Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение высшего образования
«Московский государственный технический университет имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ «Информатика и системы управления»

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

Отчет по лабораторной работе №5
по курсу
”Проектирование Рекомендательных Систем”

Тема Гибридные рекомендательные системы

Студент Якуба Д. В.

Группа ИУ7-33М

Оценка (баллы) _____

Преподаватели Быстрицкая А.Ю.

Москва — 2023 г.

Оглавление

Введение **3**

1 Аналитический раздел **4**
1.1 Гибридные рекомендательные системы 4
1.2 TF-IDF 4
1.3 Матричная факторизация 5
1.4 Funk SVD 6

2 Конструкторский раздел **7**
2.1 MovieLens 100K Dataset 7
2.2 Архитектура гибридной рекомендательной системы 7

3 Технологический раздел **8**
3.1 Средства реализации 8
3.2 Библиотеки 8

4 Исследовательский раздел **9**
4.1 Условия исследований 9
4.2 Зависимость значения метрики RMSE алгоритмов 9
4.3 Зависимость значения метрики MAE алгоритмов 11

ЗАКЛЮЧЕНИЕ **14**

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ **15**

ВВЕДЕНИЕ

Цель работы – реализовать гибридную рекомендательную систему.

Для достижения поставленной цели потребуется:

- спроектировать гибридную рекомендательную систему;
- привести описание задействованных в системе алгоритмов;
- привести описание используемых для исследования данных;
- привести анализ эффективности работы гибридной системы.

1. Аналитический раздел

1.1 Гибридные рекомендательные системы

Гибридные рекомендательные системы сочетают в себе различные методы рекомендаций для достижения лучшей оптимизации системы, избежать некоторые ограничения или проблем, свойственных отдельным рекомендательным моделям. Идея гибридных методов заключается в том, что комбинация алгоритмов обеспечивает более точные и эффективные рекомендации, чем один алгоритм, поскольку недостатки одного алгоритма могут быть преодолены другим алгоритмом. [1]

Термин “гибридная рекомендательная система” используется для описания любой рекомендательной системы, которая объединяет несколько методов рекомендаций для получения результата. Сами гибридные системы разделяют на: монолитные, смешанные и ансамбли. Монолитные рекомендаторы берут компоненты различных рекомендаторов и реализуют новый алгоритм. Ансамбль – это несколько работающих рекомендаторов, результаты работы которых комбинируются в одну рекомендацию. Смешанный рекомендатор возвращает результат работы сразу нескольких рекомендаторов. [1]

1.2 TF-IDF

TF-IDF (Term Frequency-Inverse Document Frequency) – это статистическая мера, используемая в информационном поиске и анализе текста для оценки важности слова в документе относительно всей коллекции документов. Эта мера может быть полезной и в рекомендательных системах для оценки сходства между элементами и пользователями. [2]

TF – частота слова, отношение числа вхождений некоторого слова к общему числу слов документа, так оценивается важность слова t_i в пределах отдельного документа:

$$tf(t,d) = \frac{n_t}{\sum_k n_k}, \tag{1}$$

где n_t — число вхождений слова t в документ;

$\sum_k n_k$ — общее количество слов в данном документе.

IDF – обратная частота документа, инверсия частоты, с которой некоторое слово встречается в документах коллекции.

$$IDF(t, D) = log\frac{|D|}{|\{d_i \in D | t \in d_i\}|}, \quad (2)$$

где $|D|$ — число документов в коллекции;

$|\{d_i \in D | t \in d_i\}|$ — число документов из коллекции D, в которых встречается t (когда $n_t \neq 0$).

Данная мера может быть использована в рекомендательных системах для:

- Представления контента, такого как текстовые описания товаров, фильмов или музыкальных треков; каждый объект (например, товар) будет представлен его описанием-вектором, в котором каждое слово представлено его TF-IDF весом, что позволит понимать, какие слова играют важную роль в этом описании – выделить “тэги”;
- Определения сходства элементов и пользователя через косинусное сходство между векторами; элементы, чьи векторы более похожи на вектор пользователя, могут быть ему рекомендованы;
- Улучшения рекомендаций путем подсчета весовых коэффициентов для слов или фраз в профилях пользователей; если пользователь часто взаимодействует с элементами, содержащими определенные ключевые слова, то можно увеличить вес для этих слов в профиле пользователя;
- Модификации; TF-IDF может быть использован вместе с другими методами рекомендации, например, с коллаборативной фильтрацией, для улучшения точности и разнообразия рекомендаций.

При этом TF-IDF имеет некоторые ограничения: он не учитывает контекст слов и не способен обрабатывать синонимы. [2]

1.3 Матричная факторизация

Матричная факторизация – это класс алгоритмов коллаборативной фильтрации, используемых в рекомендательных системах. Данные алгоритмы работают путем разложения матрицы взаимодействия пользователя с объектами на произведение двух прямоугольных матриц меньшей размерности. Зачастую матричная факторизация используется для улучшения качества персонализированных рекомендаций, позволяя выявить скрытые паттерны и взаимосвязи между пользователями и товарами.[3]

Методы матричной факторизации в рекомендательных системах обладают следующими аспектами:

- Снижение размерности – уменьшение объема вычислений и увеличение эффективности;
- Скрытые факторы – данные методы предполагают, что в системе присутствуют некоторые латентные признаки, которые влияют на предпочтения пользователей и характеристики товаров;
- Эффективность работы с разреженными данными – матричная факторизация может эффективно работать с разреженными данными, заполняя недостающие значения.

1.4 Funk SVD

Funk SVD – это один из методов матричной факторизации, который был предложен Саймоном Функом и является одним из ранних подходов к коллаборативной фильтрации.

Целью обучения Funk SVD является минимизация разницы между фактическими оценками пользователей и предсказанными на основе разложения матрицы. Для оптимизации параметров разложения и нахождения оптимальных значений скрытых факторов используется градиентный спуск. [4]

Для оценки качества модели обычно используются среднеквадратичная ошибка (RMSE) и средняя абсолютная ошибка (MAE).

Funk SVD имеет также и свои ограничения – он не способен учитывать неявные обратные связи и у него отсутствует возможность холодного старта.

Прогнозируемую оценку можно рассчитать как:

$$\widetilde{R} = HW \tag{3}$$

где $\widetilde{R} \in \mathbb{R}^{users \times items}$ — матрица оценок пользователя;
$H \in \mathbb{R}^{userx \times latentfactors}$ — содержит латентные признаки пользователя;
$W \in \mathbb{R}^{latentfactors \times items}$ — скрытые признаки объекта.

В частности, прогнозируемая оценка пользователя u объекту i:

$$\widetilde{r}_{ui} = \sum_{f=0}^{nfactors} H_{u,f} W_{f,i} \tag{4}$$

2. Конструкторский раздел

2.1 Источник данных

В качестве источника данных был взят датасет, располагающийся в свободном доступе на веб-сайте Statso [5]. Набор данных содержит информацию о взаимодействии (покупках) пользователя с продуктом в интернет-магазине одежды.

2.2 Архитектура гибридной рекомендательной системы

В качестве гибридной рекомендательной системы будет использоваться система, построенная на комбинации результатов нахождения похожих объектов по описанию с использованием TF-IDF, и результатом рекомендаций FunkSVD.

3. Технологический раздел

3.1 Средства реализации

В качестве используемого был выбран язык программирования Python [6].

Данный выбор обусловлен следующими факторами:

- Большое количество исчерпывающей документации;
- Широкий выбор доступных библиотек для разработки;
- Простота синтаксиса языка и высокая скорость разработки.

При написании программного продукта использовалась среда разработки Visual Studio Code. Данный выбор обусловлен тем, что данная среда распространяется по свободной лицензии, поставляется для конечного пользователя с открытым исходным кодом, а также имеет большое число расширений, ускоряющих разработку.

3.2 Библиотеки

При анализе и обработке датасета, а также для решения поставленных задач использовались библиотеки:

- pandas;
- numpy;
- matplotlib;
- scikit-learn;
- scikit-surprise [7].

Данные библиотеки позволили полностью покрыть спектр потребностей при выполнении работы.

4. Исследовательский раздел

4.1 Условия исследований

Исследование проводилось на персональном вычислительной машине со следующими характеристиками:

- процессор Apple M1 Pro,
- операционная система Ventura 13.5.2,
- 32 Гб оперативной памяти.

Временные затраты определялись с использованием библиотеки time.

Оценки RMSE и MAE определялись внутренними средствами библиотеки scikit-surprise.

Так как рекомендации, полученные с использование TF-IDF носят дополнительный характер, их вес составляет 0.2, в то время как вес рекомендаций SVD составляет 0.8.

4.2 Зависимость значения метрики RMSE алгоритмов

На рисунке 4.1 представлен график зависимости значения метрики RMSE от значения параметра регуляризации.

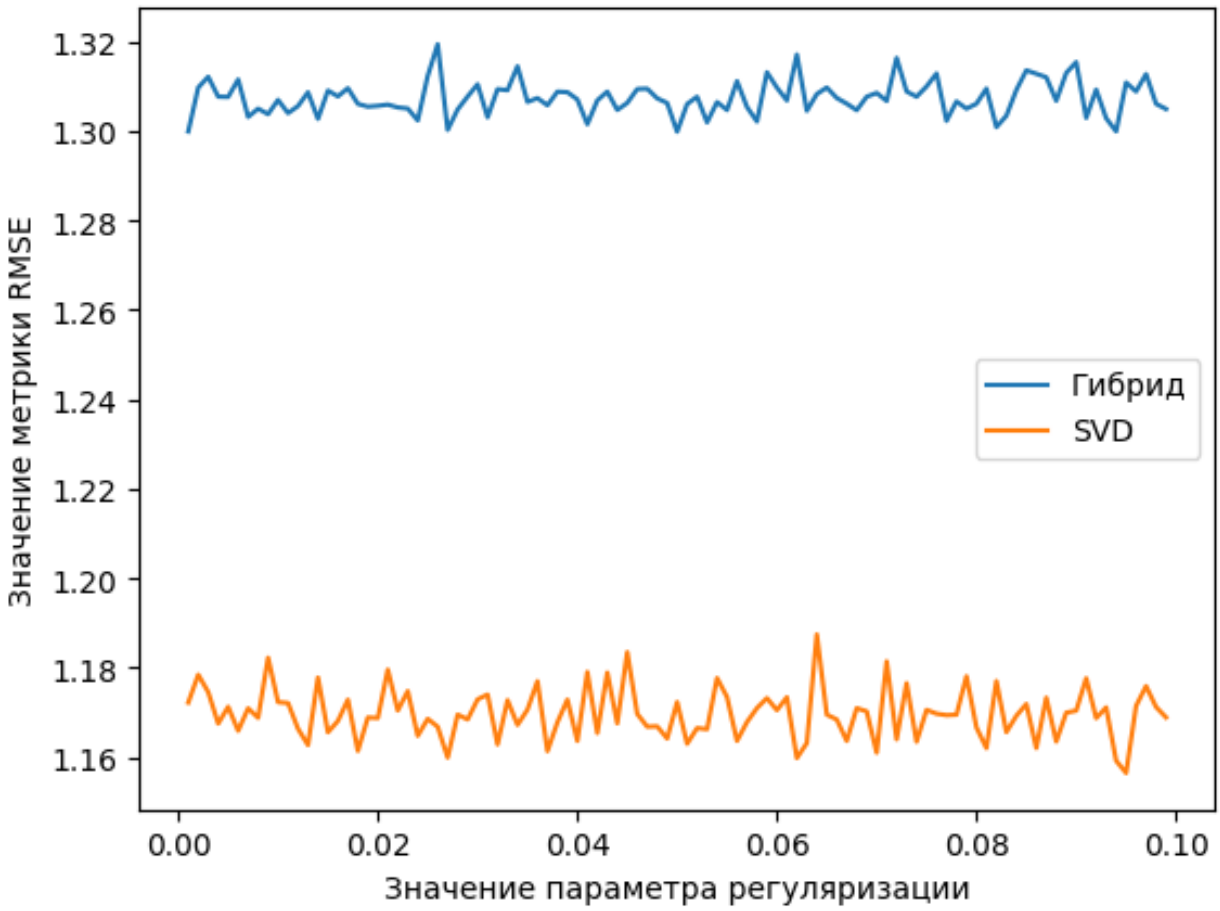

Рис. 4.1: График зависимости значения метрики RMSE от значения параметра регуляризации.

На рисунке 4.2 представлен график зависимости значения метрики RMSE от значения параметра количества эпох.

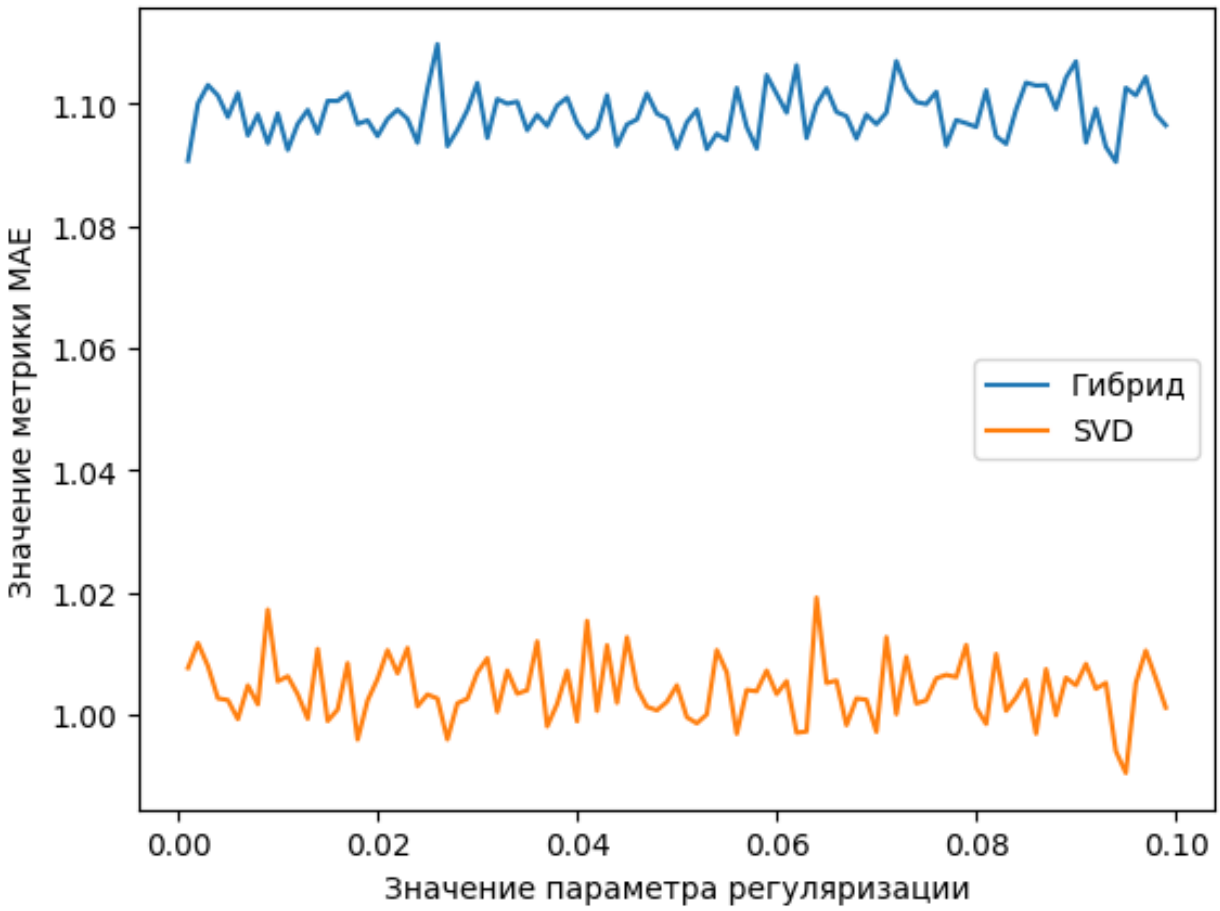

Рис. 4.2: График зависимости значения метрики RMSE от значения параметра количества эпох.

4.3 Зависимость значения метрики MAE алгоритмов

На рисунке 4.3 представлен график зависимости значения метрики MAE от значения параметра регуляризации.

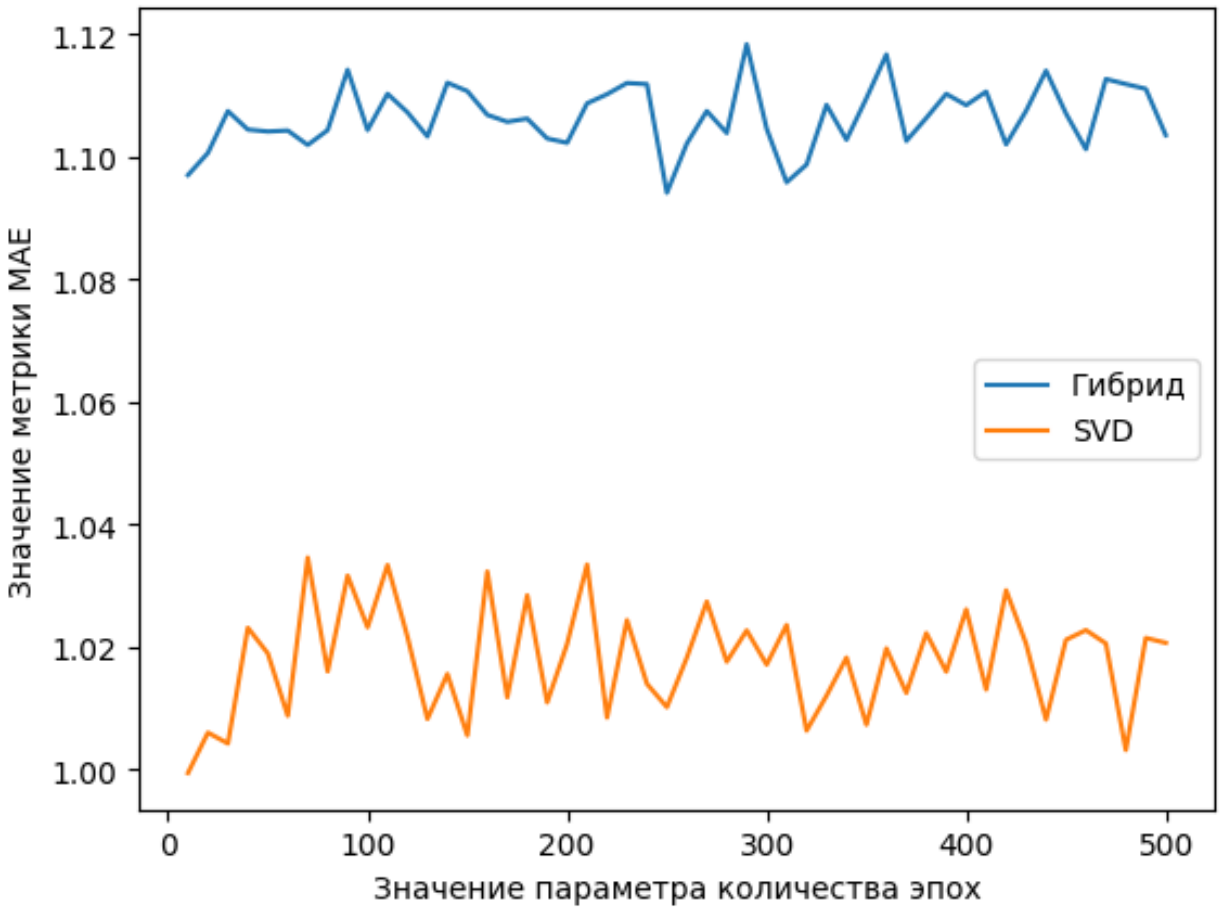

Рис. 4.3: График зависимости значения метрики MAE от значения параметра регуляризации.

На рисунке 4.4 представлен график зависимости значения метрики RMSE от значения параметра количества эпох.

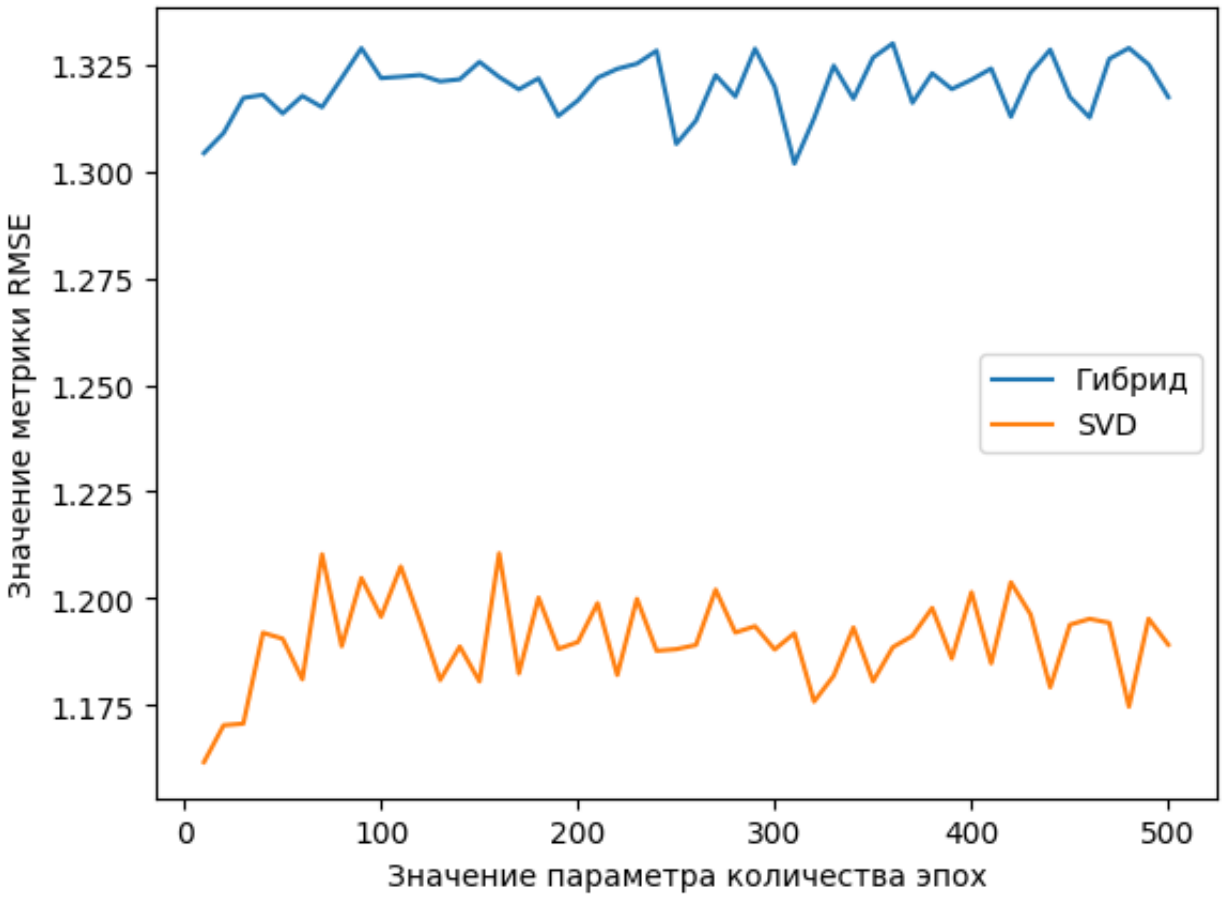

Рис. 4.4: График зависимости значения метрики MAE от значения параметра количества эпох.

Заключение

В результате проведенных исследований легко увидеть, что реализованная гибридная система уступает обычному алгоритму SVD, причем разница и по MAE, и по RMSE составляет в среднем 10 процентов. Таким образом, можно сделать вывод, что либо были подобраны неправильные веса систем, либо слишком мала выборка дополнительных рекомендаций, вносимых с использованием TF-IDF.

Так или иначе, сама реализация гибридной системы не столько удовлетворяет потребности увеличения точности, сколько большему разнообразию рекомендаций.

ЗАКЛЮЧЕНИЕ

В ходе выполнения работы была реализована гибридная рекомендательная система.

Были решены следующие задачи:

- спроектирована гибридная рекомендательная система;
- приведено описание задействованных в системе алгоритмов;
- приведено описание используемых для исследования данных;
- приведен анализ эффективности работы гибридной системы.

В результате проведенного исследования, был сделан вывод о том, что реализованная гибридная система требует либо донастройки, либо полного пересмотра архитектуры, а также исследования эффективности с использованием A/B-тестирования.

Список литературы

1. Еремин О.Ю. Моркулев Д.В. Методы реализации гибридных рекомендательных систем // E-Scio. 2023. № 3.

2. Rajaraman A. Ullman J.D. Data Mining. 2011. С. 1–17.

3. Koren Y. Bell R. Volinsky C. MATRIX FACTORIZATION TECHNIQUES FOR RECOMMENDER SYSTEMS // IEEE Computer. 2009. № 42.

4. Cornell University: An introduction to Matrix factorization and Factorization Machines in Recommendation System, and Beyond by Yuefeng Zhang [Электронный ресурс]. Режим доступа: https://arxiv.org/abs/2203.11026 (дата обращения 16.09.2023).

5. Statso: study case of user-product interactions [Электронный ресурс]. Режим доступа: https://statso.io/hybrid-recommendations-case-study/ (дата обращения 26.11.2023).

6. Python official page [Электронный ресурс]. Режим доступа: https://www.python.org/ (дата обращения 10.05.2023).

7. Scikit-surprise: official PyPl project page [Электронный ресурс]. Режим доступа: https://pypi.org/project/scikit-surprise/ (дата обращения 16.09.2023).