

# Архитектура рекомендательного сервиса для выбора направления подготовки в вузе абитуриентами с использованием метода коллаборативной фильтрации машинного обучения

Прохорова Анна Михайловна

Ст. преп. каф. информационных систем и прикладной информатики  
ORCID: 0000-0002-7228-9742, e-mail: annaprohorovam@gmail.com

Ростовский государственный экономический университет, г. Ростов-на-Дону, Россия

## Аннотация

Целью научной статьи является представление исследования архитектуры рекомендательного сервиса, разработанного для помощи абитуриентам в выборе направления подготовки в вузе. Главной функцией сервиса выступает предоставление абитуриентам персонализированных рекомендаций по подготовке на основе их предпочтений, интересов, академических достижений и рейтинга учебного заведения. Архитектура базируется на принципе клиент-серверного взаимодействия, когда клиенты могут получать персонализированные рекомендации и взаимодействовать с сервисом через веб-интерфейс. В статье были решены следующие задачи: выполнены архитектурная декомпозиция и описание основных компонентов сервиса; представлен метод машинного обучения, включая алгоритм коллаборативной фильтрации, который применяется в сервисе и позволяет учитывать предпочтения и предложения других абитуриентов с похожими интересами и образовательным профилем; разработаны рекомендации по выбору пользовательского интерфейса для удобного взаимодействия с сервисом; проведены контрольные примеры с целью оценки эффективности работы рекомендательного сервиса. Исследование показывает, что использование метода коллаборативной фильтрации в архитектуре сервиса позволяет достичь высокой точности и удовлетворения абитуриентов при предоставлении рекомендаций по выбору направления подготовки в вузе. Статья имеет практическую значимость, так как представляет собой реальное применение метода машинного обучения и архитектуры сервиса для помощи абитуриентам в выборе направления подготовки. Результаты исследования могут быть полезными для разработки подобных сервисов в образовательной сфере.

## Ключевые слова

Рекомендательный сервис, архитектура рекомендательного сервиса, выбор направления подготовки, вуз, абитуриенты, метод коллаборативной фильтрации, машинное обучение

**Для цитирования:** Прохорова А.М. Архитектура рекомендательного сервиса для выбора направления подготовки в вузе абитуриентами с использованием метода коллаборативной фильтрации машинного обучения // Вестник университета. 2024. № 5. С. 212–224.



# Architecture of a recommendation service for choosing training areas at a higher education institution by applicants with the method of collaborative filtering of machine learning

Anna M. Prokhorova

Senior Lecturer at the Information Systems and Applied Informatics Department  
ORCID: 0000-0002-7228-9742, e-mail: annaprohorovam@gmail.com

Rostov State University of Economics, Rostov-on-Don, Russia

## Abstract

The purpose of the scientific article is to present a study of a recommendation service architecture designed to help applicants in choosing the training area at a higher education institution. The main function of the service is to provide applicants with personalised recommendations for training based on their preferences, interests, academic achievements and ranking of the institution. The architecture is founded on the principle of client-server interaction when clients can receive personalised recommendations and interact with the service through a web interface. The article accomplished the following objectives: the architectural decomposition and description of the main components of the service are completed; a machine learning method is presented, including a collaborative filtering algorithm that is used in the service and allows to consider preferences and offers of other applicants with similar interests and educational profile; recommendations for choosing a user interface for convenient interaction with the service have been developed; test cases have been conducted to assess the effectiveness of the recommendation service. The study shows that the use of the collaborative filtering method in the service architecture makes it possible to achieve high accuracy and satisfaction of applicants when providing recommendations on choosing the training area at a higher education institution. The article has practical significance, as it represents a real application of the machine learning method and service architecture to help applicants choose the field of study. The research results may be useful for the development of similar services in the educational field.

## Keywords

Recommendation service, recommendation service architecture, choice of training area, higher education institution, applicants, collaborative filtering method, machine learning

**For citation:** Prokhorova A.M. (2024) Architecture of a recommendation service for choosing training areas at a higher education institution by applicants with the method of collaborative filtering of machine learning. *Vestnik universiteta*, no. 5, pp. 212–224.



## ВВЕДЕНИЕ

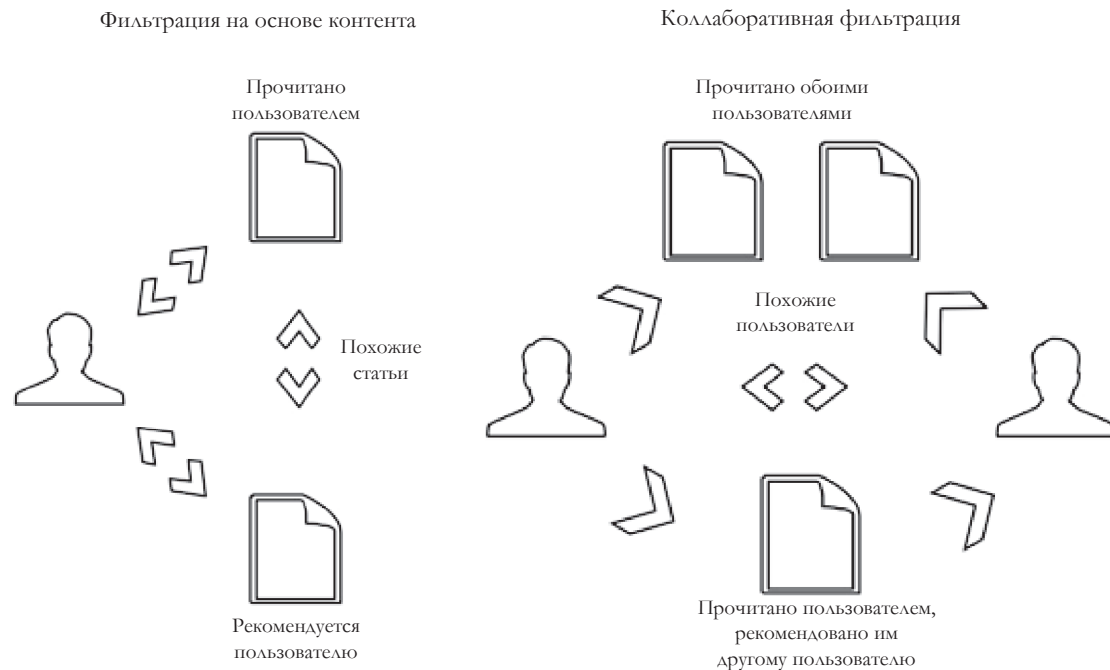
В настоящее время возникла актуальная проблема, связанная с необходимостью создания рекомендательного сервиса для абитуриентов, помогающего им выбрать подходящее направление обучения в вузе. Кроме того, множество абитуриентов сталкиваются с трудностями при выборе подготовки из-за большого количества специальностей и специализаций, предлагаемых вузами. Рекомендательный сервис позволит будущим студентам получить информацию о различных направлениях и профилях подготовки, об их особенностях и требуемых квалификациях (компетенциях), что поможет им принять более обоснованное решение [1; 2].

В некоторых научных исследованиях, посвященных профориентации в вузах с использованием инструментов машинного обучения [3–9], были выявлены следующие существенные недостатки. Во-первых, отсутствие достаточного количества качественных и полных данных для анализа эффективности профориентационных мероприятий может привести к получению неполных или неоднозначных результатов и выводов. Во-вторых, данные статьи ограничиваются применением только одного или нескольких алгоритмов машинного обучения, не проводя сравнительный анализ других методов или подходов, так как это может снизить качество и надежность полученных результатов и рекомендаций. В-третьих, не предоставлена достаточно подробная информация о применяемом алгоритме машинного обучения и его настройках, что затрудняет воспроизводимость и оценку полученных результатов и рекомендаций. Наконец, не были учтены или сравнены другие методы профориентации или алгоритмы машинного обучения с использованными в статьях, и это не позволяет оценить превосходство выбранного алгоритма и его применимость в сравнительном контексте.

В современном мире, где количество образовательных учреждений растет в геометрической прогрессии, а выбор направления подготовки становится все сложнее, необходимо обеспечить абитуриентам максимально информативный и персонализированный подход [10]. Поэтому вузы уже не ограничиваются традиционными методами предоставления информации об образовательных программах, а все чаще обращаются к разработке рекомендательных сервисов, так как эти сервисы помогают абитуриентам определиться с выбором исходя из их личных интересов, способностей и карьерных планов [11]. Однако, чтобы эффективно продвигать образовательные услуги, необходимо также создать удобную и информативную платформу. Именно поэтому образовательные порталы становятся все более популярными площадками для продвижения образовательных услуг вузов. Благодаря им абитуриенты могут получить все необходимые сведения об учебных заведениях, их преимуществах и условиях поступления. Таким образом, разработка рекомендательного сервиса и использование образовательного портала взаимосвязаны и являются важными шагами в продвижении образовательных услуг, чтобы помочь абитуриентам сделать правильный выбор и успешно получить свое образование и построить карьеру.

Рекомендательный сервис предложит абитуриентам персонализированные рекомендации на основе их интересов, навыков и предпочтений [12]. Это сократит время, затрачиваемое на поиск информации о разных направлениях обучения, а также поможет сэкономить деньги, так как абитуриенты будут более осознанно выбирать. Кроме того, сервис позволит абитуриентам выбрать направление обучения, в котором они достигнут максимального успеха и удовлетворения [13]. Например, раньше для этого использовали простой метод – составляли список самых популярных образовательных услуг. Однако постепенно такой подход уступает место более целевым рекомендациям. Теперь абитуриенты должны получать не просто список популярных образовательных услуг, а точно подобранные образовательные предложения, которые, вероятнее всего, им понравятся [14]. На рис. 1 представлены фильтрация на основе контента и коллаборативная фильтрация для пользователя.

Правильный выбор направления обучения избавит абитуриентов от ситуации, когда они позднее осознают, что выбранный ими путь не соответствует их интересам и способностям, а также от низкой мотивации и неполноценного обучения. Рекомендательный сервис сможет уменьшить количество несоответствующих опций, предлагая наиболее подходящие направления обучения [15].



Составлено автором по материалам исследования

Рис. 1. Фильтрация на основе контента и коллаборативная фильтрация для пользователя

## ОБОСНОВАНИЕ ИСПОЛЬЗОВАНИЯ МЕТОДА КОЛЛАБОРАТИВНОЙ ФИЛЬТРАЦИИ С ЦЕЛЮ РАЗРАБОТКИ АРХИТЕКТУРЫ РЕКОМЕНДАТЕЛЬНОГО СЕРВИСА ДЛЯ ВЫБОРА НАПРАВЛЕНИЯ ПОДГОТОВКИ В ВУЗЕ АБИТУРИЕНТАМИ

Коллаборативная фильтрация является одним из наиболее популярных и эффективных методов разработки рекомендательных систем. Ее использование в рекомендательном сервисе выбора направления подготовки для абитуриентов обусловлено тем, что сервис:

- основан на определении сходства между абитуриентами и предполагает, что будущие студенты с похожими предпочтениями будут иметь и схожие интересы в выборе направления;
- использует данные, собранные от большого количества абитуриентов, чтобы построить рекомендации. Для рекомендательного сервиса выбора направления подготовки это особенно важно, так как учебные курсы и программы обучения могут быть очень разнообразными и требуют значительной базы данных для эффективной рекомендации;
- не требует знания о характеристиках образовательных курсов, а только предпочтения абитуриентов, то есть позволяет системе давать рекомендации, основанные лишь на предпочтениях, что может быть очень полезно в случаях, когда информация об образовательных курсах ограничена или недоступна;
- помогает повысить удовлетворенность абитуриентов, так как составляет персонализированные рекомендации и учитывает их предпочтения.

Фильтрация на базе содержимого, гибридные рекомендательные системы и основанные на знаниях системы рекомендаций имеют некоторые ограничения:

- фильтрация на основе содержимого ограничена доступными характеристиками и описанием направлений обучения. Возможно, отсутствует полная информация о всех возможных вариантах, что может привести к узкому и ограниченному спектру рекомендаций для абитуриентов;
- гибридные рекомендательные системы нуждаются в дополнительной работе по сочетанию и интеграции различных методов и подходов. Это может быть сложным и трудоемким процессом, требующим достаточных ресурсов и экспертизы;
- системам рекомендаций, основанным на знаниях, необходимы детальный предварительный анализ предметной области и разработка базы знаний, что потребует значительных временных и финансовых ресурсов.

В целом уникальность подхода использования метода коллаборативной фильтрации в рекомендательном сервисе выбора направления подготовки обеспечивает точные, персонализированные и удовлетворительные рекомендации для абитуриентов на основе их предпочтений и поведения [16].

Алгоритм реализации рекомендательного сервиса сводится к следующему.

Шаг 1. Сбор данных о предпочтениях абитуриентов по дальнейшему высшему образованию (оценки, отзывы, рейтинги) как по самим вузам, так и по тем образовательным услугам, которые они оказывают, об историях запросов и покупок, интересах, хобби, регионах проживания, об изучаемых предметах в школе и т.д. Такие данные необходимо представить в виде матрицы, где строки соответствуют абитуриентам, а столбцы – элементам, которые они оценили. То есть был определен и оценен список критериев, важных для будущих студентов при выборе направления подготовки и самого вуза:

- 1) уровень престижности вуза и его рейтинг;
- 2) качество образовательной программы в выбранной области;
- 3) наличие профильных факультетов или специализированных направлений;
- 4) расположение университета и наличие необходимых инфраструктурных объектов (например, библиотек, спортивных объектов, студенческого общежития);
- 5) карьерные перспективы и возможности трудоустройства после окончания университета;
- 6) уровень стипендии и возможность участия в научных исследованиях или обменах;
- 7) уровень конкурсной подготовки и необходимые вступительные испытания;
- 8) рейтинг дисциплин в вузе и т.д.

Шаг 2. Создание базы данных вузов и их характеристик (включая все приведенные выше критерии). Для исследования использовалась база данных SQLite, чтобы хранить и организовывать информацию по вузам.

Для хранения информации об оценках, выставленных студентами образовательным услугам (или о рейтинге дисциплин), можно воспользоваться файлами в формате csv, где будут находиться матрицы оценок. В табл. 1 приведен пример такой матрицы, в которой представлены оценки студентов образовательным услугам (дисциплинам) вузов (далее – ОУВ). Для удобства ОУВ и студенты пронумерованы от 1 до n.

Таблица 1

Матрица оценок, выставленных абитуриентами ОУВ

| Фамилия, имя, отчество студента | ОУВ 1 | ОУВ 2 | ОУВ 3 | ОУВ 4 | ОУВ 5 | ОУВ 6 | ОУВ 7 | ОУВ 8 | ОУВ 9 | ОУВ 10 | ... | ОУВ n |
|---------------------------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|--------|-----|-------|
| Студент 1                       | 1,00  | 0,00  | 0,00  | 0,00  | 5,00  | 2,00  | 9,00  | 5,00  | 0,00  | 3,00   | ... | 0,00  |
| Студент 2                       | 0,00  | 0,00  | 5,00  | 0,00  | 2,00  | 0,00  | 8,00  | 0,00  | 0,00  | 4,00   | ... | 7,04  |
| Студент 3                       | 3,00  | 9,00  | 3,00  | 10,00 | 5,00  | 10,00 | 0,00  | 2,00  | 4,00  | 0,00   | ... | 0,00  |
| Студент 4                       | 0,00  | 0,00  | 8,00  | 0,00  | 4,00  | 9,00  | 0,00  | 9,00  | 0,00  | 8,00   | ... | 0,01  |
| Студент 5                       | 4,00  | 8,00  | 5,00  | 3,00  | 7,00  | 10,00 | 6,00  | 5,00  | 7,00  | 8,00   | ... | 0,00  |
| Студент 6                       | 0,00  | 4,00  | 0,00  | 0,00  | 0,00  | 0,00  | 10,00 | 6,00  | 3,00  | 0,00   | ... | 0,00  |
| ...                             | ...   | ...   | ...   | ...   | ...   | ...   | ...   | ...   | ...   | ...    | ... | ...   |
| Студент n                       | 0,00  | 0,00  | 0,00  | 2,00  | 4,00  | 0,00  | 9,00  | 10,00 | 0,00  | 10,00  | ... | 0,00  |

Составлено автором по материалам исследования

Для установки этой информации можно использовать csv-файл, где имя студента будет находиться в первом столбце, ID (англ. identifier – идентификатор) образовательной услуги вуза – во втором, а оценка, поставленная студентом, – в третьем (табл. 2).

Пример csv-файла для матрицы оценок

| Имя студента | ID образовательной услуги вуза | Оценка, поставленная студентом |
|--------------|--------------------------------|--------------------------------|
| Студент 1    | 1                              | 1,0                            |
| Студент 1    | 5                              | 5,0                            |
| Студент 1    | 6                              | 2,0                            |
| ...          | ...                            | ...                            |
| Студент n    | n                              | 10,0                           |

Примечание: пример csv-файла для матрицы оценок был разработан автором

Составлено автором по материалам исследования

Для хранения рекомендаций применяется стандартная структура данных в языке программирования Python – словарь (dict). В этом словаре каждому студенту соответствует словарь его оценок различным ОУВ в формате «студент», «образовательная услуга», «оценка». Представленный ниже код является примером функции, которая осуществляет чтение файла. Пример работы функции анализа csv-файла с матрицей оценок приведен на рис. 2.

```
import csv
def ReadFile (filename = "<csv_file_educational_service>"):
    f = open (filename)
    r = csv.reader (f)
    mentions = dict()
    for line in r:
        student    = line[0]
        e_s_u = line[1]
        rate       = float(line[2])
        if not student in mentions:
            mentions[student] = dict()
        mentions[student][product] = rate
    f.close()
    return mentions
```

Примечание: программный код был разработан автором

Составлено автором по материалам исследования

Рис. 2. Пример работы функции анализа csv-файла с матрицей оценок

Данная функция позволяет эффективно обрабатывать csv-файл, создавая словарь, где каждый студент имеет свой подсловарь с оценками ОУВ.

Шаг 3. Реализация функционала для рекомендательного сервиса анализа предпочтений абитуриента с применением методов машинного обучения, таких как коллаборативная фильтрация или методы классификации. Использование библиотеки Scikit-learn для обучения и прогнозирования. Выбор подхода для разработки математической модели рекомендательного сервиса. На данный момент существуют два подхода, которые идеальны для разработки рекомендательного сервиса:

1) подход, основанный на памяти. Он базируется на поиске схожих пользователей или элементов исходя из похожих предпочтений и на использовании их оценок для предсказания новых. Часто применяются методы сходства Жаккара, коэффициента Пирсона или косинусного сходства;

2) подход, основанный на модели. Он заключается в создании математической модели, которая учитывает связь между пользователями и элементами. Самыми распространенными методами здесь являются сингулярное разложение матрицы (англ. singular value decomposition) и метод наименьших квадратов (англ. least squares method).

Шаг 4. Необходимо проверить качество разработанной модели. Для этого обычно применяют кросс-валидацию или другие метрики оценки ошибки, такие как среднеквадратичная ошибка (англ. root mean square error), точность (англ. precision) и полнота (англ. recall).

Шаг 5. После валидации модели можно использовать ее для предоставления рекомендаций абитуриенту. На основе предсказанных оценок выделяется топ- $n$  элементов, которые будут наиболее релевантными для абитуриента.

Шаг 6. Разработка пользовательского интерфейса для рекомендательного сервиса по выбору направления подготовки и вуза для абитуриентов с использованием фреймворка Flask. Создание веб-страниц, на которых абитуриенты вводят свои предпочтения и параметры. Затем сервис использует алгоритмы, чтобы предоставить рекомендации по направлению подготовки и университетам, соответствующим критериям абитуриентов.

Была составлена математическая модель для прогнозирования оценок, которые пользователь (абитуриент или студент) может выставить неопределенным элементам (дисциплинам или образовательным услугам), то есть данная модель основана на поиске похожих пользователей (абитуриентов или студентов) или элементов (образовательных услуг или дисциплин). Цели и назначения корректируются в разрезе разрабатываемой математической модели, в которой также точно указываются объекты и субъекты исследования. Предполагается, что пользователи, предпочитающие или оценивающие похожие элементы, будут иметь похожие вкусы в целом.

Далее для исследования была составлена матрица пользователей (строки) и элементов (столбцы), где каждый элемент  $(i, j)$  представляет оценку пользователя  $i$  для элемента  $j$ . Затем необходимо вычислить сходство между каждой парой пользователей и сохранить результаты в матрицу сходства. Стоит отметить, что косинусная схожесть – это мера схожести между двумя векторами в многомерном пространстве. В контексте рекомендательной системы на основе коллаборативной фильтрации каждый пользователь представлен вектором с оценками для различных элементов. Математическая формула представления схожести элементов приведена ниже:

$$\text{sim}(i, j) = \frac{\sum_k r_{i,k} \cdot r_{j,k}}{\sqrt{\sum_k r_{i,k}^2} \cdot \sqrt{\sum_k r_{j,k}^2}}, \quad (1)$$

где  $\text{sim}(i, j)$  – сходство между объектами  $i$  и  $j$ ;  $r_{i,k}$  – оценка объекта  $i$  по критерию  $k$ ;  $\sum_k$  – суммирование по всем критериям.

Таким образом, формула вычисляет сходство между объектами  $i$  и  $j$  путем нахождения отношения суммы произведений значений векторов к произведению квадратных корней от суммы квадратов значений векторов. Чем ближе значение  $\text{sim}(i, j)$  к 1, тем более схожи объекты  $i$  и  $j$ .

Также можно использовать корреляцию между пользователями  $i$  и  $j$ , которая вычисляется для определения степени связи между их оценками элементов. Корреляция может показать, двигается ли пара пользователей в одном направлении при оценке элементов. Высокая корреляция означает, что оценки пользователей  $i$  и  $j$  часто совпадают или изменяются в одном направлении, а низкая корреляция указывает на противоположные оценки. Математическая формула (2) представляет собой выражение для подсчета коэффициента сходства (similarity) между двумя объектами  $i$  и  $j$  на базе их оценок по некоторым критериям, представленных в виде матрицы  $r$ :

$$\text{sim}(i, j) = \frac{\sum_k (r_{i,k} - \bar{r}_i) \cdot (r_{j,k} - \bar{r}_j)}{\sqrt{\sum_k (r_{i,k} - \bar{r}_i)^2} \cdot \sqrt{\sum_k (r_{j,k} - \bar{r}_j)^2}}, \quad (2)$$

где  $\text{sim}(i, j)$  – коэффициент сходства (similarity) между объектами  $i$  и  $j$ , который равен частному от деления суммы всех произведений;  $\bar{r}_i$  – средняя оценка объекта  $i$  по всем критериям;  $\bar{r}_j$  – средняя оценка объекта  $j$  по всем критериям.

После того как была получена матрица сходства, ее необходимо использовать для прогнозирования рейтинга, который определенный пользователь может составить для конкретного элемента. Также стоит

использовать взвешенную сумму оценок пользователей, похожих на текущего пользователя, для получения прогноза. Прогнозируется оценка или рейтинг объекта. Для этого применяется взвешенная сумма оценок, данных пользователями-соседями и умноженных на их степень схождения. Таким образом будет получена предсказанная оценка.

В данном рекомендательном сервисе для прогнозирования оценки пользователя  $i$  для элемента  $j$  используется сходство с пользователями от 1 до  $N$ . Это означает, что алгоритм прибегает к оценкам и предпочтениям других пользователей, имеющих высокое сходство с пользователем  $i$ , для выполнения прогнозирования. Путем анализа и учета оценок похожих пользователей программа может предсказать, какую оценку пользователь  $i$  мог бы поставить элементу  $j$ , на основе оценок, которые он мог бы дать элементам, уже оцененным похожими пользователями. Формула (3) выражает вычисление значения  $p(i, j)$ , где  $p(i, j)$  представляет собой предсказание для элемента  $(i, j)$  в матрице данных:

$$p(i, j) = \bar{r}_i + \frac{\sum_{k \in N} (r_{k,j} - \bar{r}_k) \cdot \text{sim}(i, k)}{\sum_{k \in N} \text{sim}(i, k)}, \quad (3)$$

где  $p(i, j)$  – прогнозируемая оценка для элемента  $(i, j)$  матрицы;  $\bar{r}_i$  – среднее значение оценок для пользователя  $i$ ;  $r_{k,j}$  – оценка пользователя  $k$  для элемента  $j$ ;  $\text{sim}(i, k)$  – сходство между пользователями  $i$  и  $k$ ;  $N$  – множество пользователей, которые оценили элемент  $j$ .

Таким образом, формула (3) представляет собой вычисление предсказания оценки пользователя  $i$  для элемента  $j$ , основываясь на средней оценке пользователя  $i$  и сходстве между пользователем  $i$  и другими пользователями  $k$ .

## АЛГОРИТМ АНАЛИЗА ПРЕДПОЧТЕНИЙ СТУДЕНТОВ, ОСНОВАННОГО НА КОЛЛАБОРАТИВНОЙ ФИЛЬТРАЦИИ ПО ОЦЕНКЕ НАПРАВЛЕНИЙ ПОДГОТОВКИ В ВУЗАХ

Для определения совпадений вкусов и предпочтений студентов была предварительно составлена матрица предпочтений для абитуриентов. Далее применялся алгоритм коллаборативной фильтрации, который включает в себя следующие шаги.

Во-первых, необходимо найти студентов, у которых самое большое сходство во вкусах по образовательным услугам (дисциплинам) с рассматриваемыми студентами. Для этого мы вычисляем выбранную меру схождения (в данном случае – косинусную) относительно изучаемого студента и отбираем студентов с наибольшим сходством.

На рис. 3 представлено, как может быть реализовано на языке Python вычисление косинусной меры между двумя студентами на основе их векторных представлений.

Приведенный код (рис. 3) читает файл “ratings\_matrix.csv”, в котором хранится матрица оценок студентов. Затем создается векторное представление для каждого студента и вычисляется косинусное сходство между ними. Результаты выводятся на экран, но они также могут быть сохранены в файле “ratings\_matrix.csv” для дальнейшей работы над данными.

В табл. 3 представлены значения с наибольшей схожестью по образовательным услугам, которые выбрал студент 1.

```
from math import sqrt
from collections import defaultdict
from scipy.spatial import distance

# Создать векторное представление для каждого студента
ratings = defaultdict(dict)
for line in open("ratings_matrix.csv"):
    student, e_s, rating = line.strip().split(",")
    ratings[student][e_s] = float(rating)

students = list(ratings.keys())
vectors = {}
for student in students:
    vector = []
    for e_s in sorted(ratings[student]):
        vector.append(ratings[student][e_s])
    vectors[student] = vector

# Вычислить косинусное сходство между студентами
similarities = {}
for i in range(len(students)):
    for j in range(i+1, len(students)):
        student1 = students[i]
        student2 = students[j]
        similarity = 1 - distance.cosine(vectors[student1], vectors[student2])
        similarities[(student1, student2)] = similarity

# Вывести результаты
for (student1, student2), similarity in similarities.items():
    print("Similarity between", student1, "and", student2, ":", similarity)
```

Примечание: программный код был разработан автором

Составлено автором по материалам исследования

Рис. 3. Вычисление косинусной меры между двумя студентами на основе их векторных представлений на языке Python

Таблица 3

**Значения с наибольшей схожестью по выбранным образовательным услугам для студента 1**

| № студента | Студент 2 | Студент 3 | Студент 4 | ... | Студент n | Сумма |
|------------|-----------|-----------|-----------|-----|-----------|-------|
| Студент 1  | 0,004     | 0,251     | 0         | ... | ...       | 3,005 |

Составлено автором по материалам исследования

Во-вторых, для каждого студента необходимо перемножить его оценки на найденную меру, чтобы оценки студентов, более похожих между собой, имели больший вес при определении итоговой позиции ОУВ (табл. 3).

В-третьих, для каждой образовательной услуги необходимо вычислить сумму взвешенных оценок наиболее похожих студентов – Сумма, а также разделить ее на общую сумму выбранных студентов – Результат (табл. 4).

Таблица 4

**Таблица взвешенных оценок студентов**

| NN        | 1      | 2     | 3     | 4     | 5     | 6     | 7     | ... | n     |
|-----------|--------|-------|-------|-------|-------|-------|-------|-----|-------|
| Студент 2 | 0,000  | 0,204 | 0,000 | 0,294 | 3,260 | 0,000 | 0,381 | ... | 0,000 |
| Студент 3 | 3,251  | 0,892 | 0,209 | 3,719 | 0,701 | 0,026 | 0,000 | ... | 0,000 |
| Студент 4 | 0,000  | 3,539 | 2,895 | 0,000 | 6,802 | 0,903 | 0,000 | ... | 2,008 |
| ...       | ...    | ...   | ...   | ...   | ...   | ...   | ...   | ... | ...   |
| Студент n | 8,005  | 0,000 | 1,876 | 0,181 | 0,000 | 1,701 | 1,803 | ... | 4,073 |
| Сумма     | 12,256 | 5,096 | 7,085 | 4,194 | 5,961 | 2,727 | 3,074 | ... | 6,073 |
| Результат | 3,745  | 1,695 | 2,357 | 1,395 | 1,983 | 0,907 | 1,022 | ... | 2,02  |

Составлено автором по материалам исследования

## КОНТРОЛЬНЫЙ ПРИМЕР РАСЧЕТА МАТРИЦЫ ПРЕДПОЧТЕНИЙ ДЛЯ АБИТУРИЕНТОВ

В табл. 5 представлена следующая матрица предпочтений абитуриентов и их оценки (где прочерк обозначает отсутствие оценки) на основе ряда важных критериев (К) по выбору направлений подготовки в вузах, которые были определены и оценены ранее.

Таблица 5

**Матрица предпочтений пользователей**

| № абитуриента | Критерий 1 | Критерий 2 | Критерий 3 | Критерий 4 |
|---------------|------------|------------|------------|------------|
| Абитуриент 1  | 4          | –          | 3          | 5          |
| Абитуриент 2  | –          | 2          | 1          | 4          |
| Абитуриент 3  | 5          | 3          | –          | 2          |
| Абитуриент 4  | 1          | –          | 4          | –          |
| Абитуриент 5  | –          | 5          | 2          | 3          |

Составлено автором по материалам исследования

Далее вычисляются результаты и выборки по абитуриенту 1 и находятся наиболее похожие по предпочтениям абитуриенты. В табл. 6 представлено вычисление схожести абитуриентов с помощью косинусной меры.

Таблица 6

## Вычисление схожести абитуриентов с помощью косинусной меры

| № абитуриента | Абитуриент 2 | Абитуриент 3 | Абитуриент 4 | Абитуриент 5 |
|---------------|--------------|--------------|--------------|--------------|
| Абитуриент 1  | 0,416        | 0,935        | 0,321        | 0,676        |

Составлено автором по материалам исследования

Анализ выявил, что наиболее схожи по предпочтениям и наиболее значимы для абитуриента 1 абитуриент 3 и абитуриент 5. Далее умножаем оценки каждого из них на соответствующие значения косинусной меры: оценки абитуриента 3 умножаем на 0,935, оценки абитуриента 5 – на 0,676.

Суммируем калиброванные оценки выбранных абитуриентов для каждого К. Для К1:  $(50,935) + (10,676) = 5,77$ ; для К2:  $(30,935) + (50,676) = 5,707$ ; для К3:  $(20,935) + (20,676) = 3,853$ ; для К4:  $(2 \cdot 0,676) = 1,352$ .

Делим полученную сумму на сумму значений косинусной меры. Для К1:  $5,77 / (0,935 + 0,676) = 3,036$ ; для К2:  $5,707 / (0,935 + 0,676) = 2,993$ ; для К3:  $3,853 / (0,935 + 0,676) = 2,028$ ; для К4:  $1,352 / (0,935 + 0,676) = 0,712$ .

Таким образом, получаем итоговые значения (result) для каждого К, из чего мы можем сделать вывод, что К1 больше всего подходит абитуриенту 1 (табл. 7).

Таблица 7

## Значения результатов по выбранным критериям

| Критерий   | Результат |
|------------|-----------|
| Критерий 1 | 3,036     |
| Критерий 2 | 2,993     |
| Критерий 3 | 2,028     |
| Критерий 4 | 0,712     |

Составлено автором по материалам исследования

В формальной нотации этого алгоритма можно записать его математическое выражение, представленное в формуле (4):

$$r_{u,i} = k \sum_{u' \in U} sim(u, u') r_{u',i}, \quad (4)$$

где функция  $sim$  – выбранная нами мера схожести двух абитуриентов;  $U$  – множество абитуриентов;  $r$  – выставленная оценка;  $k$  – нормировочный коэффициент, который вычисляется как:

$$k = \frac{1}{\sum_{u' \in U} |sim(u, u')|}. \quad (5)$$

Для проверки исправности функционирования контрольного примера есть возможность выполнить последующую команду: `res = make_recommendation('abiturient1', ReadFile(), 2,2)`. Проверка функционирования правильной работы контрольного примера представлена в табл. 8.

Таблица 8

## Проверка функционирования корректной работы контрольного примера

| Most correlated with "abiturient1" abiturients: |                         |
|-------------------------------------------------|-------------------------|
| UsersID: abiturient2                            | Coeff: 0.416            |
| UsersID: abiturient3                            | Coeff: 0.935            |
| Most correlated products:                       |                         |
| K: 1                                            | CorrelationCoeff: 3.036 |
| K: 2                                            | CorrelationCoeff: 2.993 |

Составлено автором по материалам исследования

```

import math
def make_recommendation(abiturientID, abiturient_rates, n_best_abiturients,
n_best_K):
    matches = [(u, dist_cosine(abiturient_rates[abiturientID], abiturient_rates
[u])) for u in abiturient_rates if u != abiturientID]
    best_matches = sorted(matches, key=lambda x: (x[1], x[0]), reverse=True)
    [:n_best_abiturients]
    print("Most correlated with '%s' abiturients:" % abiturientID)
    for line in best_matches:
        print(" AbiturientID: %6s Coeff: %6.4f" % (line[0], line[1]))
    sim = dict()
    sim_all = sum([x[1] for x in best_matches])
    best_matches = {x: y for x, y in best_matches if y > 0.0}
    for related_abiturient in best_matches:
        for K in abiturient_rates[related_abiturient]:
            if K not in abiturient_rates[abiturientID]:
                if K not in sim:
                    sim[K] = 0.0
                sim[K] += abiturient_rates[related_abiturient][K] * best_matches
[related_abiturient]
    for K in sim:
        sim[K] /= sim_all
    best_K = sorted(sim.items(), key=lambda x: (x[1], x[0]), reverse=True)
    [:n_best_K]
    print("Most correlated K:")
    for prod_info in best_products:
        print(" KID: %6s CorrelationCoeff: %6.4f" % (k_info[0], k_info[1]))
    return [(x[0], x[1]) for x in best_k]

```

Примечание: программный код был разработан автором

Составлено автором по материалам исследования

Рис. 4. Программный код на Python контрольного примера

```

from sklearn.metrics.pairwise import cosine_similarity
import numpy as np
class RecommenderSystem:
    def __init__(self, data):
        self.data = data

    def preprocess_data(self):
        # Преобразование данных в матрицу абитуриент-предпочтение
        abits = list(set(data['abit_id']))
        pref = list(set(data['pref_id']))
        self.num_abits = len(abits)
        self.num_prefs = len(pref)
        self.abit_mapping = {abit: i for i, abit in enumerate(abits)}
        self.pref_mapping = {pref: i for i, pref in enumerate(pref)}
        self.abit_pref_matrix = np.zeros((self.num_abits, self.num_prefs))
        for i, row in data.iterrows():
            abit_id = self.abit_mapping[row['abit_id']]
            pref_id = self.pref_mapping[row['pref_id']]
            preference = row['preference']
            self.abit_pref_matrix[abit_id, pref_id] = preference

    def train(self):
        # Обучение модели
        self.similarity_matrix = cosine_similarity(self.abit_pref_matrix)

    def recommend(self, abit_id, n=5):
        # Генерация рекомендаций для абитуриента
        abit_index = self.abit_mapping[abit_id]
        abit_preferences = self.abit_pref_matrix[abit_index, :]
        similarity_scores = self.similarity_matrix[abit_index, :]
        # Сортировка по убыванию схожести и выбор топ N
        top_indices = np.argsort(similarity_scores)[-1:-n]
        recommendations = []
        for index in top_indices:
            pref_id = [pref for pref, mapping_index in self.pref_mapping.items() if
mapping_index == index][0]
            recommendations.append(pref_id)
        return recommendations

# Пример использования
data = pd.DataFrame({
    'abit_id': ['abit1', 'abit1', 'abit1', 'abit2', 'abit2', 'abit3', 'abit3'],
    'pref_id': ['pref1', 'pref2', 'pref3', 'pref1', 'pref3', 'pref2', 'pref3'],
    'preference': [5, 3, 2, 4, 5, 1, 2]
})

recommender = RecommenderSystem(data)
recommender.preprocess_data()
recommender.train()
abit_id = 'abit1'
recommended_prefs = recommender.recommend(abit_id)
print(f"Recommended prefs for abit {abit_id}:")
for pref in recommended_prefs:
    print(pref)

```

Примечание: программный код был разработан автором

Составлено автором по материалам исследования

Рис. 5. Фрагмент кода рекомендательного сервиса по выбору направления подготовки для абитуриентов

Стоит добавить, что данный рекомендательный сервис предполагается реализовать в виде веб-приложения, где абитуриенты могут ввести свои предпочтения (критерии вузов), а затем получить список рекомендованных направлений подготовки в определенных заведениях. Кроме того, стоит отметить, что рейтинги дисциплин (образовательных услуг) в вузах составляют студенты старших курсов. Это позволит абитуриентам принимать более информированные решения о выборе своего будущего образования.

В данном примере был создан класс RecommenderSystem, который инициализируется данными предпочтений абитуриентов. Далее происходит предобработка данных и обучение модели на базе матрицы схожести абитуриентов. Затем с помощью метода recommend можно получить рекомендации для конкретного студента.

Именно в рассмотренном виде представленный сервис имеет ряд преимуществ. Во-первых, он помогает абитуриентам выбрать направление подготовки, которое соответствует их интересам и предпочтениям. Во-вторых, он учитывает предпочтения других студентов схожего профиля и направления, что увеличивает точность рекомендаций. В-третьих, он может быть полезным инструментом вузов, способствуя привлечению абитуриентов и увеличению рейтинга своих программ и направлений подготовки, а также эффективной платформой для продвижения образовательных услуг образовательного учреждения.

Часть кода рекомендательного сервиса по выбору направления подготовки для будущих студентов представлена на рис. 5.

## ЗАКЛЮЧЕНИЕ

На основе предоставленного рекомендательного сервиса была разработана и реализована упрощенная версия коллаборативной фильтрации с использованием косинусной меры сходства. Важно отметить, что существуют и другие подходы к коллаборативной

фильтрации, а также разные формулы для расчета рейтингов и меры схожести, особенно в образовательной сфере. Тем самым разработка рекомендательного сервиса для абитуриентов по выбору направления подготовки в вузе на основе метода коллаборативной фильтрации машинного обучения является актуальной и полезной задачей.

Для дальнейшего развития научного исследования будет проведена оптимизация структуры данных с использованием разреженных матриц для эффективного хранения информации. Текстовые имена студентов, пользователей, критериев и вузов заменят числовые идентификаторы с целью повышения производительности [16]. Далее исследуются другие алгоритмы фильтрации с учетом особенностей этой системы, что поможет улучшить точность и качество рекомендаций. Также планируется изучить и выбрать наилучшую меру сходства на основе анализа данных системы, что также сделает процесс коллаборативной фильтрации более результативным и повысит точность рекомендаций. Кроме того, будет проведена оптимизация производительности. Планируется использование кластеризации пользователей и расчета меры сходства только между пользователями внутри одного кластера.

Также усовершенствуется расчет коэффициентов схожести между услугами для периодического выполнения. Для этого необходимо транспонировать матрицу абитуриент – образовательная услуга и вычислить наиболее похожие услуги при помощи косинусной меры. Это потребует значительных ресурсов и может выполняться по расписанию. Затем, используя полученный список похожих услуг и оценки пользователей, можно предложить рекомендации. Все данные шаги направлены на оптимизацию и улучшение производительности системы коллаборативной фильтрации, чтобы она стала более адаптивной и эффективной в контексте конкретных требований и данных.

### Список литературы

1. *Martínez-Maldonado T., Yacef K.* Toward a learning analytics dashboard for supporting teachers as “learning designers” of orchestrated CSCL orchestration scripts. *Journal of Learning Analytics*. 2019.
2. *Атаян А.М., Гурьева Т.Н., Шабабаева А.Ю.* Цифровая трансформация высшего образования: проблемы, возможности, перспективы и риски. *Отечественная и зарубежная педагогика*. 2021;2(75(1)):7–22.
3. *Седова Н.О., Перцева И.А., Юрьева О.А.* Анализ эффективности профориентационных мероприятий вуза с использованием инструментов машинного обучения. *Современные информационные технологии и ИТ-образование*. 2021;4(17):1032–1046. <https://doi.org/10.25559/SITITO.17.202104.1032-1046>
4. *Забокшицкая А.Д., Орешкина Т.А., Обабков И.Н., Чепуров Е.Г.* Применение алгоритма машинного обучения для профориентации абитуриентов высшего учебного заведения. *Вестник Томского государственного университета*. 2022;485:217–225. <http://doi.org/10.17223/15617793/485/24>
5. *Бозиева А.М., Цеева Ф.М., Хатухова Д.В.* Применение методов машинного обучения при оценке деятельности образовательной организации высшей школы. *Известия Кабардино-Балкарского научного центра РАН*. 2023;3(113):11–19. <http://doi.org/10.35330/1991-6639-2023-3-113-11-19>
6. *Васильева Е.В., Сахнюк П.А.* Разработка модели классификации квалифицированных кадров для IT-отрасли с помощью методов машинного обучения. *Инновации и Инвестиции*. 2023;9:296–299.
7. *Каледин О.Е., Каледина Е.А., Щербяков Д.В.* Прогнозирование успеваемости студентов вуза на основе алгоритмов коллаборативной фильтрации. *Вестник компьютерных и информационных технологий*. 2023;2(224(20)):37–43. <http://doi.org/10.14489/vkit.2023.02.pp.037-043>
8. *Старостин В.С., Аржанова К.А., Долгополов Д.В., Дитрих А.А.* Определение готовности вуза к внедрению технологий искусственного интеллекта для персонализации образовательных траекторий *Вестник университета*. 2023;10:29–39. <http://doi.org/10.26425/1816-4277-2023-10-29-39>
9. *Дорофеева Ю.А., Правдин К.В., Михайлов В.И.* Моделирование и прогнозирование эффективности образовательного процесса в университете. *Международный научно-исследовательский журнал*. 2023;11(137). <https://doi.org/10.23670/IRJ.2023.137.15>
10. *Мищенко В., Ромакина О.М., Арсеньева А.З.* О применении цифрового инструментария при выборе абитуриентом направления подготовки. *Известия Воронежского государственного педагогического университета*. 2023;2(299):14–18.
11. *Никандров А.А., Пиотровская К.Р.* Организация мультиплатформенного гибридного онлайн-курса по «Машинному обучению» для системы высшего образования. *Преподаватель XXI век*. 2023;3–1:110–127.
12. *Diakopoulos N.J., Koliska A., Karatzoglou S.G.* Fairness-aware learning for recommendation ranking in two-sided platforms. *ACM Transactions on Information Systems*. 2020.

13. Захарова И.Г., Воробьева М.С., Боганюк Ю.В. Сопровождение индивидуальных образовательных траекторий на основе концепции объяснимого искусственного интеллекта. Образование и наука. 2022;1(24):163–190. <https://doi.org/10.17853/1994-5639-2022-1-163-190>
14. Преображенский Ю.П., Коновалов В.М. О методах создания рекомендательных систем. Вестник Воронежского института высоких технологий. 2019;4(31):75–79.
15. Егина Н.А. Трансформация модели поведения потребителя в условиях цифровой экономики. Финансы и кредит. 2019;9(789(25)):1971–1986. <https://doi.org/10.24891/fc.25.9.1971>
16. Wang G., Karypis G. Item2Vec: neural item embedding for collaborative filtering. ACM Transactions on Information Systems. 2019.

## References

1. Martínez-Maldonado T., Yacef K. Toward a learning analytics dashboard for supporting teachers as “learning designers” of orchestrated CSCL orchestration scripts. Journal of Learning Analytics. 2019.
2. Atayan A.M., Guryeva T.N., Sharabaeva L.Yu. Digital transformation of higher education: challenges, opportunities, prospects and risks. Domestic and Foreign Pedagogy. 2021;2(75(1)):7–22. (In Russian).
3. Sedova N.O., Pertseva I.A., Yur'eva O.D. Analysis of the effectiveness of vocational guidance activities of the university using machine learning. Modern Information Technologies and IT education. 2021;4(17):1032–1046. (In Russian). <https://doi.org/10.25559/SITITO.17.202104.1032-1046>
4. Zabokritskaya L.D., Oreshkina T.A., Obabkov I.N., Chepurov E.G. Application of machine learning algorithm for career guidance of university applicants. Tomsk State University Journal. 2022;485:217–225. (In Russian). <http://doi.org/10.17223/15617793/485/24>
5. Bozjeva A.M., Tseeva F.M., Khatukhova D.V. Application of machine learning methods in assessing the activities of an educational organization of higher school. News of the Kabardino-Balkarian Scientific Center of RAS. 2023;3(113):11–19. (In Russian). <http://doi.org/10.35330/1991-6639-2023-3-113-11-19>
6. Vasilieva E.V., Sakbnyuk P.A. Development of a model for classifying qualified personnel for the IT industry using machine learning methods. Innovations and Investments. 2023;9:296–299. (In Russian).
7. Kaledin O.E., Kaledina E.A., Shcherbakov D.V. Prediction of the progress of university students based on collaborative filtering algorithms. Herald of computer and information technologies. 2023;2(224(20)):37–43. (In Russian). <http://doi.org/10.14489/vkit.2023.02.pp.037-043>
8. Starostin V.S., Arzhanova K.A., Dolgoplov D.V., Ditrakh A.D. Determining university's readiness to implement AI technologies for personalizing educational paths. Vestnik universiteta. 2023;10:29–39. (In Russian). <http://doi.org/10.26425/1816-4277-2023-10-29-39>
9. Dorofeeva Y.A., Prandin K.V., Mikhailov V.I. Modelling and forecasting the effectiveness of the educational process at universities. International Research Journal. 2023;11(137). (In Russian). <https://doi.org/10.23670/IRJ.2023.137.15>
10. Mischenko V., Romakina O.M., Arseneva A.Z. Application of digital tools when choosing the direction of training by applicant. Izvestia Voronezh State Pedagogical University. 2023;2(299):14–18. (In Russian).
11. Nikandrov A.A., Piotrowska X.R. Organization of a hybrid multi-platform online course on “Machine learning” for higher education systems. Prepodavatel XXI vek. 2023;3–1:110–127. (In Russian).
12. Diakopoulos N.J., Koliska A., Karatzoglou S.G. Fairness-aware learning for recommendation ranking in two-sided platforms. ACM Transactions on Information Systems. 2020.
13. Zakharova I.G., Vorobeva M.S., Boganyuk Yu.V. Support of individual educational trajectories based on the concept of explainable artificial intelligence The Education and science journal. 2022;1(24):163–190. (In Russian). <https://doi.org/10.17853/1994-5639-2022-1-163-190>
14. Preobrazhenskiy Y.P., Konovalov V.M. About methods for creating recommendation systems. Bulletin of Voronezh Institute of High Technologies. 2019;4(31):75–79. (In Russian).
15. Egina N.A. Transformation of the consumer behavior model in the digital economy. Finance and Credit. 2019;9(789(25)):1971–1986. (In Russian). <https://doi.org/10.24891/fc.25.9.1971>
16. Wang G., Karypis G. Item2Vec: neural item embedding for collaborative filtering. ACM Transactions on Information Systems. 2019.