Отчет

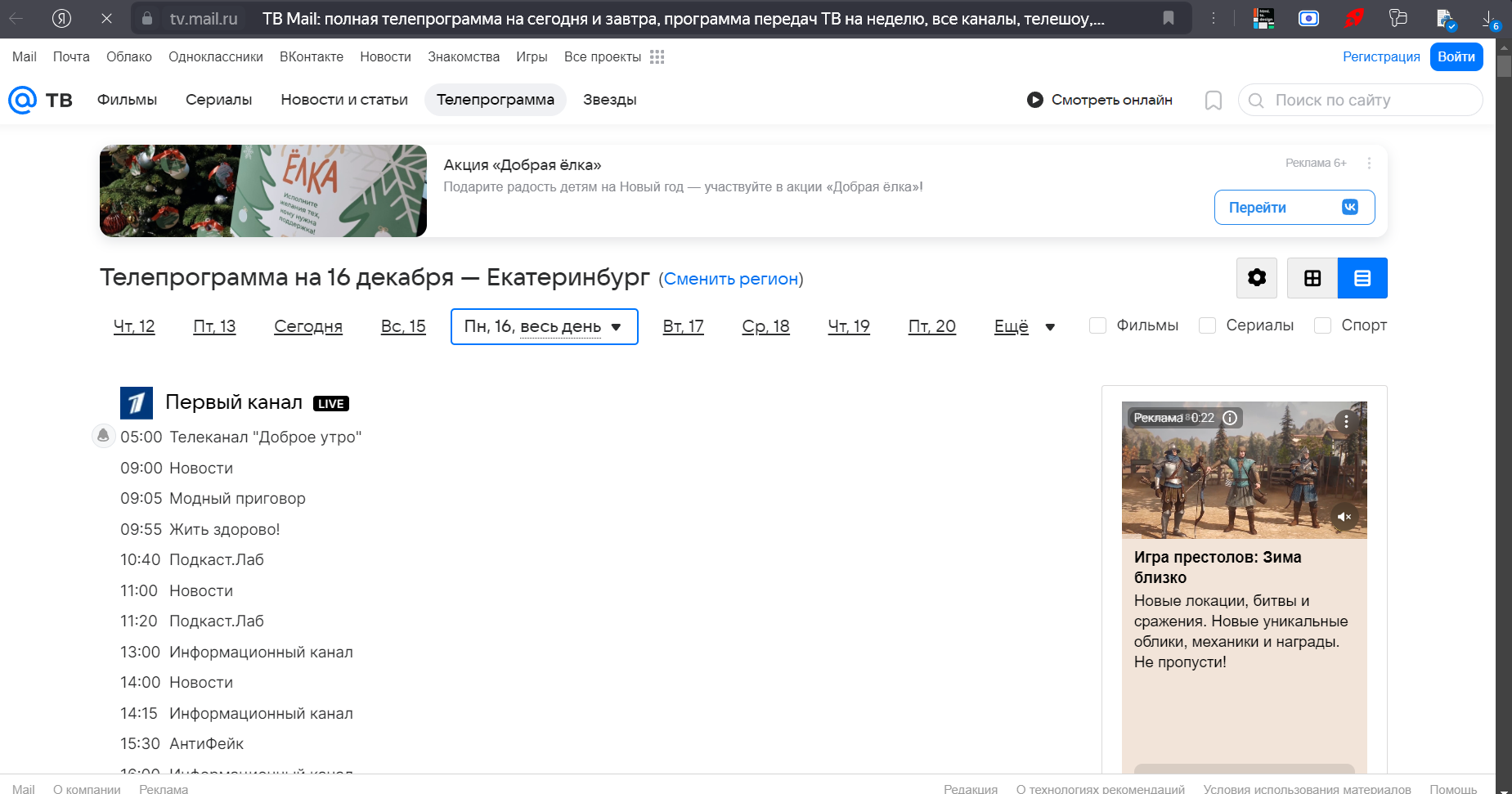
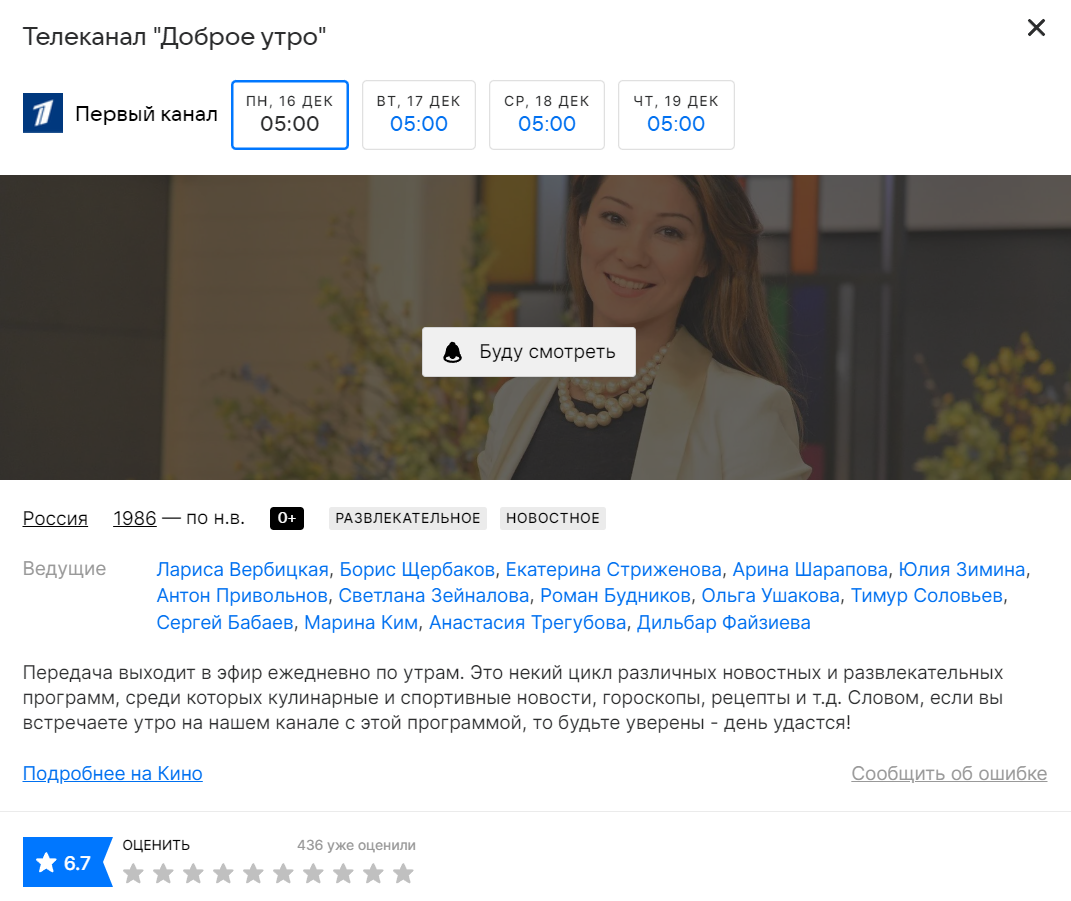
**1. ВВЕДЕНИЕ**

В данном проекте была поставлена задача разработки модели машинного обучения для прогнозирования рейтинга телеканалов на основе данных их телепрограмм, включающих различные характеристики, такие как возрастные ограничения, список ведущих, категории и др. Для достижения этой цели были выполнены следующие шаги: парсинг данных телепрограммы, предобработка данных, выбор и обучение модели машинного обучения, а также оценка ее эффективности.

**2. ОСНОВНАЯ ЧАСТЬ**

2.1 Решение задачи парсинга телепрограммы

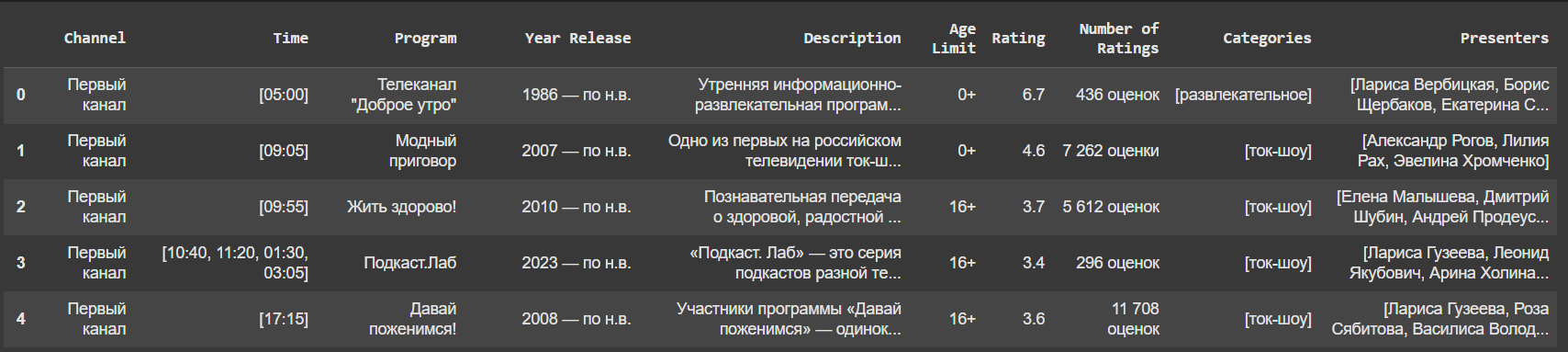
Для получения данных о телепрограммах был использован парсинг веб-страниц. С помощью библиотеки BeautifulSoup были извлечены данные о каналах, времени показа, названиях программ и других характеристиках. Парсинг осуществлялся с сайта <https://tv.mail.ru/ekaterinburg/>. Выбор пал на данный сайт по той причине, что он имеет рейтинговую систему - пользователи данного сайта могут формировать рейтинги телепередач путем оценивания от 0 до 10.



Данный сайт имеет ограниченный список дат для просмотра телепрограммы, поэтому было принято решение распарсить телепрограмму на 1 неделю.

2.2. Парсинг открытых источников

В процессе парсинга использовались открытые источники данных, доступные на сайте <https://tv.mail.ru/ekaterinburg/>. Эти данные были сохранены в формате CSV для дальнейшего анализа и использования в модели машинного обучения.

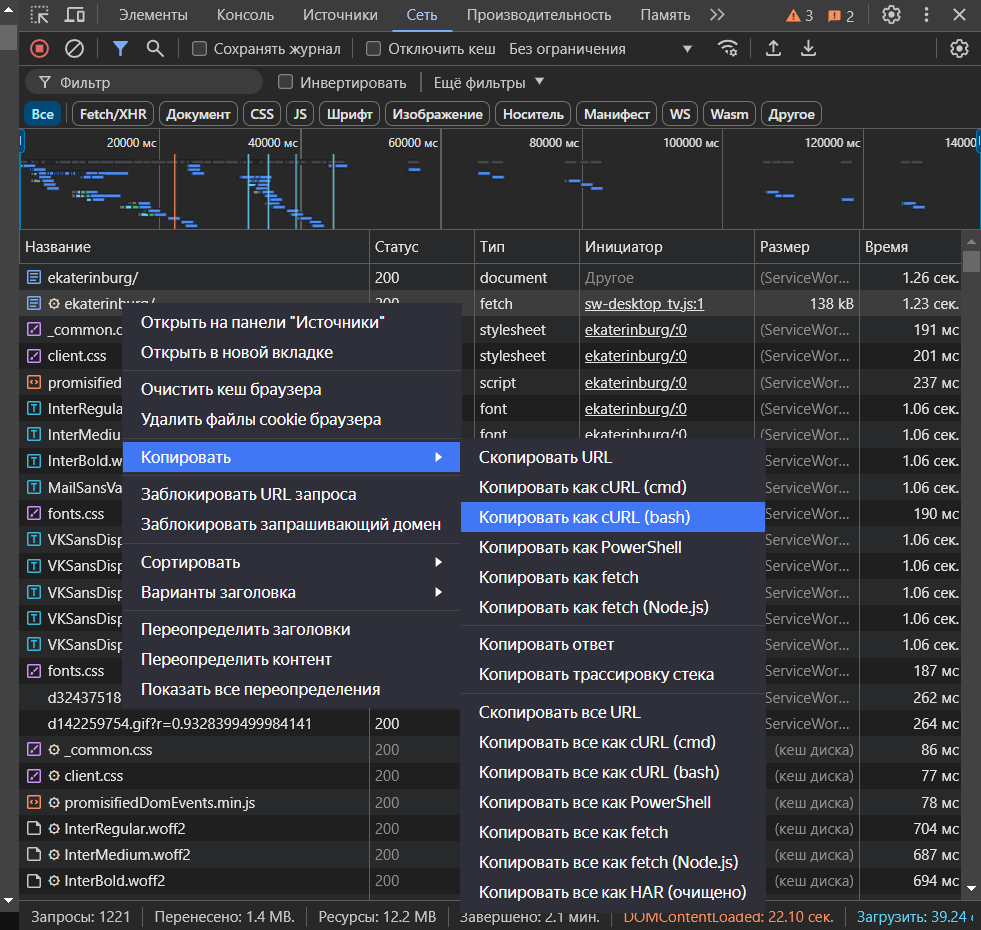
Структура извлекаемых данных 

*(каждая строка - это подробная информация о различных телепрограммах)*

2.2 (дополнение) Скрипт, выполняющий HTTP-запрос к веб-сайту

Данный скрипт отвечает за настройку и выполнение HTTP-запроса к веб-сайту с использованием определенных куки и заголовков. Это важно для обеспечения корректного взаимодействия с сервером, особенно если сайт требует аутентификации или определенных настроек сессии.

Инструкция по формированию скрипта:

1. Открыть веб-сайт <https://tv.mail.ru/ekaterinburg/> и выбрать дату телепрограммы из предоставленного списка (выбрать параметр отображения - весь день).
2. Открыть код элемента, выбрать пункт “Сеть” и обновить страницу. В данной вкладке отображаются запросы, которые производит веб-страница.
3. Повторить действия на скриншоте ниже:
4. Конвертировать скопированную *curl команду* в скрипт на Python с помощью веб-сервиса <https://curlconverter.com/>

2.3. Анализ методов машинного обучения и обоснование выбора

Для решения задачи прогнозирования рейтинга была выбрана модель линейной регрессии, а именно Ridge Regression. Этот метод был выбран из-за его способности справляться с мультиколлинеарностью (сильная зависимость входных переменных между собой) и предотвращать переобучение за счет регуляризации. Ridge Regression позволяет учитывать большое количество признаков, что было важно в условиях наличия как числовых, так и категориальных данных.

2.4. Описание реализации модели машинного обучения

Для реализации модели использовались библиотеки pandas, scikit-learn, nltk и другие. Данные были разделены на обучающую и тестовую выборки. Для предобработки данных использовались StandardScaler для числовых признаков и OneHotEncoder для категориальных. Модель Ridge Regression была обучена на обучающей выборке, а затем протестирована на тестовой. Были рассчитаны метрики оценки модели: среднеквадратичная ошибка (MSE), средняя абсолютная ошибка (MAE) и коэффициент детерминации (R²).

Мне удалось достичь следующих результатов:

* Среднеквадратичная ошибка (MSE): 0.322

MSE измеряет среднее значение квадратов ошибок, то есть разницу между предсказанными и фактическими значениями. Чем меньше значение MSE, тем лучше модель предсказывает данные.

* Средняя абсолютная ошибка (MAE): 0.316

MAE измеряет среднее значение абсолютных ошибок. Это более интерпретируемая метрика, так как она показывает среднюю величину ошибки в тех же единицах, что и данные.

* Коэффициент детерминации (R²): 0.8008

R² показывает, какая доля дисперсии зависимой переменной объясняется моделью. Значение R² варьируется от 0 до 1, где 1 означает, что модель идеально объясняет дисперсию данных. В данном случае R² равен 0.8008, что указывает на то, что модель объясняет около 80% вариации в данных.

2.5. Программная реализация прогнозной модели

Программная реализация включала в себя предобработку данных и обучение модели. Код был написан на языке Python с использованием Google Colab Notebook. В результате работы модели были получены предсказанные рейтинги для всех передач в тестовом наборе данных, которые затем были добавлены в DataFrame для дальнейшего анализа.

2.6. Демонстрация решения

Итоговые результаты были продемонстрированы с помощью визуализации фактических и предсказанных рейтингов каналов. Для подсчета рейтинга каналов (актуальных / предсказанных) брались средние значения рейтингов их телепередач. Для демонстрации результатов использовалась библиотека Seaborn, которая позволила наглядно представить результаты работы модели. Также была проведена оценка важности признаков, что позволило определить, какие из них наиболее сильно влияют на предсказательную способность модели.

Ознакомиться с решением задачи парсинга, сформированными датасетами, построением прогнозной модели рейтинга телеканалов можно по [ссылке](https://drive.google.com/drive/folders/1RdrXmGLQdQpCvLZdqH-jepBPRuhOm78h?usp=sharing).