an empirical study of apparent gender-based discrimination in the display of stem career ads," *Management science*, pp. 2966–2981, 2019.

[5] A. K. Dixit and J. W. Weibull, "Political polarization," *Proceedings of the National Academy of sciences*, pp. 7351–7356, 2007.

[6] M. Kunaver and T. Pozrl, "Diversity in recommender systems–a survey," *Knowledge-based systems*, pp. 154–162, 2017.

[7] L. Chen, G. Zhang, and E. Zhou, "Fast greedy map inference for determinantal point process to improve recommendation diversity," *Advances in Neural Information Processing Systems*, vol. 31, 2018.

[8] D. Holtz, B. Carterette, P. Chandar, Z. Nazari, H. Cramer, and S. Aral, "The engagement-diversity connection: Evidence from a field experiment on spotify," in *Proceedings of the 21st ACM Conference on Economics and Computation*, 2020, pp. 75–76.

[9] A. Anderson, L. Maystre, I. Anderson, R. Mehrotra, and M. Lalmas, "Algorithmic effects on the diversity of consumption on spotify," in *Proceedings of The Web Conference 2020*. ACM, pp. 2155–2165. [Online]. Available: https://dl.acm.org/doi/10.1145/3366423.3380281

[10] J. Carbonell and J. Goldstein, "The use of mmr, diversity-based reranking for reordering documents and producing summaries," in *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, 1998, pp. 335–336.

[11] A. Ashkan, B. Kveton, S. Berkovsky, and Z. Wen, "Optimal greedy diversity for recommendation." in *IJCAI*, vol. 15, 2015, pp. 1742–1748.

[12] L. Chen, G. Zhang, and E. Zhou, "Fast greedy map inference for determinantal point process to improve recommendation diversity," *Advances in Neural Information Processing Systems*, 2018.

[13] L. Chen, G. Zhang, and H. Zhou, "Improving the diversity of top-n recommendation via determinantal point process," in *Large Scale Recommendation Systems Workshop*, 2017, p. 3.

[14] A. Borodin, A. Jain, H. C. Lee, and Y. Ye, "Max-sum diversification, monotone submodular functions, and dynamic updates," *ACM Transactions on Algorithms (TALG)*, pp. 1–25, 2017.

[15] N. J. Hurley, "Personalised ranking with diversity," in *Proceedings of the 7th ACM Conference on Recommender Systems*, 2013, pp. 379–382.

[16] Z. Wang, Y. Shao, J. He, J. Liu, S. Xiao, T. Feng, and M. Liu, "Diversity-aware deep ranking network for recommendation," in *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, 2023, pp. 2564–2573.

[17] M. Haldar, M. Abdool, L. He, D. Davis, H. Gao, and S. Katariya, "Learning to rank diversely at airbnb," in *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, 2023, pp. 4609–4615.

[18] W. Wang, F. Feng, X. He, L. Nie, and T.-S. Chua, "Denoising implicit feedback for recommendation," in *Proceedings of the 14th ACM international conference on web search and data mining*, 2021, pp. 373–381.

[19] R. Lai, L. Chen, Y. Zhao, R. Chen, and Q. Han, "Disentangled negative sampling for collaborative filtering," in *Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining*, 2023, pp. 96–104.

[20] R. Yu, S. Liu, and X. Wang, "Dataset distillation: A comprehensive review," *IEEE transactions on pattern analysis and machine intelligence*, pp. 150–170, 2023.

[21] A. Zhang, W. Ma, X. Wang, and T.-S. Chua, "Incorporating bias-aware margins into contrastive loss for collaborative filtering," *Advances in Neural Information Processing Systems*, vol. 35, pp. 7866–7878, 2022.

[22] Y. Zheng, C. Gao, X. Li, X. He, Y. Li, and D. Jin, "Disentangling user interest and conformity for recommendation with causal embedding," in *Proceedings of the Web Conference 2021*, 2021, pp. 2980–2991.

[23] M. Zhang and N. Hurley, "Avoiding monotony: improving the diversity of recommendation lists," in *Proceedings of the 2008 ACM conference on Recommender systems*, 2008, pp. 123–130.

[24] Y. Yue and C. Guestrin, "Linear submodular bandits and their application to diversified retrieval," *Advances in Neural Information Processing Systems*, vol. 24, 2011.

[25] S. Vargas and P. Castells, "Rank and relevance in novelty and diversity metrics for recommender systems," in *Proceedings of the fifth ACM conference on Recommender systems*, 2011, pp. 109–116.

[26] S. A. Puthiya Parambath, N. Usunier, and Y. Grandvalet, "A coverage-based approach to recommendation diversity on similarity graph," in *Proceedings of the 10th ACM Conference on Recommender Systems*, 2016, pp. 15–22.

[27] A. Antikacioglu and R. Ravi, "Post processing recommender systems for diversity," in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2017, pp. 707–716.

[28] A. Ashkan, B. Kveton, S. Berkovsky, and Z. Wen, "Optimal greedy diversity for recommendation." in *IJCAI*, vol. 15, 2015, pp. 1742–1748.

[29] J. Wasilewski and N. Hurley, "Incorporating diversity in a learning to rank recommender system." in *FLAIRS*, 2016, pp. 572–578.

[30] Y. Zheng, C. Gao, L. Chen, D. Jin, and Y. Li, "Dgcn: Diversified recommendation with graph convolutional networks," in *Proceedings of the web conference 2021*, 2021, pp. 401–412.

[31] L. Yang, S. Wang, Y. Tao, J. Sun, X. Liu, P. S. Yu, and T. Wang, "Dgrec: Graph neural network for recommendation with diversified embedding generation," in *Proceedings of the sixteenth ACM international conference on web search and data mining*, 2023, pp. 661–669.

[32] R. Lai, R. Chen, and C. Zhang, "A survey on data-centric recommender systems," *arXiv preprint arXiv:2401.17878*, 2024.

[33] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Computer*, pp. 30–37, 2009.

[34] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, and T.-S. Chua, "Neural collaborative filtering," in *Proceedings of the 26th International Conference on World Wide Web*, 2017, pp. 173–182.

[35] E. Coppolillo, G. Manco, and A. Gionis, "Relevance meets diversity: A user-centric framework for knowledge exploration through recommendations," in *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2024, pp. 490–501.

[36] R. Hassin, S. Rubinstein, and A. Tamir, "Approximation algorithms for maximum dispersion," *Operations Research Letters*, pp. 133–137, 1997.

[37] M. Blondel, O. Teboul, Q. Berthet, and J. Djolonga, "Fast differentiable sorting and ranking," in *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, ser. Proceedings of Machine Learning Research. PMLR, 2020, pp. 950–959.

[38] M. Ren, W. Zeng, B. Yang, and R. Urtasun, "Learning to Reweight Examples for Robust Deep Learning," in *Proceedings of the 35th International Conference on Machine Learning*. PMLR, 2018, pp. 4334–4343.

[39] J. Bennett and S. Lanning, "The netflix prize," 2007.

[40] F. M. Harper and J. A. Konstan, "The movielens datasets: History and context," *Acm transactions on interactive intelligent systems (tiis)*, 2015.

[41] G. Dror, N. Koenigstein, Y. Koren, and M. Weimer, "The yahoo! music dataset and kdd-cup'11," in *Proceedings of KDD Cup 2011*. PMLR, 2012, pp. 3–18.

[42] T. Schnabel, A. Swaminathan, A. Singh, N. Chandak, and T. Joachims, "Recommendations as treatments: Debiasing learning and evaluation," in *international conference on machine learning*. PMLR, 2016.

[43] C. Gao, S. Li, W. Lei, J. Chen, B. Li, P. Jiang, X. He, J. Mao, and T.-S. Chua, "Kuairec: A fully-observed dataset and insights for evaluating recommender systems," in *CIKM*, 2022, p. 540–550.

[44] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *nature*, 1999.

APPENDIX

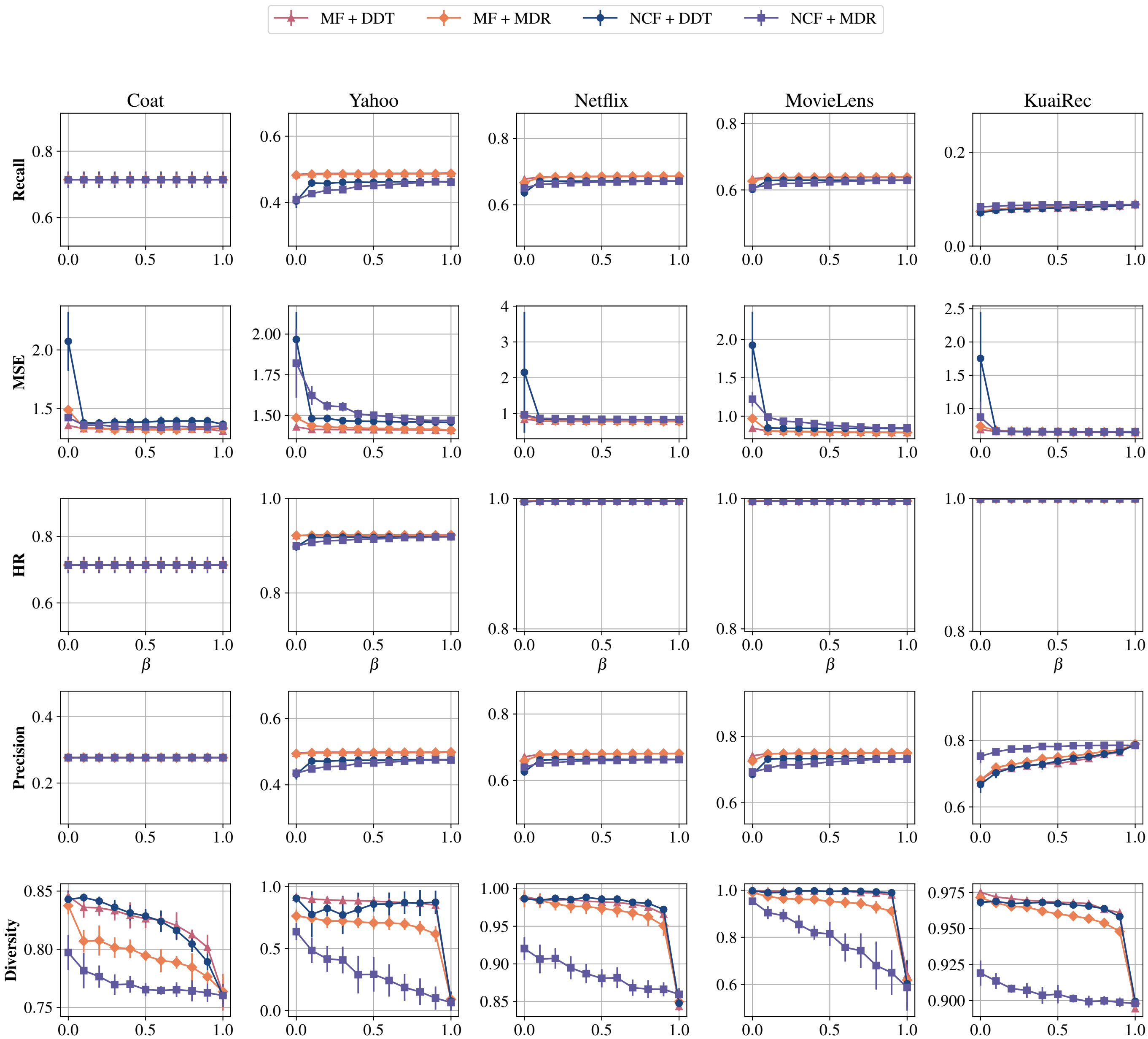

Fig. 3. Performance comparison of DDT and MDR applied to two recommender models (NMF and NCF) across five datasets: Coat, Yahoo-R2, Netflix, MovieLens, and KuaiRec. We vary the parameter $\beta \in [0, 1]$ to control the trade-off between relevance and diversity, where $\beta = 1$ corresponds to optimizing only the relevance loss $\mathcal{L}_{\text{MSE}}$ and $\beta = 0$ corresponds to optimizing only the diversity loss $\mathcal{L}_{\text{DDRO}}$. The x-axis indicates the value of β. The y-axis shows recall (top row), diversity reword objective (DRO) score with $k = 10$ (middle row), and category coverage (bottom row). In each setting, we initialize from a pre-trained model (using $\mathcal{L}_{\text{MSE}}$ only), then fine-tune with the joint loss $\mathcal{L}_{\text{JOINT}}$ for 10 epochs, selecting the best result by diversity score. All experiments are repeated 10 times, and we report the mean and standard deviation.