

Федеральное государственное автономное образовательное учреждение  
высшего образования  
«Московский физико-технический институт  
(национальный исследовательский университет)»  
Физтех-школа Прикладной Математики и Информатики  
Кафедра интеллектуальных систем

**Направление подготовки / специальность:** 03.04.01 Прикладные математика и физика

**Направленность (профиль) подготовки:** Математическая физика, компьютерные технологии и математическое моделирование в экономике

## **МНОГОКРИТЕРИАЛЬНАЯ ОПТИМИЗАЦИЯ В ЗАДАЧЕ ПЕРСОНАЛЬНЫХ РЕКОМЕНДАЦИЙ**

(магистерская диссертация)

**Студент:**

Кислинский Вадим Геннадьевич

---

*(подпись студента)*

**Научный руководитель:**

Воронцов Константин Вячеславович,  
д-р физ.-мат. наук

---

*(подпись научного руководителя)*

**Консультант (при наличии):**

---

*(подпись консультанта)*

Москва 2021

## СОДЕРЖАНИЕ

1	ВВЕДЕНИЕ . . . . .	4
2	ОБОЗНАЧЕНИЯ . . . . .	6
3	КРИТЕРИИ КАЧЕСТВА В ЗАДАЧЕ ПЕРСОНАЛЬНЫХ РЕКОМЕНДАЦИЙ . . . . .	7
4	ЗАДАЧА ПЕРСОНАЛЬНЫХ РЕКОМЕНДАЦИЙ . . . . .	12
5	МОДЕЛЬ МАТРИЧНОГО РАЗЛОЖЕНИЯ . . . . .	13
6	КЛАССИФИКАЦИЯ КРИТЕРИЕВ . . . . .	14
7	ОПТИМИЗАЦИЯ СПИСОЧНЫХ КРИТЕРИЕВ . . . . .	16
7.1	LambdaRank и релевантность . . . . .	18
7.2	LambdaRank и разнообразие . . . . .	18
7.3	LambdaRank и новизна . . . . .	19
7.4	LambdaRank и прозрачность . . . . .	19
8	ЭКСПЕРИМЕНТ . . . . .	20
8.1	Описание данных . . . . .	20
8.2	Методология эксперимента . . . . .	20
8.3	Зависимость критериев качества от веса в общей функции потерь . . . . .	21
8.4	Сравнение алгоритмов на основе матричных разложений . . . . .	24
9	ЗАКЛЮЧЕНИЕ . . . . .	26
	СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ . . . . .	27

## Аннотация

Рекомендательные системы — это системы, которые на базе алгоритмов машинного обучения предоставляют пользователям подборки релевантного контента. Оценка качества предоставленных рекомендаций является важной задачей при разработке рекомендательных систем. Для понимания того, насколько рекомендации релевантны, полезны пользователю, не являются навязчивыми или бессмысленными, удовлетворяют ли они требованиям заказчика рекомендательной системы, необходимо вводить несколько критериев, которые показывают не только точность и релевантность рекомендаций, но и разнообразие, новизну, покрытие, прозорливость. Работа посвящена разработке общего подхода к совместной оптимизации различных критериев. Вводится классификация критериев на основе правил расчета численных значений критериев. Предлагается общий подход оптимизации каждого класса критериев и вводится общий функционал качества.

**Ключевые слова:** *рекомендательные системы, коллаборативная фильтрация, матричное разложение, разнообразие, прозорливость, новизна, покрытие, совместная оптимизация, критерии качества рекомендательных систем.*

## 1 ВВЕДЕНИЕ

**Актуальность темы.** Рекомендательные системы — это неотъемлемая часть крупных онлайн-платформ, они помогают пользователям находить релевантный контент в огромном объеме информации и представляют персональный интерес. Развитие алгоритмов рекомендательных систем позволяет пользователям тратить меньше времени на поиск нужного контента. Но что делает рекомендации хорошими, полезными для пользователей?

Традиционно, исследования рекомендательных систем фокусируются на точности предсказания рейтингов объектов, которые пользователь не видел ранее. Но точность — это только один из важных критериев качества рекомендательных систем [1]. В последнее время исследователи стали фокусироваться на проблеме правильного ранжирования множества объектов, обучении ранжированию [2], также изучать проблему разнообразия списков рекомендаций [3] и новизны списка рекомендаций [4]. Все эти свойства важны для реальных рекомендательных систем, в которых пользователи взаимодействуют с небольшими списками рекомендаций из  $\text{top-}N$  объектов. В данной работе будут исследоваться критерии качества разнообразия, новизны, покрытия и прозорливости рекомендательных систем совместно с критериями точности и ранжирования.

Одним из основных подходов к генерации рекомендаций является коллаборативная фильтрация, важным подмножеством подходов коллаборативной фильтрации являются алгоритмы матричного разложения и обучения скрытых векторных представлений пользователей и объектов. Существует множество подходов на основе матричного разложения, оптимизирующих точность [5] или ранжирование [6] рекомендаций. В данной работе будут исследоваться критерии качества разнообразия, новизны, покрытия и прозорливости рекомендательных систем совместно с критериями точности и

ранжирования, подход к совместной оптимизации этих критериев при обучение скрытых векторных представлений объектов.

**Цель работы.** Формализовать и классифицировать критерии качества в рекомендательных системах, предложить подход к совместной оптимизации нескольких критериев, провести вычислительные эксперименты, в которых предложенный подход будет сравниваться с другими подходами по критериям точности, ранжирования, разнообразия, новизны, прозорливости и покрытия.

**Методы исследования.** Для достижения поставленных целей используется аппарат матричных разложений и методов обучения ранжированию. Для программной реализации разработанного алгоритма используется язык программирования Python.

**Основные положения, выносимые на защиту.**

- Предложена классификация критериев качества рекомендаций на основе вычислительных свойств.
- Предложен алгоритм совместной оптимизации критериев качества в задаче персональных рекомендаций.
- Обобщен подход к оптимизации списочных критериев качества в рекомендациях.

**Научная новизна.** Введена классификация критериев качества в задаче персональных рекомендаций на основе вычислительных свойств. Предложен алгоритм совместной оптимизации критериев качества в задаче персональных рекомендаций на основе методов матричного разложения и обучения ранжированию.

**Теоретическая и практическая значимость.** Данное исследование вносит вклад в область рекомендательных систем, позволяет строить рекомендательные системы, которые отвечают более широкому набору критериев, чем точность или ранжирование.

## 2 ОБОЗНАЧЕНИЯ

Предоставим некоторые понятия из рекомендательных систем и введем соответствующие обозначения, которые будут использоваться в работе.

- $\mathcal{U}$  — множество пользователей,
- $\mathcal{I}$  — множество объектов,
- $\mathcal{D} = \{(u_t, i_t)\}_{i=1}^T \in \mathcal{U} \times \mathcal{I}$  — транзакционные данные,
- $R = (r_{ui})_{|\mathcal{U}| \times |\mathcal{I}|}$  — матрица отношений  $r_{ui} \in \{0, 1\}$  или  $r_{ui} \in \{1, \dots, M\}$ ,
- $\mathcal{L}$  — список объектов.

### 3 КРИТЕРИИ КАЧЕСТВА В ЗАДАЧЕ ПЕРСОНАЛЬНЫХ РЕКОМЕНДАЦИЙ

**Точность** — один из основных критериев качества рекомендаций, в зависимости от постановки задачи, показывает насколько точно рекомендательная система предсказывает неизвестные рейтинги пользователей, примерами таких критериев являются: Mean squared Error (MSE), Mean Absolute Error (MAE), Cross Entropy (logloss) и другие.

**Ранжирование** — это критерий, которые показывают насколько точно рекомендательная система угадывает будущие действия пользователя с объектами рекомендательной системы и верно упорядочивает их в списке рекомендаций, примерами таких критериев являются: Precision@k, Recall@k, Mean Average Precision at K (MAP@k), Normalized Discounted Cumulative Gain at K (nDCG@k) и другие.

**Разнообразие** — критерий качества, который показывает насколько непохожими являются объекты, рекомендованные пользователю. Разнообразие списка рекомендаций  $\mathcal{L}$ ,  $|\mathcal{L}| > 0$  измеряют как среднее попарное расстояние между элементами списка.

$$Diversity(\mathcal{L}) = \frac{\sum_{i \in \mathcal{L}} \sum_{j \in \mathcal{L} \setminus \{i\}} dist(i, j)}{|\mathcal{L}|(|\mathcal{L}| - 1)} \quad (3.1)$$

Расстояние между элементами в списке  $dist(i, j)$  может рассчитываться различными способами: если объекты имеют дополнительное описание, то в качестве функции расстояния можно взять любую функцию, введенную в пространстве описаний объектов и удовлетворяющую свойствам расстояния, также расстояние между объектами можно определять на основе матрицы рейтингов или скрытых представлений объектов.

Методы диверсификации списков рекомендаций делятся на два типа: методы повторного ранжирования списков рекомендаций для увеличения

разнообразия [7, 3, 8] и методы, моделирующие разнообразие [9, 10, 11]. Методы повторного ранжирования из расширенного списка рекомендаций  $\mathcal{C}$  отбирают итоговый список объектов  $\mathcal{L}$  размера  $N$ ,  $N < |\mathcal{C}|$ . Повторное ранжирование часто используют жадную стратегию при составление списков рекомендаций: на каждой итерации элемент из  $\mathcal{C}$ , который максимизирует целевую функцию, перемещается в  $\mathcal{L}$ . Преимуществом методов повторного ранжирования является их простота применения в уже существующих рекомендательных системах. Такие методы могут использоваться вместе с другими алгоритмами рекомендательных систем, а уровень диверсификации можно явно контролировать. Методы, моделирующие разнообразие напрямую, оптимизируют разнообразие совместно с другими критериями качества. Существуют разные подходы оптимизации разнообразия при обучении: сочетание матричного разложения с портфельной теорией, регуляризационный подход, а также различные подходы к определению специальной функции потерь.

**Новизна** — критерий качества, оценивающий насколько предложенные объекты являются неизвестными, новыми для пользователя. Новизна объекта оценивается через его популярность, чем меньше оценок было выставлено объекту, тем более вероятно, что объект окажется новым для пользователя. Тогда новизну объекта  $i$  можно оценить, как  $1 - p(i)$ , где  $p(i) = \frac{|u \in \mathcal{U}, (u, i) \in \mathcal{D}|}{|\mathcal{U}|}$ . Также в качестве оценки новизны объекта берут  $-\log p(i)$ , такая оценка называется *самоинформацией* объекта  $i$  или “длинными хвостами”. Новизну списка рекомендаций можно оценить как среднее самоинформаций элементов.

$$Novelty(\mathcal{L}) = \frac{\sum_{i \in \mathcal{L}} -\log p(i)}{|\mathcal{L}|} \quad (3.2)$$

Учитывая предыдущее определение новизны, новые объекты идентифицируются как “длинные хвосты”, то есть часть каталога объектов, которую видела, оценивала или покупала небольшая часть сообщества пользователей.

Большинство подходов к повышению новизны рекомендаций основывается на продвижении редких объектов, объектов с “длинным хвостом”. Одна из первых попыток проанализировать свойства “длинного хвоста” в рекомендательных системах была представлена в работе [12]. Хотя эта работа не связана напрямую с оптимизацией новизны, она касалась повышения точности прогнозирования рейтингов для объектов с “длинным хвостом”. Они заметили, что при использовании алгоритмов прогнозирования на основе рейтингов точность прогнозирования для редких объектов ниже, чем для популярных объектов из-за меньшего количества оценок, на которых основан прогноз. Предложенное ими решение для повышения точности прогнозов было основано на кластеризации объектов с “длинным хвостом” и создании совместных рейтинговых профилей для кластеров. Затем для данного объекта с “длинным хвостом” прогноз рейтинга может быть сделан с использованием всех оценок в его кластере. Эксперименты с набором данных MovieLens показали снижение количества ошибок при использовании предложенного подхода. Однако предложенный метод не гарантировал продвижение объектов с “длинным хвостом” в списки рекомендаций пользователей.

Авторы [13] использовали информацию о популярности объектов для увеличения как новизны, так и межпользовательского разнообразия. Авторы предложили алгоритм, основанный на распределении веса в двудольном графе пользовательских объектов. Алгоритм работает путем присвоения весов объектам, оцененным целевым пользователем, а затем равного распределения веса каждого объекта среди других пользователей, которые его оценили. Затем вес каждого пользователя распределяется между его оцененными объектами. Эта процедура распределения веса отдает предпочтение узлам объектов с небольшим количеством связей графа (т.е. редким объектам), что

приводит к новым рекомендациям. В аналогичном исследовании [14] авторы описали алгоритм на основе графов с распределением веса и показали, что присвоение большего веса пользователям с небольшими профилями увеличивает как межпользовательское разнообразие, так и новизну рекомендаций. Авторы работы [4] придерживаются идеи, что новизна связана как с популярностью, так и с разнообразием рекомендаций между пользователями. Они работали с набором данных MovieLens и измеряли популярность предметов, используя кассовые сборы фильмов (применяя логарифмическую шкалу для сглаживания эффектов степенного распределения). Предложенный ими подход к оптимизации новизны пересматривает рекомендации, штрафует объекты, которые не соответствуют тенденции «личной популярности» пользователя. В работе [15] авторы предлагают оптимизировать новизну напрямую с помощью методов ранжирования, добавляя больше популярных объектов как негативные примеры для пользователей.

**Покрытие** — доля объектов, которые появляются в списках рекомендаций пользователей.

$$Coverage = \frac{|\cup_{u \in \mathcal{U}} \mathcal{L}_u|}{|\mathcal{I}|} \quad (3.3)$$

где  $\mathcal{L}_u$  — это набор всех рекомендаций, сгенерированных для пользователя  $u$ . Покрытие определяется не на уровне одного пользователя, а на уровне рекомендательной системы в целом. Подходы к повышению покрытия часто используют методы, продвигающие объекты с “длинным хвостом”, поэтому повышение покрытия тесно связано с повышением новизны рекомендаций, также в некоторых работах показывается связь с разнообразием.

**Прозорливость** — критерий качества, оценивающий способность рекомендательной системы предлагать неожиданные и релевантные объекты пользователям. Для определения прозорливости в первую очередь необходимо ввести определение неожиданной рекомендации или критерий измерения неожиданности объекта для пользователя. Один из способов измерить

неожиданность объекта для пользователя — это сравнить этот объект с объектами, которые в прошлом оценил пользователь.

$$Serendipity(\mathcal{L}, u) = \frac{1}{|\mathcal{H}_u|} \frac{1}{|\mathcal{L}|} \sum_{i \in \mathcal{L}, j \in \mathcal{H}_u} dist(i, j) rel(i, u) \quad , \quad (3.4)$$

где  $\mathcal{H}_u$  — это список объектов, с которым пользователь взаимодействовал в прошлом,  $rel(i, u)$  — релевантен объект  $i$  пользователю  $u$  или нет,  $dist(i, j)$  — расстояние между объектами  $i$  и  $j$ .

Авторы [16] одними из первых исследовали проблему интуиции рекомендательных систем. Они предложили подход к персональным рекомендациям на основе текстовых описаний объектов. На основе профиля гостя и описаний объектов был получен классификатор, предсказывающий вероятность релевантности объекта пользователю, те объекты, в которых классификатор был не уверен признавались неожиданными и включались в рекомендации. В работах [17, 18] предлагают подходы на основе графов связи объектов рекомендательной системы.

## 4 ЗАДАЧА ПЕРСОНАЛЬНЫХ РЕКОМЕНДАЦИЙ

В работе рассматривается задача персональных рекомендаций. Заданы множество пользователей  $\mathcal{U} = \{u_k\}_{k=1}^n$ , множество объектов  $\mathcal{I} = \{i_k\}_{k=1}^m$ , множество транзакций  $\mathcal{D} = \{(u, i) : u \in \mathcal{U}, i \in \mathcal{I}\}$ , матрица отношений  $R \in \mathbb{R}^{n \times m}$ , где

$$r_{ui} = \begin{cases} 1, & (u, i) \in \mathcal{D} \\ 0, & (u, i) \notin \mathcal{D}. \end{cases}$$

Каждый объект  $i \in \mathcal{I}$  имеет описание  $w_i \in \mathcal{W}$ , где  $\mathcal{W}$  – множество допустимых описаний. Задано отображение  $h : \mathcal{W} \rightarrow \mathbb{R}^d$ , где  $d$  размерность признакового пространства объектов.

Необходимо найти ранжирующую функцию:

$$a : \mathcal{U} \times \mathcal{I} \rightarrow \mathbb{R},$$

восстанавливающую правильное отношение порядка объектов для пользователя  $u$ ,  $a(u, i) < a(u, j)$ , если  $i \prec j$  для  $u$ .

Критерий формируются относительно выбранного подхода к оценке качества.

## 5 МОДЕЛЬ МАТРИЧНОГО РАЗЛОЖЕНИЯ

Модель матричного разложения — один из основных подходов к задаче персональных рекомендаций. В этой модели пользователи и объекты получают обучаемые скрытые векторные представления, на основе которых предсказывается величина, которая интерпретирует релевантность объекта пользователю.

$$\hat{r}_{ui} = p_u^T q_i, \quad (5.1)$$

где  $p_u \in \mathbb{R}^d$  — векторное представление пользователя  $u$ ,  $q_i \in \mathbb{R}^d$  — векторное представление объекта  $i$ ,  $d \ll n, d \ll m$  — размерность скрытых векторных представлений.

Существует много различных подходов обучения скрытых векторных представлений, максимизирующие разные критерии качества рекомендаций, точность [19, 20, 21, 22, 23], релевантность [6].

$$P^*, Q^* = \underset{P, Q}{\operatorname{argmin}} L(P, Q), \quad (5.2)$$

где  $L$  — функция потерь, позволяющая оптимизировать выбранный критерий.

В работе будет описан общий подход обучения скрытых векторных представлений для оптимизации различных критериев.

## 6 КЛАССИФИКАЦИЯ КРИТЕРИЕВ

**Поточечные критерии** — это критерии, оценивающие точность предсказания рейтинга, который пользователь выставит объекту (RMSE, MAE, MAPE), или точность предсказания вероятности будущего взаимодействия пользователя с объектом (logloss).

Оптимизация поточечных критериев сводится к оптимизации суммы потерь на парах пользователь-объект по заданному множеству  $\mathcal{D} = \{(u, i) : u \in \mathcal{U}, i \in \mathcal{I}\}$ .

$$P^*, Q^* = \underset{P, Q}{\operatorname{argmin}} \sum_{(u, i) \in \mathcal{D}} l(p_u^T q_i, r(u, i)), \quad (6.1)$$

где  $r(u, i)$  — рейтинг, который пользователь  $u$  выставил объекту  $i$ , или флаг наличия взаимодействия пользователя  $u$  с объектом  $i$ ,  $l$  — точечная функция потерь, например  $l(x, y) = (x - y)^2$ .

**Общие критерии** — это критерии, никак не оценивающие связь объекта с пользователем, но показывающие общие свойства рекомендательной системы (Coverage).

Общие критерии оптимизируются путем введения регуляризаторов на матрицы  $P, Q$ .

$$P^*, Q^* = \underset{P, Q}{\operatorname{argmin}} R_Q(Q) + R_P(P) \quad (6.2)$$

где  $R_Q, R_P$  — произвольные регуляризаторы.

**Списочные критерии, не зависящие от порядка** — это критерии, которые рассчитываются на основе полученных списков рекомендаций, но не зависящие от порядка объектов внутри списков (Precision@k, Diversity, Serendipity, Novelty).

**Списочные критерии, зависящие от порядка** — это критерии, которые рассчитываются на основе полученных списков рекомендаций и зависят от порядка рекомендаций (NDCG, MAP, MRR).

Списочные критерии оптимизируются методами ранжирования с помощью обучения на заданном множестве троек  $\mathcal{D}_S$ .

$$\mathcal{D}_S = \{(u, i, j) : i \succ j; i, j \in \mathcal{I}; u \in \mathcal{U}\}$$

При этом для оптимизации определенных критериев можно воспользоваться способом, предложенном в [24]. В статье описан подход оптимизации списочных критериев, зависящих от порядка, этот же подход можно обобщить для списочных критериев, не зависящих от порядка, что будет показано далее.

$$P^*, Q^* = \underset{P, Q}{\operatorname{argmin}} \sum_{(u, i, j) \in \mathcal{D}_S} h(p_u^T q_i - p_u^T q_j) \quad (6.3)$$

где  $h$  — функция потерь, например при максимизации правдоподобия  $h(x) = -\ln \sigma(x)$

Объединяя 6.1, 6.2, 6.3 можем выписать общую функцию потерь  $L$ :

$$L = \alpha \sum_{(u, i) \in \mathcal{D}} l(p_u^T q_i, r_{ui}) + \beta \sum_{(u, i, j) \in \mathcal{D}_S} h(p_u^T q_i - p_u^T q_j) + \lambda_P R_P(P) + \lambda_Q R_Q(Q), \quad (6.4)$$

где  $\alpha, \beta, \lambda_P, \lambda_Q > 0$ .

## 7 ОПТИМИЗАЦИЯ СПИСОЧНЫХ КРИТЕРИЕВ

В работах [24] описывают подход, который позволяет перейти от максимизации правдоподобия к оптимизации списочных критериев, зависящих от порядка, путем модификации правил обновления параметров модели. Введем  $MLE(\mathcal{D}_S, P, Q)$  – функция потерь на множестве  $\mathcal{D}_S$ , возникающая при максимизации правдоподобия, и  $mle(u, i, j)$  – потери на тройке  $(u, i, j)$ , где  $i$  более релевантный объект по сравнению с объектом  $j$  для пользователя  $u$ .

$$MLE(\mathcal{D}_S, P, Q) = \sum_{(u,i,j) \in \mathcal{D}_S} -\ln \frac{1}{1 + e^{-\sigma(s_i - s_j)}} \quad (7.1)$$

$$mle(u, i, j) = -\ln \frac{1}{1 + e^{-\sigma(s_i - s_j)}} \quad (7.2)$$

где  $s_i = p_u^T q_i$ ,  $s_j = p_u^T q_j$ .

Заметим, что частные производные от  $mle$  по  $s_i$  отличаются только знаком

$$\frac{\partial mle(u, i, j)}{\partial s_i} = \frac{-\sigma}{1 + e^{\sigma(s_i - s_j)}} = -\frac{\partial mle(u, i, j)}{\partial s_j}.$$

Тогда частную производную от  $mle$  по параметру  $w_k$  можно представить в виде 7.3.

$$\frac{\partial mle(u, i, j)}{\partial w_k} = \frac{\partial mle(u, i, j)}{\partial s_i} \left( \frac{\partial s_i}{\partial w_k} - \frac{\partial s_j}{\partial w_k} \right) \quad (7.3)$$

Формулу 7.3 можно переписать в виде 7.4.

$$\frac{\partial mle(u, i, j)}{\partial w_k} = \lambda_{ij} \left( \frac{\partial s_i}{\partial w_k} - \frac{\partial s_j}{\partial w_k} \right), \quad (7.4)$$

где  $\lambda_{ij}$  – не зависит от  $w_k$ .

В работе [24] показано, что  $\lambda_{ij}$  можно видоизменить так, что обновляя параметры по новым градиентам будет оптимизироваться определенный критерий ранжирования, для этого достаточно домножить  $\lambda_{ij}$  на изменение заданного критерия при перестановке объектов  $i$  и  $j$ , переопределим  $\lambda_{ij}$

$$\lambda_{ij} = \frac{-\sigma}{1 + e^{\sigma(s_i - s_j)}} \Delta_{ij}, \quad (7.5)$$

где  $\Delta_{ij}$  – изменение критерия при перестановке объектов  $i, j$ .

Выпишем правила обновления матриц  $P, Q$  при стохастическом градиентном спуске:

$$p_{uk} = p_{uk} - t \frac{-\sigma}{1 + e^{\sigma(s_i - s_j)}} (q_{ik} - q_{jk}) \Delta_{ij} \quad (7.6)$$

$$q_{ik} = q_{ik} - t \frac{-\sigma}{1 + e^{\sigma(s_i - s_j)}} p_{uk} \Delta_{ij} \quad (7.7)$$

$$q_{ik} = q_{ik} + t \frac{-\sigma}{1 + e^{\sigma(s_i - s_j)}} p_{uk} \Delta_{ij}, \quad (7.8)$$

где  $t$  — шаг градиентного спуска.

Предложенный в [24] подход к оптимизации критериев ранжирования или, по введенной выше терминологии, списочных критериев, зависящих от порядка, можно обобщить на произвольные списочные критерии. Для этого введем следующие правила вычисления  $\Delta$ -критериев:

Пусть  $\mathcal{H}_u$  – список объектов, прикрепленных к пользователю при обучении,  $rank_i$  – порядковый номер объекта  $i$  в списке  $\mathcal{H}_u$  после сортировки по значениям  $p_u^T q_i$ , тогда:

- а) для критериев, учитывающих порядок  $\Delta_{ij}$  – это разница между критериями вычисленными до и после перестановки объектов  $i, j$  в  $\mathcal{H}_u$ ;
- б) для критериев, не учитывающих порядок  $\Delta_{ij}$  – это разница между критериями вычисленными до и после перестановки объектов  $i, j$ , только критерии поочередно вычисляются на части объектов, попавших в

список  $\{k \in \mathcal{H}_u : rank_k \leq \max(\min(rank_i, rank_j), N)\}$ ,  $N$  – минимальное число рекомендаций.

Рассмотрим примеры  $\lambda$ -градиентов для оптимизации критериев релевантности, разнообразия, новизны и прозорливости.

### 7.1 LambdaRank и релевантность

Для оценки релевантности существуют множество критериев, можно взять любой из них для расчета  $\lambda$ -градиентов, в работе будет использоваться  $NDCG$ , как это предлагалось и в оригинальной статье [24].

$$\Delta_{ij}^{rel} = |(2^i - 2^j) \left( \frac{1}{\log_2(i+1)} - \frac{1}{\log_2(j+1)} \right)| \quad (7.9)$$

### 7.2 LambdaRank и разнообразие

Для оптимизации разнообразия будем использовать метрику *Diversity* определенную в 3.1. Пусть  $rank = \max(\min(rank_i, rank_j), N)$ ,  $\mathcal{H}_u(rank)$  – первые  $rank$  объектов в списке  $\mathcal{H}_u$ , отсортированном по  $p_u^T q_i$ . Тогда

$$\Delta_{ij}^{div} = \left| \frac{2}{rank(rank-1)} \sum_{k \in \mathcal{H}_u(rank) \setminus \{i,j\}} dist(i, k) - dist(j, k) \right| \quad (7.10)$$

Также для оценки и оптимизации разнообразия введем критерий *RILD* (rolling intra list distances) на основе *Diversity*:

$$RILD(\mathcal{L}, w) = \sum_{i=1}^{N-w+1} Diversity([\mathcal{L}_i, \mathcal{L}_{i+1}, \dots, \mathcal{L}_{i+w-1}]), \quad (7.11)$$

где  $w$  – окно, в котором вычисляются попарные расстояния между объектами.

При  $w = 2$  формулу 7.11 можно записать в виде:

$$RILD(L, 2) = \sum_{i=1}^{N-1} \frac{dist(\mathcal{L}_i, \mathcal{L}_{i+1})}{2}, \quad (7.12)$$

далее, если  $w$  явно не задано будем считать, что  $w = 2$ .

Пусть  $rank = \min(rank_i, rank_j)$  в списке  $\mathcal{H}_u$ ,  $k$  – объект на месте  $rank - 1$ ,  $l$  – объект на месте  $rank + 1$ , тогда

$$\Delta_{ij}^{rild} = |dist(i, k) + dist(i, l) - dist(j, k) - dist(j, l)| \quad (7.13)$$

### 7.3 LambdaRank и новизна

Для оптимизации новизны будем использовать критерий *Novelty* введенный в 3.2. Пусть  $rank = \max(\min(rank_i, rank_j), N)$ ,  $\mathcal{H}_u(rank)$  – первые  $rank$  объектов в списке  $\mathcal{H}_u$ , отсортированном по  $p_u^T q_i$ . Тогда

$$\Delta_{ij}^{nov} = \frac{|\log p(i) - \log p(j)|}{rank} \quad (7.14)$$

### 7.4 LambdaRank и прозорливость

Для оптимизации прозорливости будем использовать критерий *Serendipity* введенный в 3.4. Для расчета  $\Delta_{ij}$  критерия *Serendipity* введем дополнительный параметр  $\phi \in [0, 1] \in \mathbb{R}$ . Пусть  $l_u$  – длина списка объектов  $\mathcal{H}_u$ , из  $\mathcal{H}_u$  выберем  $\lceil \phi l_u \rceil$  случайных элементов,  $\mathcal{L}_u$  – объединение выбранных  $\lceil \phi l_u \rceil$  с парой  $\{i, j\}$ . Тогда

$$\Delta_{ij}^{ser} = \sum_{k \in \mathcal{H}_u \setminus \mathcal{L}_u} (dist(i, k) - dist(j, k)) \quad (7.15)$$

## 8 ЭКСПЕРИМЕНТ

### 8.1 Описание данных

Эксперименты проводились на наборе данных MovieLens 1M Dataset, собранных на платформе MovieLens в 2003 году. Данный набор содержит рейтинги пользователей MovieLens, которые присоединились к данной платформе в 2000 году. MovieLens 1M Dataset содержит 1 миллион рейтингов от 6000 пользователей по 4000 фильмов, также в этом наборе есть демографические данные о пользователях – пол, возраст, профессия и данные о фильмах – название, категория. Каждый рейтинг, выставленный пользователями – это натуральное число от 1 до 5. В наборе MovieLens 1M Dataset также к каждому рейтингу привязана временная метка, показывающая время его выставления.

Пользователи	6000
Фильмы	3700
Количество рейтингов	1млн

Таблица 8.1 – Набор рейтингов

фильм
название
категория

Таблица 8.2 – Описания фильмов

### 8.2 Методология эксперимента

Для настройки моделей и сравнения качества изначальный набор рейтингов 8.1 разбивался на три части относительно времени выставления рейтингов. Сначала на первых двух частях проводился подбор гиперпараметров исследуемых подходов: обучение алгоритмов проводилось на первой части, а оценка качества на второй, при этом второй датасет разбивался на несколько частей по пользователям. После чего алгоритмы с зафиксированными

гиперпараметрами обучались на первых двух частях, а финальная оценка проводилась на третьей части.

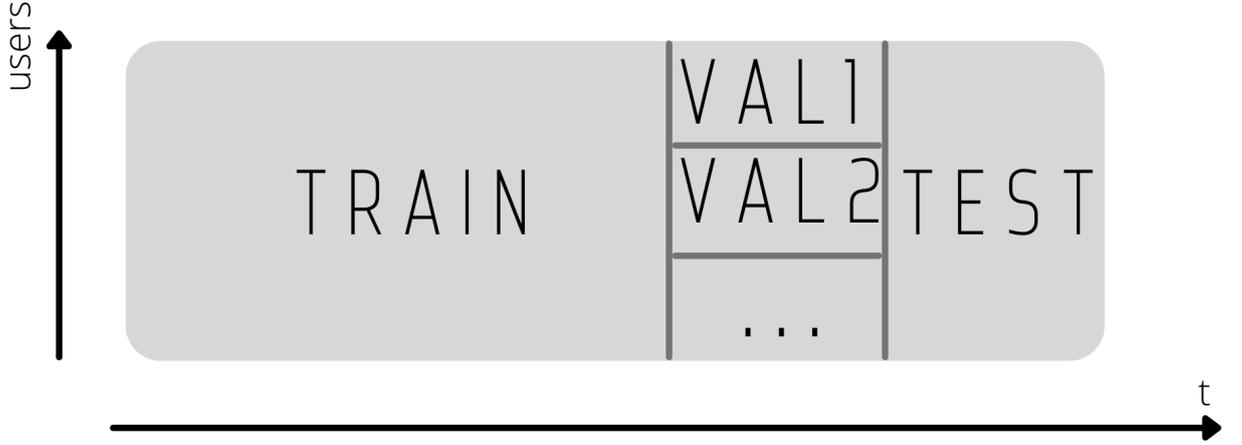


Рис. 8.1 – Стратегия валидации и оценки качества

### 8.3 Зависимость критериев качества от веса в общей функции потерь

Для анализа влияния оптимизации различных критериев на итоговый вид и качество рекомендаций, проводился эксперимент с функцией потерь 8.1:

$$L = \alpha \sum_{(u,i) \in \mathcal{D}} (p_u^T q_i - r_{ui})^2 + \sum_{(u,i,j) \in \mathcal{D}_s} h(s_{ui} - s_{uj}, \beta_{rel}, \beta_{rild}, \beta_{nov}, \beta_{ser}) + \lambda(\|P\|_2^2 + \|Q\|_2^2), \quad (8.1)$$

$$\text{где } \frac{\partial h}{\partial w_k} = - \frac{\left( \frac{\partial s_{ui}}{\partial w_k} - \frac{\partial s_{uj}}{\partial w_k} \right)}{1 + e^{s_{ui} - s_{uj}}} (\beta_{rel} \Delta_{ij}^{rel} + \beta_{div} \Delta_{ij}^{rild} + \beta_{nov} \Delta_{ij}^{nov} + \beta_{ser} \Delta_{ij}^{ser}),$$

$$s_{ui} = p_u^T q_i, \quad s_{uj} = p_u^T q_j.$$

После подбора оптимальных весов  $\alpha$ ,  $\beta_{rel}$ ,  $\beta_{rild}$ ,  $\beta_{nov}$ ,  $\beta_{ser}$ ,  $\lambda$ , в экспериментах каждый вес изменялся при фиксированных остальных. Шаг изменения веса подбирался в соответствии с численным масштабом, изменения

веса проводились как в большую, так и в меньшую сторону. После чего снимались численные показатели критериев качества рекомендаций.

В качестве критерия ранжирования по релевантности был выбран критерий  $MAP@K$  (Mean Average Precision at K):

$$MAP@K = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \frac{1}{K} \sum_{k=1}^K \frac{\sum_{i=1}^k rel(\mathcal{L}_i^u, u)}{k} rel(\mathcal{L}_k^u, u), \quad (8.2)$$

где  $\mathcal{L}^u$  – список рекомендаций для пользователя  $u$ , отсортированный в порядке убывания по  $p_u^T q_i$ ,  $\mathcal{L}_i^u$ ,  $\mathcal{L}_k^u$  – соответственно  $i$ -й и  $k$ -й элементы списка.

Для оценки качества разнообразия использовался критерий  $MDIV@K$  (Mean Diversity at K):

$$MDIV@K = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \frac{1}{|\mathcal{L}^u|(|\mathcal{L}^u| - 1)} \sum_{i \in \mathcal{L}^u} \sum_{j \in \mathcal{L}^u \setminus \{i\}} dist(i, j), \quad (8.3)$$

где  $dist(i, j)$  – расстояние между объектами  $i, j$ .

Для оценки новизны рекомендаций использовался критерий  $MNOV@K$  (Mean Novelty at K):

$$MNOV@K = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \frac{1}{|\mathcal{L}^u|} \sum_{i \in \mathcal{L}^u} -\log p(i), \quad p(i) = \frac{|u \in \mathcal{U}, (u, i) \in \mathcal{D}|}{|\mathcal{U}|}. \quad (8.4)$$

Чтобы оценить свойство рекомендательной системы выдавать пользователю неожиданные рекомендации, но при этом релевантные, то есть оценить качество прозорливости рекомендательной системы был выбран критерий  $MSER@K$  (Mean Serendipity at K):

$$MSER@K = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \frac{1}{|\mathcal{H}_u|} \frac{1}{|\mathcal{L}^u|} \sum_{i \in \mathcal{L}^u} \sum_{j \in \mathcal{H}_u} dist(i, j) rel(i, u), \quad (8.5)$$

где  $\mathcal{H}_u$  – известные транзакции пользователя.

На графике 8.2 показана зависимость критериев качества от весов  $\alpha$ ,  $\beta_{rel}$ ,  $\beta_{rild}$ ,  $\beta_{nov}$ ,  $\beta_{ser}$ .

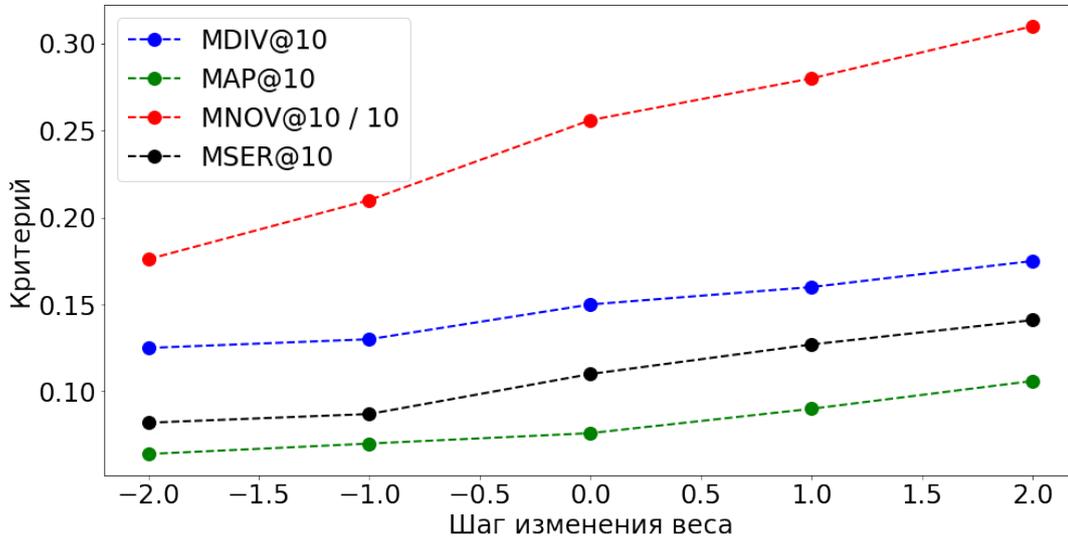


Рис. 8.2 – Зависимость качества от веса критерия в общей функции потерь

Из графика 8.2 видно, что все критерии растут при увеличении соответствующих весов. Таким образом предложенный подход позволяет увеличивать качество рекомендаций по более важным критериям или уменьшать по менее значимым, при этом этим можно явно управлять.

На графике 8.3 показана зависимость критериев качества разнообразия (MDIV@K), новизны (MNOV@K), прозорливости (MSER@K) от критерия ранжирования по релевантности (MAP@K).

Из графика 8.3 можно сделать вывод, что оптимизация критериев разнообразия, новизны, прозорливости приводит к уменьшению релевантности рекомендаций и наоборот оптимизация качества ранжирования по релевантности приводит к снижению разнообразия, новизны, прозорливости рекомендаций. Поэтому необходимо искать баланс между различными критериями качества.

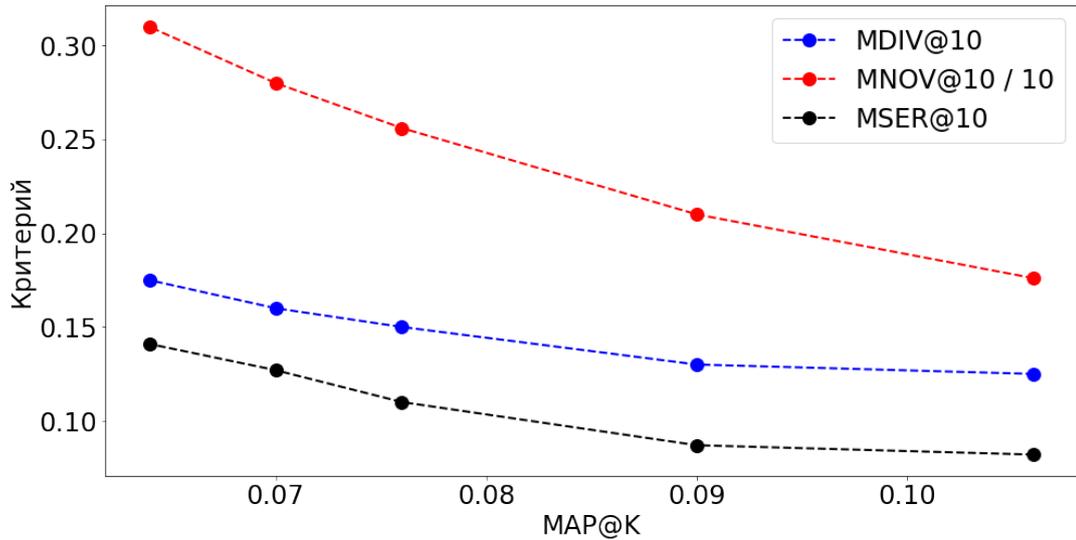


Рис. 8.3 – Зависимость MDIV@K, MNOV@K, MSER@K от MAP@K

#### 8.4 Сравнение алгоритмов на основе матричных разложений

Предложенный подход сравнивался с классическими алгоритмами рекомендательных систем: *ALS* [25] с неявными откликами пользователей, который является подходом на основе матричного разложения с поточечной функцией потерь 8.6 и  $l_2$ -регуляризаторами, и с подходом *BPR* [6] на основе матричного разложения, который оптимизирует критерии ранжирования 8.7.

$$L_{ALS} = \sum_{(u,i) \in \mathcal{D}} C_{ui} (r_{ui} - p_u^T q_i)^2 + \lambda \|P\|_2^2 + \lambda \|Q\|_2^2 \quad (8.6)$$

$$L_{BPR} = \sum_{(u,i,j) \in \mathcal{D}_s} -\ln(\sigma(p_u^T q_i - p_u^T q_j)) + \lambda (\|P\|_2^2 + \|Q\|_2^2) \quad (8.7)$$

В таблице 8.3 представлены результаты сравнения алгоритмов *ALS*, *BPR* и предложенного подхода 8.1 по критериям качества:  $MAP@K$ ,  $MDIV@K$ ,  $MNOV@K$ ,  $MSER@K$ ,  $COV@K$ , где  $COV@K$  вычисляется по формуле 8.8.

$$COV@K = \frac{|\cup_{u \in \mathcal{U}} \{\mathcal{L}_k^u : k = \overline{1, K}\}|}{|\mathcal{I}|} \quad (8.8)$$

Алгоритм	MAP	MDIV	MNOV	MSER	COV
ALS	0.097	0.12	2.12	0.10	0.45
BPR	<b>0.15</b>	0.13	1.67	0.087	0.20
Proposed method	0.076	<b>0.15</b>	<b>2.56</b>	<b>0.11</b>	<b>0.52</b>

Таблица 8.3 – Результаты экспериментов на датасете Movielens-1M,  $K = 10$

При оценки качества рекомендательной системы по нескольким разносторонним критериям можно сделать более точные выводы о работе рекомендательной системы. Например, алгоритм *BPR*, который показывает наиболее высокую оценку качества ранжирования, раскрывает лишь 20% каталога фильмов и рекомендует менее редкие фильмы. Предложенный подход показывает более высокие оценки качества по разнообразию, новизне, прозорливости и покрытию, что говорит о целесообразности его использования при необходимости повышать эти критерии качества совместно с критериями ранжирования и точности.

## 9 ЗАКЛЮЧЕНИЕ

В рамках работы был проведен анализ и составлен обзор основных критериев качества в задаче персональных рекомендаций. Была предложена классификация критериев на основе их вычислительных свойств для вывода общего подхода к совместной оптимизации набора нескольких критериев качества рекомендаций. Предложен алгоритм совместной оптимизации критериев качества в задаче персональных рекомендаций на базе алгоритмов матричного разложения, объединяющий поточечные методы и методы обучения ранжированию. Обобщен подход к оптимизации списочных критериев качества в рекомендациях. Были проведены вычислительные эксперименты, подтверждающие эффективность использования предложенного подхода и важность использования разных критериев для оценки качества рекомендательной системы.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- [1] Sean M. McNee, John Riedl и Joseph A. Konstan. “Being Accurate is Not Enough: How Accuracy Metrics Have Hurt Recommender Systems”. В: (2006).
- [2] Yue Shi, Martha Larson и Alan Hanjalic. “List-Wise Learning to Rank with Matrix Factorization for Collaborative Filtering”. В: (2010).
- [3] Saúl Vargas и Pablo Castells. “Improving Sales Diversity by Recommending Users to Items”. В: (2014).
- [4] Jinho Oh и др. “Novel Recommendation Based on Personal Popularity Tendency”. В: (2011).
- [5] Yehuda Koren, Robert Bell и Chris Volinsky. “Matrix Factorization Techniques for Recommender Systems”. В: (2009).
- [6] Steffen Rendle и др. “BPR: Bayesian Personalized Ranking from Implicit Feedback”. В: (2009).
- [7] Saúl Vargas и Pablo Castells. “Rank and Relevance in Novelty and Diversity Metrics for Recommender Systems”. В: (2011).
- [8] Andrea Barraza-Urbina и др. “XPLODIV: An Exploitation-Exploration Aware Diversification Approach for Recommender Systems”. В: (2015).
- [9] Neil J. Hurley. “Personalised Ranking with Diversity”. В: (2013).
- [10] Yue Shi и др. “Adaptive Diversification of Recommendation Results via Latent Factor Portfolio”. В: (2012).
- [11] Ruilong Su и др. “Set-Oriented Personalized Ranking for Diversified Top-n Recommendation”. В: (2013).
- [12] Yoon-Joo Park и Alexander Tuzhilin. “The Long Tail of Recommender Systems and How to Leverage It”. В: (2008).

- [13] T. Zhou и др. “Solving the apparent diversity-accuracy dilemma of recommender systems”. В: (2010).
- [14] Jian-Guo Liu, Kerui Shi и Qiang Guo. “Solving the accuracy-diversity dilemma via directed random walks”. В: (2012).
- [15] Jacek Wasilewski и Neil Hurley. “Bayesian Personalized Ranking for Novelty Enhancement”. В: (2019).
- [16] Leo Iaquinta и др. “Introducing Serendipity in a Content-Based Recommender System”. В: (2008).
- [17] Kensuke Onuma, Hanghang Tong и Christos Faloutsos. “TANGENT: a novel, ‘Surprise me’, recommendation algorithm”. В: (2009).
- [18] Makoto Nakatsuji и др. “Classical Music for Rock Fans? Novel Recommendations for Expanding User Interests”. В: (2010).
- [19] Paolo Cremonesi, Yehuda Koren и Roberto Turrin. “Performance of Recommender Algorithms on Top-n Recommendation Tasks”. В: (2010).
- [20] Yifan Hu, Yehuda Koren и Chris Volinsky. “Collaborative Filtering for Implicit Feedback Datasets”. В: (2008).
- [21] Yehuda Koren. “Factorization Meets the Neighborhood: A Multifaceted Collaborative Filtering Model”. В: (2008).
- [22] Xia Ning и George Karypis. “SLIM: Sparse Linear Methods for Top-N Recommender Systems”. В: (2011).
- [23] Ruslan Salakhutdinov и Andriy Mnih. “Probabilistic Matrix Factorization”. В: (2007).
- [24] Christopher J. C. Burges. “From RankNet to LambdaRank to LambdaMART: An Overview”. В: (2010).
- [25] Yifan Hu, Yehuda Koren и Chris Volinsky. “Collaborative Filtering for Implicit Feedback Datasets”. В: (2008).