

**МАТЕМАТИЧЕСКОЕ И ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ,
КОМПЛЕКСОВ И КОМПЬЮТЕРНЫХ СЕТЕЙ/MATHEMATICAL SOFTWARE FOR COMPUTERS,
COMPLEXES AND COMPUTER NETWORKS**

DOI: <https://doi.org/10.60797/IRJ.2025.156.41>

РАЗРАБОТКА ГИБРИДНОЙ РЕКОМЕНДАТЕЛЬНОЙ СИСТЕМЫ ДЛЯ АВИАПАССАЖИРОВ

Научная статья

Староверов И.М.^{1,*}, Князев В.Н.²

¹ ORCID : 0009-0002-4977-953X;

² ORCID : 0000-0003-2142-0277;

^{1,2} Пензенский государственный университет, Пенза, Российская Федерация

* Корреспондирующий автор (ilstar01[at]yandex.ru)

Аннотация

Рассматриваются актуальные вопросы разработки и применения рекомендательных систем в сфере пассажирских авиаперевозок.

В качестве научной новизны проведенного исследования предлагается оригинальный гибридный подход, сочетающий колаборативную фильтрацию, графовые модели и методы машинного обучения. Предложенный подход отличается от известных решений тем, что позволяет повысить точность прогнозов и релевантность предложений.

В качестве практической значимости проведенного исследования на основе предложенного оригинального подхода проведена реализация гибридной рекомендательной системы для авиапассажиров. Выполнено тестирование разработанной рекомендательной системы, которое показало корректность работы и эффективность ее применения. Программное обеспечение разработанной рекомендательной системы зарегистрировано в Роспатенте путем получения Свидетельства о государственной регистрации программы для ЭВМ [17].

Также описаны этапы разработки предложенной рекомендательной системы, описана разработка и реализация программного интерфейса (API) для интеграции рекомендательной системы с разработанным мобильным приложением для покупки и бронирования авиабилетов в составе автоматизированной системы обслуживания авиапассажиров [7].

Рассматриваются перспективы дальнейшего развития системы, включая оценку качества, использование методов глубокого обучения и расширение функциональности рекомендаций.

Ключевые слова: рекомендательная система, колаборативная фильтрация, графовые модели, PageRank, автоматизированные системы обслуживания авипассажиров.

DEVELOPMENT OF A HYBRID RECOMMENDATION SYSTEM FOR AIR PASSENGERS

Research article

Staroverov I.M.^{1,*}, Knyazev V.N.²

¹ ORCID : 0009-0002-4977-953X;

² ORCID : 0000-0003-2142-0277;

^{1,2} Penza State University, Penza, Russian Federation

* Corresponding author (ilstar01[at]yandex.ru)

Abstract

The current issues of development and application of recommendation systems in the field of passenger air travel are examined.

As scientific novelty of the conducted research, an original hybrid approach combining collaborative filtering, graph models and machine learning methods is suggested. The proposed approach differs from the known solutions in that it allows increasing the accuracy of predictions and relevance of suggestions.

As a practical significance of the conducted research, a hybrid recommendation system for air passengers was implemented on the basis of the suggested original approach. The developed recommendation system was tested, which showed its correct operation and efficiency. The software of the developed recommendation system was registered in Rospatent by obtaining the Certificate of State Registration of Computer Programme [17].

The stages of development of the proposed recommendation system are also described, and the design and application programming interface (API) for the integration of the recommendation system with the developed mobile application for buying and booking airline tickets as part of the automated airline passenger service system is outlined [7].

Prospects for further development of the system are discussed, including quality assessment, the use of deep learning techniques, and the expansion of recommendation functionality.

Keywords: recommendation system, collaborative filtering, graph models, PageRank, automated air passenger service systems.

Введение

В современном мире рекомендательные системы на базе искусственного интеллекта играют все более значимую роль, предоставляя персонализированный контент и помогая пользователям ориентироваться в обилии доступных опций [1]. В сфере авиаперелетов рекомендательные системы позволяют путешественникам находить наиболее

подходящие варианты, облегчая выбор среди множества направлений [2]. Для создания эффективных рекомендаций используются различные подходы и алгоритмы, такие как ассоциативные правила, метрики сходства (L_p -нормы, коэффициенты Оттаи, Пирсона и другие), фильтрации по контексту и др. [3], [4].

При этом эффективность систем повышается при одновременном использовании различных методов. Например, ассоциативные правила определяют общие шаблоны в данных, а метрики сходства персонализируют рекомендации. Это подтверждается практическими результатами, полученными крупными авиакомпаниями и сервисами по бронированию авиабилетов, такими как «S7 Airlines», «Booking.com», «Авиасейлс» [5], [6]. Применение таких систем привело к значительному росту конверсии из просмотров в бронирования, а также упрощению планирования путешествий и разработке маршрутов.

В условиях быстрого развития информационных технологий и роста объемов данных перспективы рекомендательных систем в сфере авиаперелетов расширяются. В данной статье будет рассмотрена разработка оригинальной рекомендательной системы в сфере авиаперелетов на основе гибридного подхода, которая будет анализировать различные факторы: предпочтения пассажира, историю его путешествий, текущие сезонные тренды и многое другое. Такая система делает процесс выбора авианаправления более персонализированным и удовлетворяющим потребностям каждого пользователя.

Этапы разработки рекомендательной системы

Разработка эффективной рекомендательной системы включает в себя множество этапов. Рассмотрим общую схему взаимодействий между различными этапами системы, включая сбор данных, их обработку, обучение моделей и предоставление рекомендаций пользователям.

В отличие от существующих решений, в данной работе предлагается принципиально новый гибридный подход, основанный на оригинальной комбинации методов коллаборативной и контентной фильтрации, графовых моделей и машинного обучения. Основные этапы разработки системы представлены на рисунке 1.

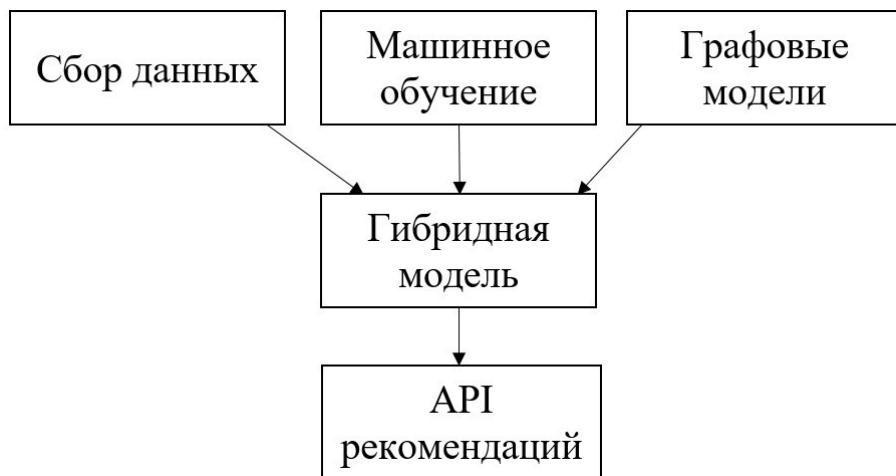


Рисунок 1 - Этапы разработки рекомендательной системы

DOI: <https://doi.org/10.60797/IRJ.2025.156.41.1>

На этапе сбора данных были использованы разнообразные источники информации.

1. История просмотров городов: данные о том, какие города пользователи просматривали до этого в разработанном мобильном приложении для покупки и бронирования авиабилетов [7].

2. Покупки билетов: информация о купленных билетах, включая города отправления и назначения, даты.

3. Предварительные бронирования: данные о городах, на которые пользователи предварительно бронировали билеты.

4. Интересы пользователей: анкеты и профили пользователей, содержащие их предпочтения и интересы (например, пляжный отдых, исторические места и т.д.).

5. Сезонность: коэффициенты популярности городов в зависимости от текущего сезона.

6. Популярность: общие коэффициенты популярности городов.

Сбор данных осуществляется путем выполнения SQL-запросов к базе данных аэропорта.

На этапе машинного обучения были использованы различные модели для прогнозирования предпочтений пользователей.

1. Коллаборативная фильтрация: метод сингулярного разложения матрицы (SVD) для определения схожести между пользователями и городами.

2. Модель случайного леса: используется для прогнозирования вероятности выбора конкретного направления на основе профиля пользователя и его истории.

3. Линейная регрессия: объединяет результаты коллаборативной фильтрации и графовых моделей для создания финальной гибридной модели.

Графовые модели позволяют представить города и перелеты в виде ориентированного взвешенного графа.

1. Вершины представляют города, а дуги — вероятности перелетов между ними на основе предпочтений пользователя.

2. Алгоритм PageRank [8] определяет ранжирование городов на основе вероятности посещения каждого города.

Гибридная модель объединяет результаты всех вышеуказанных моделей, выполняя следующие этапы.

1. Ранжирование городов осуществляется на основе коллаборативной фильтрации и модели случайного леса.

2. Смешивание результатов выполняется с помощью линейной регрессии, которая комбинирует результаты коллаборативной фильтрации, графовых моделей и модели случайного леса.

Взаимодействие рекомендательной системы с разработанным мобильным приложением для покупки и бронирования авиабилетов осуществляется через API. Оно обеспечивает:

- получение рекомендаций для конкретного пользователя, основанные на его профиле и активности;

- предоставление обновленной информации о пользователе и его предпочтениях.

Результаты коллаборативной фильтрации, графовых моделей и случайного леса комбинируются с помощью модели линейной регрессии для получения гибридной модели и определения финального рейтинга городов.

API рекомендаций выполняет следующие действия:

- предоставляет доступ к рекомендациям через HTTP-запросы для мобильного приложения;

- обновляет информацию о пользователях и сохраняет результаты рекомендаций в базу данных.

Разработанное мобильное приложение для покупки и бронирования авиабилетов взаимодействует с API рекомендаций следующим образом.

1. Запрос авторизации. Приложение отправляет запрос на авторизацию и получает JWT-токен для дальнейшей работы.

2. Получение рекомендаций. Приложение отправляет запрос с JWT-токеном для получения рекомендаций для конкретного пользователя.

3. API возвращает список городов, отсортированных по убыванию релевантности.

4. Приложение отображает полученные рекомендации в пользовательском интерфейсе.

5. Обновление данных пользователя. При изменении предпочтений или активности пользователя приложение отправляет обновленные данные на сервер.

Алгоритмы и модели рекомендаций

Рассмотрим подробнее алгоритмы и модели, заложенные в основу оригинальной рекомендательной системы для авиаперелетов. Система использует комбинацию нескольких методов, включая коллаборативную фильтрацию, графовые модели, машинное обучение. Это обеспечивает более точную и персонализированную выдачу рекомендаций для пользователей.

Коллаборативная фильтрация (Collaborative filtering) — это метод рекомендации, при котором анализируется только реакция пользователей на объекты: оценки, которые выставляют пользователи объектам [9], [10], [11]. Оценки могут быть как явными (пользователь явно указывает, на сколько «звездочек» он оценивает объект), так и неявными (например, количество просмотров одного ролика). Чем больше оценок собирается, тем точнее получаются рекомендации. В нашем случае используются неявные оценки.

Получается, что пользователи помогают друг другу в фильтрации объектов. Поэтому такой метод называется также совместной фильтрацией.

В нашем случае используется метод сингулярного разложения матрицы (SVD), который позволяет выявить скрытые факторы предпочтений пользователей.

Создание рекомендательной системы производилось на языке программирования Python. Сначала формируется матрица взаимодействий «пассажир-город», где строки представляют пассажиров, а столбцы — города. Значения в ячейках отражают коэффициент взаимодействия пассажира с определенным городом (неявные оценки) на основе его активности (покупка билета добавляет 5 единиц, бронирование — 3, просмотр — 1).

Для построения модели коллаборативной фильтрации используется сингулярное разложение (SVD). Сначала выполняется разложение матрицы на три компонента: U, sigma и Vt. Затем вычисляются факторы пассажиров и городов. Ниже приведен листинг кода, в котором производятся данные шаги.

```
from scipy.sparse.linalg import svds
import numpy as np

# Вычисление сингулярного разложения для коллаборативной фильтрации
U, sigma, Vt = svds(passenger_city_matrix.to_numpy(), k=10)
sigma = np.diag(sigma)
passenger_factors = U @ sigma
city_factors = Vt.T
```

С помощью полученных факторов можно прогнозировать предпочтения каждого пассажира к городам, например, с помощью функции, представленной в листинге ниже.

```
# Функция для прогнозирования предпочтений пассажира к городам с использованием SVD
def predict_city_preferences_svd(passenger_id):
    passenger_idx = passenger_city_matrix.index.get_loc(passenger_id)
```

```

passenger_vector = passenger_factors[passenger_idx, :]
scores = passenger_vector @ city_factors.T
return pd.Series(scores, index=passenger_city_matrix.columns)

# Предсказания на основе SVD
svd_predictions = predict_city_preferences_svd(id_passenger).sort_values(ascending=False)

```

Иллюстрацию работы алгоритма колаборативной фильтрации можно увидеть на рисунке 2, который визуализирует процесс создания и разложения матрицы «пассажир-город» с помощью SVD.

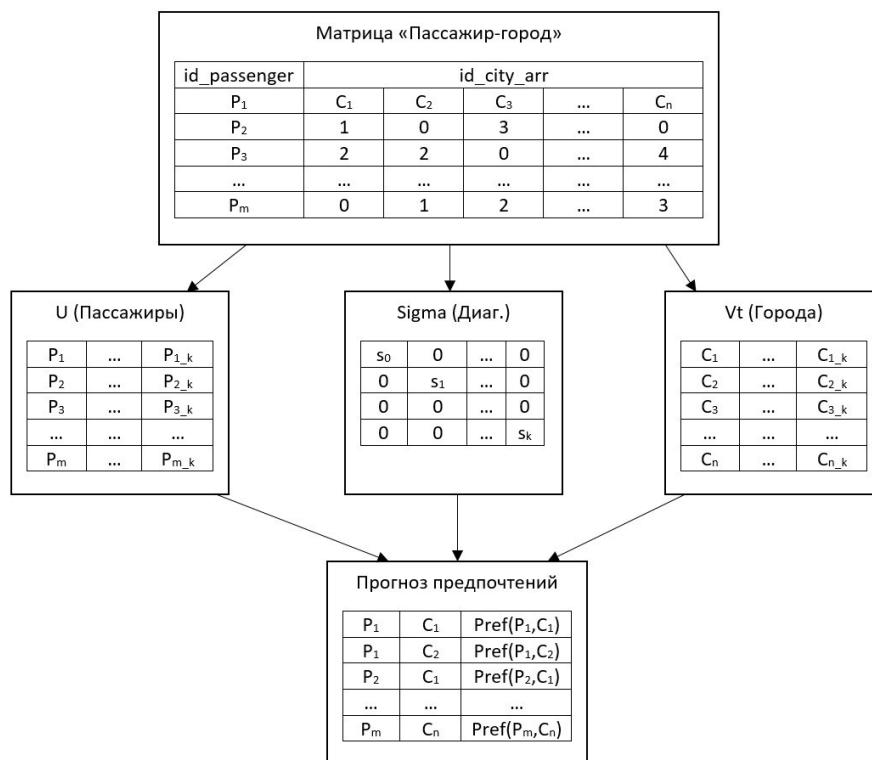


Рисунок 2 - Иллюстрация алгоритма колаборативной фильтрации
DOI: <https://doi.org/10.60797/IRJ.2025.156.41.2>

Графовые модели в рекомендательных системах представляют собой мощный инструмент для моделирования и анализа сложных структур данных, где объекты и их взаимосвязи могут быть наглядно представлены в виде вершин и дуг графа [12]. Эти модели позволяют учесть не только характеристики отдельных элементов (например, пользователей или товаров), но и типы и силу их взаимодействий, что делает их особенно ценными для создания персонализированных рекомендаций.

В контексте рекомендательных систем, графовые модели используются для представления и анализа разнообразных данных, таких как социальные сети, пользовательские предпочтения, товарные каталоги и их взаимосвязи. Эти модели способствуют более глубокому пониманию структуры предпочтений и поведения пользователей, позволяя обнаруживать скрытые паттерны и предсказывать потенциальный интерес к определенным товарам или услугам.

В нашем случае графовые модели позволяют представить города и перелеты в виде ориентированного графа, где вершины — это города, а дуги — вероятности перелетов между ними. Веса дуг зависят от предпочтений пользователя и его активности.

Для каждого пассажира создается взвешенный ориентированный граф предпочтений. Вершины представляют города, а дуги — вероятности перелетов между ними, основываясь на профиле пользователя и его предыдущей активности. Листинг кода, осуществляющего создание такого графа, представлен ниже.

```

import networkx as nx

graphs = {}

for id_passenger in data['id_passenger'].unique():
    G = nx.DiGraph()
    G.add_nodes_from(all_cities)

```

```

passenger_data = data[(data['id_passenger'] == id_passenger)].iloc[[0]]

# Добавление дуг с весами
for city_departure in all_cities:
    for city_arrival in all_cities:
        if city_departure == city_arrival:
            continue

        G.add_edge(city_departure, city_arrival, weight=coeff)
graphs[id_passenger] = G

```

Для ранжирования городов в графе предпочтений используется алгоритм PageRank. Он определяет вероятность посещения каждого города на основе случайных блужданий. Идея алгоритма заключается в определении важности веб-страницы, исходя из количества и качества ссылок на нее [8]. Если важная страница ссылается на другую страницу, то последняя также считается важной. Аналогично, в нашем случае PageRank используется для ранжирования городов в графе предпочтений пассажиров.

Для ранжирования городов в графах предпочтений применялся алгоритм PageRank, реализованный в библиотеке networkx. Алгоритм вычисляет для каждого города значение PageRank, отражающее его «важность» в графе, что интерпретируется как вероятность посещения города. Параметр $\alpha=0.85$ задает коэффициент демпфирования. Результаты PageRank, представляющие собой словарь «город-значение», сортируются по убыванию значения для получения ранжированного списка городов.

Упрощенно созданный график можно представить на рисунке 3. На нем изображено в качестве узлов 3 города (C_1 , C_2 и C_3), а также дуги, представляющие вероятность перехода из одного города в другой, веса которых высчитываются для каждого пассажира отдельно с использованием модели случайного леса. Также в каждом узле определен PageRank. Как можно заметить, наименее релевантным городом в данном графике является C_3 .

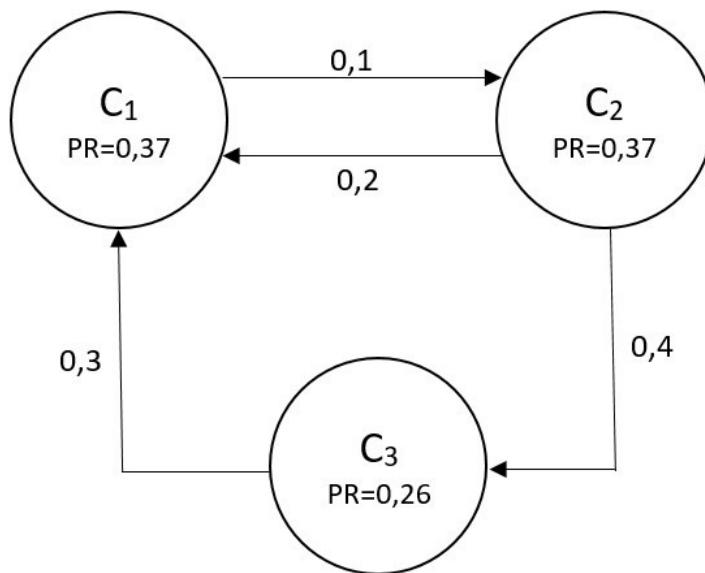


Рисунок 3 - Иллюстрация графовой модели
DOI: <https://doi.org/10.60797/IRJ.2025.156.41.3>

Случайный лес — это ансамблевый метод машинного обучения, использующий множество решающих деревьев для определения наилучшего решения через усреднение выводов отдельных деревьев [13]. Эта модель особенно эффективна для решения задач регрессии и классификации благодаря своей способности обрабатывать большие объемы данных с высокой точностью и устойчивостью к переобучению.

В контексте рекомендательных систем, случайный лес может использоваться для прогнозирования интересов пользователя, базируясь на его профиле и истории взаимодействий [14]. В нашем случае модель обучается на данных о пассажирах и их предпочтениях относительно различных городов и типов отдыха. Так как города являются категориальными признаками, то для них сначала применяется кодирование по принципу One-hot.

Помимо используемого в работе метода случайного леса в современной научной литературе также рассматриваются альтернативные методы прогнозирования предпочтений пользователей на основе популяционных алгоритмов [15].

Для прогнозирования коэффициента предпочтения города была использована модель случайного леса, реализованная в библиотеке scikit-learn (класс RandomForestRegressor). Модель была обучена на подготовленных данных признаков и целевой переменной. Для каждого пассажира были сформированы входные данные (input_data), соответствующие его характеристикам и закодированным городам, и выполнен прогноз с помощью обученной модели (RandomForestRegressor.predict(...)).

Гибридная модель объединяет результаты коллаборативной фильтрации, графовых моделей и машинного обучения для формирования финальных рекомендаций, используя модель линейной регрессии из библиотеки scikit-learn (класс LinearRegression). Модель была обучена на наборе данных, где входными признаками (X_{train}) выступали оценки, полученные методами и алгоритмами PageRank, коллаборативной фильтрации (SVD) и случайного леса, а целевой переменной (y_{train}) — финальный коэффициент предпочтения. После обучения модель использовалась для прогнозирования итогового коэффициента (predicted_coeff), комбинируя оценки от различных моделей.

После прогнозирования предпочтений результаты сохраняются в базу данных для последующего использования в API рекомендаций.

После того как все данные обработаны, и рекомендации добавлены в базу данных, следующим шагом является создание API для предоставления этих рекомендаций пользователям через мобильное приложение.

Данное приложение представляет собой удобный и простой в использовании инструмент для покупки или бронирования авиабилетов на определенные рейсы. Пользователи могут просматривать подробную информацию о рейсах, включая дату и время вылета и прилета, количество свободных мест, цену, тип самолета и многое другое. Приложение позволяет легко и быстро оформить бронирование или покупку билетов прямо с мобильного устройства, предоставляя также возможность повторного использования данных пассажиров, изменения информации о пассажире или покупателе, выкупа или снятия брони, а также возврата билетов.

Для реализации API используется контроллер в ASP.NET MVC, который обращается к базе данных, где хранятся рекомендации, и возвращает список городов, отсортированных по убыванию коэффициента (листинг представлен ниже).

```
// GET: Cities/Recommendations
[HttpGet]
public ActionResult Recommendations(int id_passenger, int n = 3)
{
    var recommendations = db.Recommendation
        .Where(r => r.id_passenger == id_passenger)
        .OrderByDescending(r => r.coeff)
        .Take(n)
        .Select(r => r.City.name_city)
        .ToList();

    if (recommendations.Count == 0)
    {
        // Список городов по умолчанию, если рекомендации не найдены
        recommendations = new List<string> { "Москва", "Пенза", "Санкт-Петербург" };
    }

    return Json(recommendations, JsonRequestBehavior.AllowGet);
}
```

Мобильное приложение взаимодействует с API, отправляя запросы на получение рекомендаций для конкретного пользователя.

Пример кода реализации запроса рекомендаций в мобильном приложении представлен ниже.

```
public static async Task<List<string>> GetRecommendationsAsync(int id_passenger, int n)
{
    // Получение jwt-токена
    var token = await SecureStorage.GetAsync("jwt_token");
    Client.DefaultRequestHeaders.Authorization = new AuthenticationHeaderValue("Bearer", token);

    // Отправка GET-запроса и обработка ответа
    var response = await Client.GetAsync(url + [_rj_content_placeholder_]quot;Cities/Recommendations?
id_passenger={id_passenger}&n={n}");
    if (response.IsSuccessStatusCode)
    {
        string responseText = await response.Content.ReadAsStringAsync();
        List<string> recommendations = JsonConvert.DeserializeObject<List<string>>(responseText);
        return recommendations;
    }
    else
    {
        return new List<string>();
    }
}
```

В контексте аппаратной реализации рекомендательной системы и API, представляющим собой связующее звено между серверной частью и мобильным приложением, стоит отметить возможность использования встраиваемых решений [16].

Было проведено тестирование разработанной рекомендательной системы. Примеры вывода рекомендованных авианаправлений из мобильного приложения представлены на рисунках 4 и 5. В первом случае пассажир периодически летает в Сочи и Пензу, поэтому ему предлагают данные направления. Но также он бывал несколько раз в Нижнем Новгороде. А так как помимо этого в его профиле указано, что он имеет высокую оценку предпочтения исторических мест, то ему также предлагается посетить Ереван, в котором он ни разу не был.

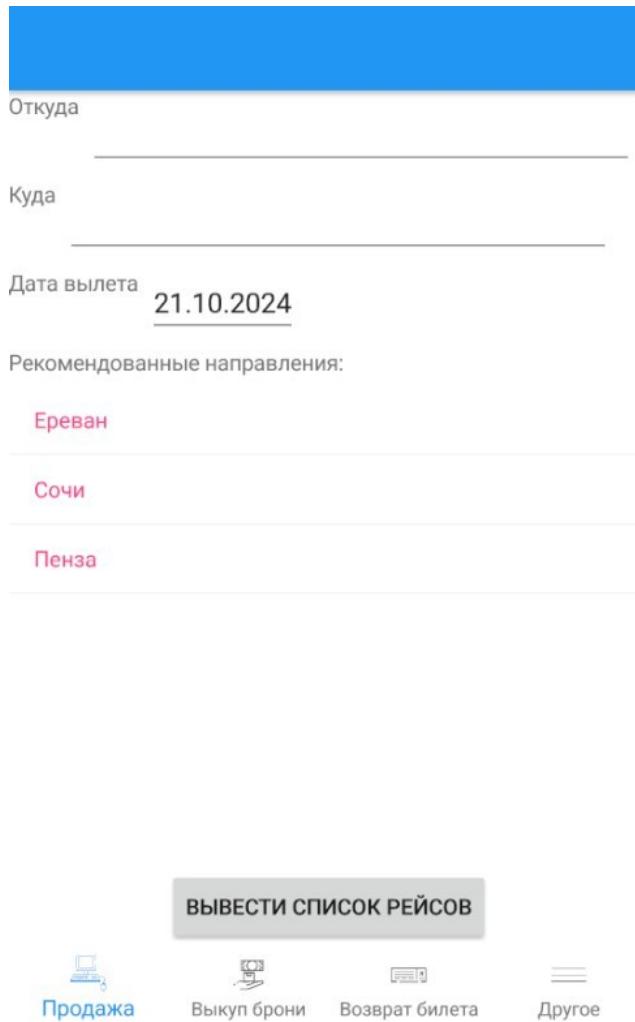


Рисунок 4 - Пример вывода рекомендованных авианаправлений для пассажира 1
DOI: <https://doi.org/10.60797/IRJ.2025.156.41.4>

Во втором случае пассажир очень часто летает в Казань и Афины через Москву. В других городах он ни разу не был, а интересы не выставлены в его профиле. Поэтому рекомендательная система предлагает ему только эти 3 города.

Откуда

Куда

Дата вылета
21.10.2024

Рекомендованные направления:

Казань

Афины

Москва

ВЫВЕСТИ СПИСОК РЕЙСОВ



Рисунок 5 - Пример вывода рекомендованных авианаправлений для пассажира 2
DOI: <https://doi.org/10.60797/IRJ.2025.156.41.5>

Таким образом, были подробно рассмотрены алгоритмы и модели, заложенные в основу рекомендательной системы. Коллаборативная фильтрация, графовые модели и машинное обучение в сочетании с гибридной моделью обеспечивают персонализированный и точный подбор рекомендаций для пользователей. Финальное API позволяет интегрировать рекомендации в мобильное приложение, обеспечивая быстрый доступ к релевантным направлениям.

Заключение

Были рассмотрены актуальные вопросы разработки и использования рекомендательных систем в сфере пассажирских авиаперевозок.

Научной новизной проведенного исследования является предложенный оригинальный гибридный подход, сочетающий в себе применение коллаборативной фильтрации, графовых моделей и методов машинного обучения. Предложенный подход отличается от известных решений тем, что позволяет повысить точность прогнозов и релевантность предложений.

Практическая значимость проведенного исследования заключается в том, что на основе предложенного оригинального подхода была разработана и реализована гибридная рекомендательная система для авиапассажиров, а также выполнено тестирование разработанной рекомендательной системы, которое показало корректность работы и эффективность ее применения. На программное обеспечение разработанной рекомендательной системы получено Свидетельство о государственной регистрации программы для ЭВМ [17].

Были описаны этапы разработки предложенной рекомендательной системы, описана разработка и реализация программного интерфейса (API) для интеграции предложенной рекомендательной системы с разработанным мобильным приложением для покупки и бронирования авиабилетов в составе комплексной автоматизированной системы обслуживания авиапассажиров.

Хотя текущая реализация рекомендательной системы уже дает хорошие результаты, есть несколько направлений для дальнейшего ее улучшения и развития.

1. Тестирование и оценка качества на реальных данных. Проведение тестирования и оценки качества рекомендательной системы на реальных данных для определения релевантности рекомендаций.

2. Улучшение точности рекомендаций. Использование глубокого обучения: рекуррентных нейронных сетей (RNN) и сверточных нейронных сетей (CNN) для более точного прогнозирования предпочтений.

3. Расширение функциональности и интеграция с другими сервисами. Добавление рекомендаций не только направлений, но и событий в выбранных городах, например, концертов, фестивалей или спортивных мероприятий. Формирование пакетных туров, включающих билеты, проживание, экскурсии и другие услуги.
4. Персонализация интерфейса. Адаптация интерфейса под предпочтения пользователя, отображение наиболее релевантной информации.
5. Интерактивные опросы. Проведение опросов и сбор обратной связи от пользователей для улучшения качества рекомендаций.
6. Оптимизация производительности. Использование технологий параллельной обработки данных для ускорения обучения моделей.

Конфликт интересов

Не указан.

Рецензия

Эминов Б.Ф., Казанский национальный исследовательский технический университет им. А.Н. Туполева – КАИ, Казань Российская Федерация
DOI: <https://doi.org/10.60797/IRJ.2025.156.41.6>

Conflict of Interest

None declared.

Review

Eminov B.F., Kazan National Research Technical University named after A.N. Tupolev – KAI, Kazan Russian Federation
DOI: <https://doi.org/10.60797/IRJ.2025.156.41.6>

Список литературы / References

1. Староверов И.М. Вопросы разработки рекомендательной системы для автоматизированной системы обслуживания авиапассажиров. / И.М. Староверов, В.Н. Князев. // Информационные технологии в науке и образовании. Проблемы и перспективы; под ред. Л.Р. Фионовой — Пенза: ПГУ, 2024. — С. 169–172.
2. Камалов К.Ф. Анализ применения искусственного интеллекта в гражданской авиации / К.Ф. Камалов // Вестник науки. — 2023. — Т. 3. — № 11. — С. 896–901.
3. Фальк К. Рекомендательные системы на практике / К. Фальк. — Москва: ДМК Пресс, 2020. — 448 с.
4. Якупов Д.Р. Рекомендательные системы на основе сессий – модели и задачи / Д.Р. Якупов, Д.Е. Намиот // Международный журнал открытых информационных технологий. — 2022. — Т. 10. — № 7. — С. 128–152.
5. S7 Airlines совместно с CleverDATA внедрила рекомендательную систему на основе машинного обучения [Электронный ресурс] // Ведомости. — 2019. — URL: https://www.vedomosti.ru/press_releases/2019/03/21/s7-airlines-sovmestno-s-cleverdata-vnedrila-rekomendatelnuyu-sistemu-na-osnove-mashinnogo-obucheniya. (дата обращения: 10.04.25)
6. РСТ: российский турбизнес активно внедряет искусственный интеллект в работу с клиентами [Электронный ресурс] // Российский союз туриндустрии. — 2023. — URL: <https://rostourunion.ru/novosti/novosti-turizma/rst-rossijskij-turbiznes-aktivno-vnedryuet-iskusstvennyj-intellekt-v-rabotu-s-klientami.html>. (дата обращения: 10.04.25)
7. Староверов И.М. Проектирование автоматизированных систем обслуживания авиапассажиров с применением современных информационных технологий. / И.М. Староверов. // Наш выбор – наука! : Сборник статей IV Международного научно-исследовательского конкурса; — Петрозаводск: Новая наука, 2024. — С. 20–28.
8. Полякова О.С. PageRank. Алгоритм ссылочного ранжирования / О.С. Полякова, А.О. Подлесный // Наука и современность. — 2013. — Т. 1. — № 20. — С. 154–157.
9. Гомzin А.Г. Системы рекомендаций: обзор современных подходов. / А.Г. Гомzin, А.В. Коршунов // Труды Института системного программирования РАН. — 2012. — № 22. — С. 401–418.
10. Диценко А.А. Большие данные и рекомендательные системы. / А.А. Диценко. // XII Всероссийская научно-практическая конференция молодых ученых «Россия молодая»; — Кемерово: КузГТУ, 2020. — С. 21156.1–21156.7.
11. Погорельская Я.С. Обзор подходов к построению рекомендательных систем. / Я.С. Погорельская // Информационно-телекоммуникационные технологии и математическое моделирование высокотехнологичных систем. — 2021. — № 1. — С. 278–282.
12. Понизовкин Д.М. Построение оптимального графа связей в системах колаборативной фильтрации. / Д.М. Понизовкин // Программные системы: теория и приложения. — 2011. — № 4. — С. 107–114.
13. Чистяков С.П. Случайные леса: обзор. / С.П. Чистяков // Труды Карельского научного центра РАН. — 2013. — № 1. — С. 117–136.
14. Бородинов А.А. Разработка и исследование алгоритмов определения предпочтаемых пользователем остановок общественного транспорта в геоинформационной системе на основе методов машинного обучения / А.А. Бородинов // Компьютерная оптика. — 2020. — Т. 44. — № 4. — С. 646–652.
15. Родзин С.И. Построение прогнозов в рекомендательных системах с помощью машинного обучения на основе популяционного алгоритма / С.И. Родзин, О.Н. Родзина // Вестник компьютерных и информационных технологий. — 2020. — Т. 17. — № 1 (187). — С. 48–56. — DOI: 10.14489/vkit.2020.01.pp.048-056
16. Гибадуллин Р.Ф. Разработка аппаратно-программного модуля обнаружения объектов для встраиваемых систем / Р.Ф. Гибадуллин, И.Н. Смирнов, Н.В. Хевронин и др. // Вестник Технологического университета. — 2018. — Т. 21. — № 6. — С. 118–122.
17. Староверов И.М. Программа персонализированных рекомендаций городов для авиапассажиров / И.М. Староверов, В.Н. Князев // Свидетельство о государственной регистрации программы для ЭВМ № 2025617775. Заявка № 2025611693, дата поступления 01.02.2025 г., дата государственной регистрации в Реестре программ для ЭВМ 28.03.2025 г. — Москва: Федеральный институт промышленной собственности, 2025.

Список литературы на английском языке / References in English

1. Staroverov I.M. Voprosy razrabotki rekomendatel'noj sistemy dlya avtomatizirovannoj sistemy obsluzhivaniya aviapassazhirov [Issues of development of a recommendation system for an automated system of service of air passengers]. / I.M. Staroverov, V.N. Knyazev. // Information technologies in science and education. Problems and prospects; edited by L.R. Fionovo — Penza: PGU, 2024. — P. 169–172. [in Russian]
2. Kamalov K.F. Analiz primeneniya iskusstvennogo intellekta v grazhdanskoi aviatsii [Analysis of the application of artificial intelligence in civil aviation] / K.F. Kamalov // Vestnik nauki [Bulletin of Science]. — 2023. — Vol. 3. — № 11. — P. 896–901. [in Russian]
3. Fal'k K. Rekomendatel'nye sistemy na praktike [Recommender systems in practice] / K. Fal'k. — Moscow: DMK Press, 2020. — 448 p. [in Russian]
4. Yakupov D.R. Rekomendatelnie sistemi na osnove sessii – modeli i zadachi [Session-based recommender systems – models and challenges] / D.R. Yakupov, D.E. Namiot // Mezhdunarodniy zhurnal otkritikh informatsionnykh tekhnologii [International Journal of Open Information Technologies]. — 2022. — Vol. 10. — № 7. — P. 128–152. [in Russian]
5. S7 Airlines sovmestno s CleverDATA vnedrila rekomendatel'nuyu sistemу na osnove mashinnogo obucheniya [S7 Airlines, together with CleverDATA, has implemented a recommendation system based on machine learning] [Electronic source] // Vedomosti. — 2019. — URL: https://www.vedomosti.ru/press_releases/2019/03/21/s7-airlines-sovmestno-s-cleverdata-vnedrila-rekomendatelnuyu-sistemу-na-osnove-mashinnogo-obucheniya. (accessed: 10.04.25) [in Russian]
6. RST: rossijskij turbiznes aktivno vnedryaet iskusstvennyj intellekt v rabotu s klientami [RST: Russian tourism industry is actively introducing artificial intelligence into work with clients] [Electronic source] // Russian Union of Travel Industry. — 2023. — URL: <https://rostourunion.ru/novosti/novosti-turizma/rst-rossijskij-turbiznes-aktivno-vnedryaet-iskusstvennyj-intellekt-v-rabotu-s-klientami.html>. (accessed: 10.04.25) [in Russian]
7. Staroverov I.M. Proektirovanie avtomatizirovannyx sistem obsluzhivaniya aviapassazhirov s primeneniem sovremennyx informacionnyx tekhnologij [Design of automated systems for servicing air passengers using modern information technologies]. / I.M. Staroverov. // Our Choice is Science! : Collection of Articles of the IV International Scientific Research Competition; — Petrozavodsk: Novaya nauka, 2024. — P. 20–28. [in Russian]
8. Polyakova O.S. PageRank. Algoritm ssilochnogo ranzhirovaniya [PageRank. Link Ranking Algorithm] / O.S. Polyakova, A.O. Podlesnii // Nauka i sovremenność [Science and modernity]. — 2013. — Vol. 1. — № 20. — P. 154–157. [in Russian]
9. Gomzin A.G. Sistemy' rekomendacij: obzor sovremennyx podxodov [Recommender Systems: A Review of Modern Approaches]. / A.G. Gomzin, A.V. Korshunov // Proceedings of the Institute for System Programming of the Russian Academy of Sciences. — 2012. — № 22. — P. 401–418. [in Russian]
10. Didenko A.A. Bol'shie dannye i rekomendatel'nye sistemy [Big data and recommender systems]. / A.A. Didenko. // XII All-Russian Scientific and Practical Conference of young scientists "YoungRussia"; — Kemerovo: KuzGTU, 2020. — P. 21156.1–21156.7. [in Russian]
11. Pogorelskaya Ya.S. Obzor podxodov k postroeniyu rekomendatel'nyx sistem [Review of approaches to building recommender systems]. / Ya.S. Pogorelskaya // Information and telecommunication technologies and mathematical modeling of high-tech systems. — 2021. — № 1. — P. 278–282. [in Russian]
12. Ponizovkin D.M. Postroenie optimal'nogo grafa svyazej v sistemakh kollaborativnoj fil'tracii [Construction of an optimal graph of connections in collaborative filtering systems]. / D.M. Ponizovkin // Software systems: theory and applications. — 2011. — № 4. — P. 107–114. [in Russian]
13. Chistyakov S.P. Sluchajnye lesa: obzor [Random Forests: An Overview]. / S.P. Chistyakov // Proceedings of the Karelian Scientific Center of the Russian Academy of Sciences. — 2013. — № 1. — P. 117–136. [in Russian]
14. Borodinov A.A. Razrabotka i issledovanie algoritmov opredeleniya predpochitaemikh polzovatelem ostanovok obshchestvennogo transporta v geoinformatsionnoi sisteme na osnove metodov mashinnogo obucheniya [Development and research of algorithms for determining user-preferred public transport stops in a geographic information system based on machine learning methods] / A.A. Borodinov // Kompyuternaya optika [Computer optics]. — 2020. — Vol. 44. — № 4. — P. 646–652. [in Russian]
15. Rodzin S.I. Postroenie prognozov v rekomendatel'nix sistemakh s pomoshchyu mashinnogo obucheniya na osnove populyatsionnogo algoritma [Building Predictions in Recommender Systems Using Machine Learning Based on Population Algorithm] / S.I. Rodzin, O.N. Rodzina // Vestnik kompyuternikh i informatsionnykh tekhnologii [Bulletin of Computer and Information Technologies]. — 2020. — Vol. 17. — № 1 (187). — P. 48–56. — DOI: 10.14489/vkit.2020.01.pp.048-056 [in Russian]
16. Gibadullin R.F. Razrabotka apparatno-programmnogo modulya obnaruzheniya obektov dlya vstraivaemikh sistem [Development of a hardware and software module for detecting objects for embedded systems] / R.F. Gibadullin, I.N. Smirnov, N.V. Khevronin et al. // Vestnik Tekhnologicheskogo universiteta [Bulletin of the Technological University]. — 2018. — Vol. 21. — № 6. — P. 118–122. [in Russian]
17. Staroverov I.M. Programma personalizirovannyh rekomendacij gorodov dlya aviapassazhirov [Personalized city recommendation program for air passengers] / I.M. Staroverov, V.N. Knyazev // Certificate of state registration of computer program No. 2025617775. Application No. 2025611693, date of receipt 01.02.2025, date of state registration in the Register of computer programs 28.03.2025. — Moscow: Federal Institute of Industrial Property, 2025. [in Russian]