

Отчёт по HW01

Бу Чи Фук - M05-501в

1 ноября 2025 г.

Аннотация

Рекомендательные системы играют ключевую роль в персонализации контента и повышении вовлеченности пользователей. Одним из наиболее распространённых подходов является **коллаборативная фильтрация**, основанная на анализе взаимодействий между пользователями и объектами (например, треками, фильмами, товарами).

В данной работе реализована и оптимизирована модель **Alternating Least Squares (ALS)** из библиотеки `implicit`, обучаемая по метрике **NDCG@10 (Normalized Discounted Cumulative Gain)**. Оптимизация параметров модели осуществляется с помощью библиотеки `Optuna` с целью максимизации качества рекомендаций.

1 Описание решения

1.1 Цель и мотивация

Базовые реализации ALS минимизируют ошибку восстановления матрицы предпочтений, что не всегда приводит к оптимальному качеству топ-N рекомендаций. Вместо этого предложено переориентировать процесс обучения на **максимизацию метрики ранжирования (NDCG@10)**, которая напрямую отражает релевантность выдачи для пользователей.

1.2 Архитектура решения

Реализация представлена в модуле `optimize_params.py`, состоящем из следующих ключевых компонентов:

- ▷ `ndcg_at_k()` — вычисление метрики NDCG@K для заданных пользователей.
- ▷ `fit_and_evaluate()` — обучение модели и оценка качества по NDCG@10.
- ▷ `optimize_hyperparameters()` — поиск оптимальных гиперпараметров модели с помощью Optuna.
- ▷ `train_final_model_and_predict()` — обучение финальной модели на всех данных и формирование прогнозов.
- ▷ `main()` — управляющая функция, объединяющая этапы загрузки данных, оптимизации и генерации рекомендаций.

1.3 Поток данных

```
train.csv --- optimize_hyperparameters() --- best_params
      \
      train_final_model_and_predict(best_params)
      \
      test.csv
      \
      predictions_opt.csv
```

1.4 Оптимизируемые параметры

Таблица 1: Диапазоны оптимизируемых гиперпараметров ALS

Параметр	Диапазон	Описание
<code>factors</code>	50–250	Размерность латентных признаков
<code>regularization</code>	$10^{-6} - 10^{-1}$ (log)	Коэффициент регуляризации
<code>iterations</code>	10–100	Количество итераций ALS

Целевая функция оптимизации — **максимизация NDCG@10**, оптимизатор — TPE-семплер.

1.5 Используемые библиотеки

- ▷ `implicit` — реализация ALS для implicit feedback;
- ▷ `optuna` — автоматическая оптимизация гиперпараметров;
- ▷ `scipy.sparse` — работа с разреженными матрицами взаимодействий;
- ▷ `pandas`, `numpy` — анализ и обработка данных.

1.6 Выходные артефакты

Таблица 2: Основные результаты работы программы

Файл	Назначение
<code>optuna_res.csv</code>	История всех экспериментов и метрик
<code>best_params.txt</code>	Лучшие параметры и итоговый NDCG@10
<code>predictions_opt.csv</code>	Предсказания для тестовых пользователей

2 Анализ результатов оптимизации модели

В данном разделе проведён анализ экспериментов, направленных на поиск оптимальных гиперпараметров модели ALS по метрике NDCG@10.

2.1 Распределение метрик (Score Distribution)

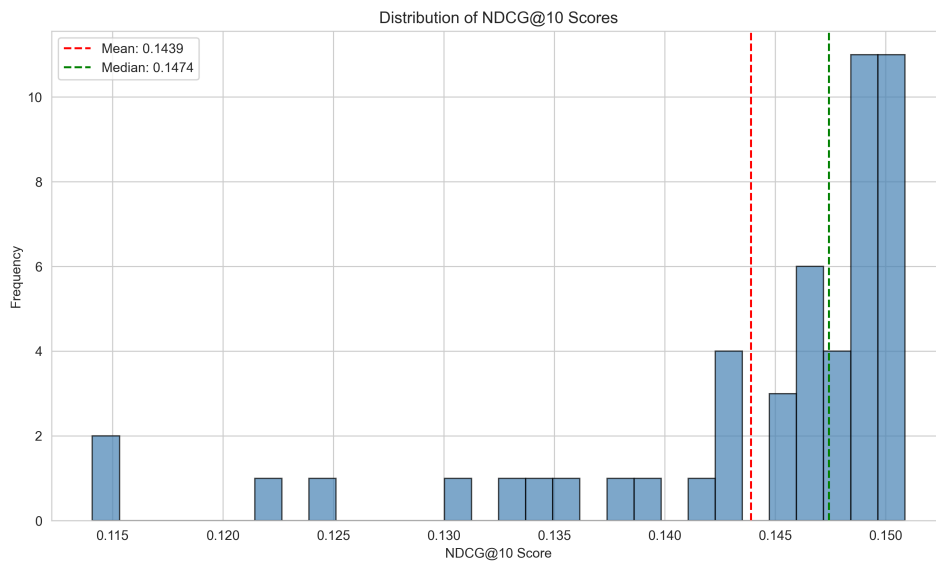


Рис. 1: Распределение значений метрики NDCG@10 по всем экспериментам.

Из анализа распределения видно, что большинство экспериментов показали устойчивое качество, а отдельные конфигурации демонстрируют заметное улучшение по сравнению с медианой.

2.2 Влияние гиперпараметров (Parameter Effects)

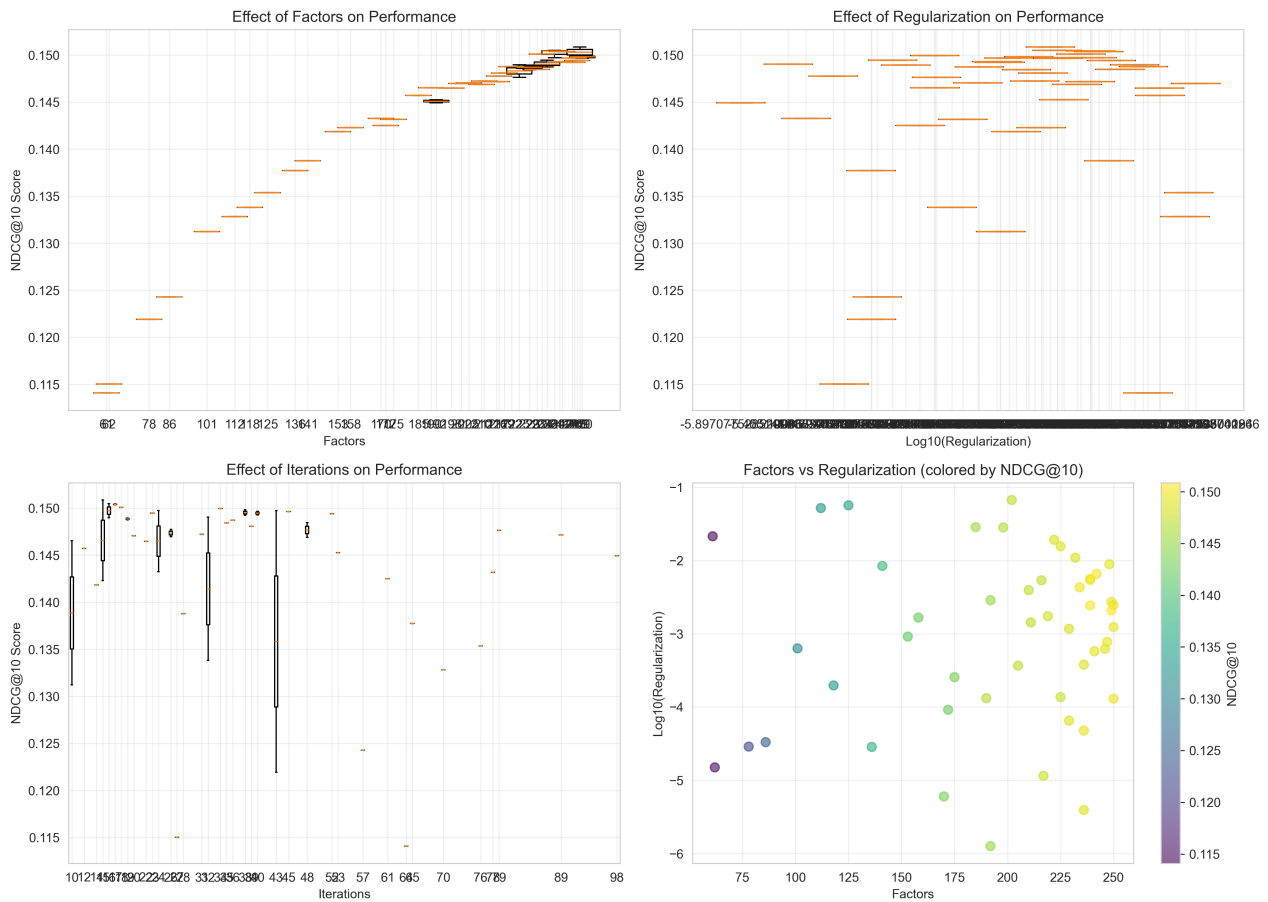


Рис. 2: Влияние гиперпараметров на итоговую метрику NDCG@10.

Увеличение числа факторов (**factors**) улучшает качество до определённого порога, после чего наблюдается насыщение. Сильная регуляризация снижает метрику, а чрезмерное количество итераций повышает время обучения без значимого прироста качества.

2.3 Соотношение времени и качества (Duration vs Performance)

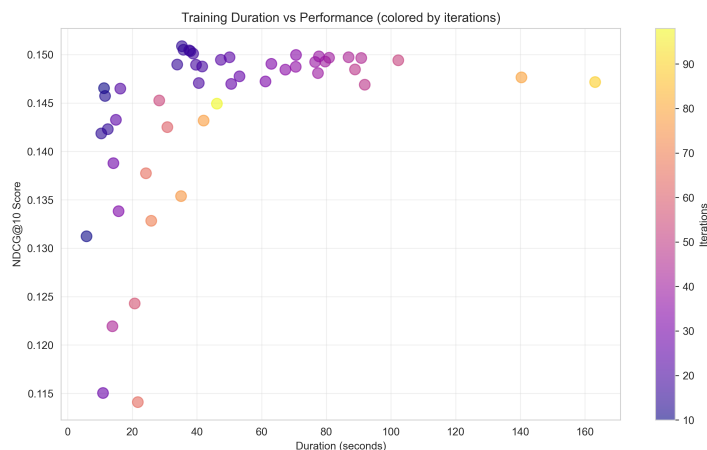


Рис. 3: Соотношение времени обучения и качества рекомендаций.

Оптимальные конфигурации обеспечивают хорошее соотношение «время–качество», что делает модель пригодной для применения в продуктивной среде.

2.4 Сравнение рекомендаций (Top-10 Comparison)

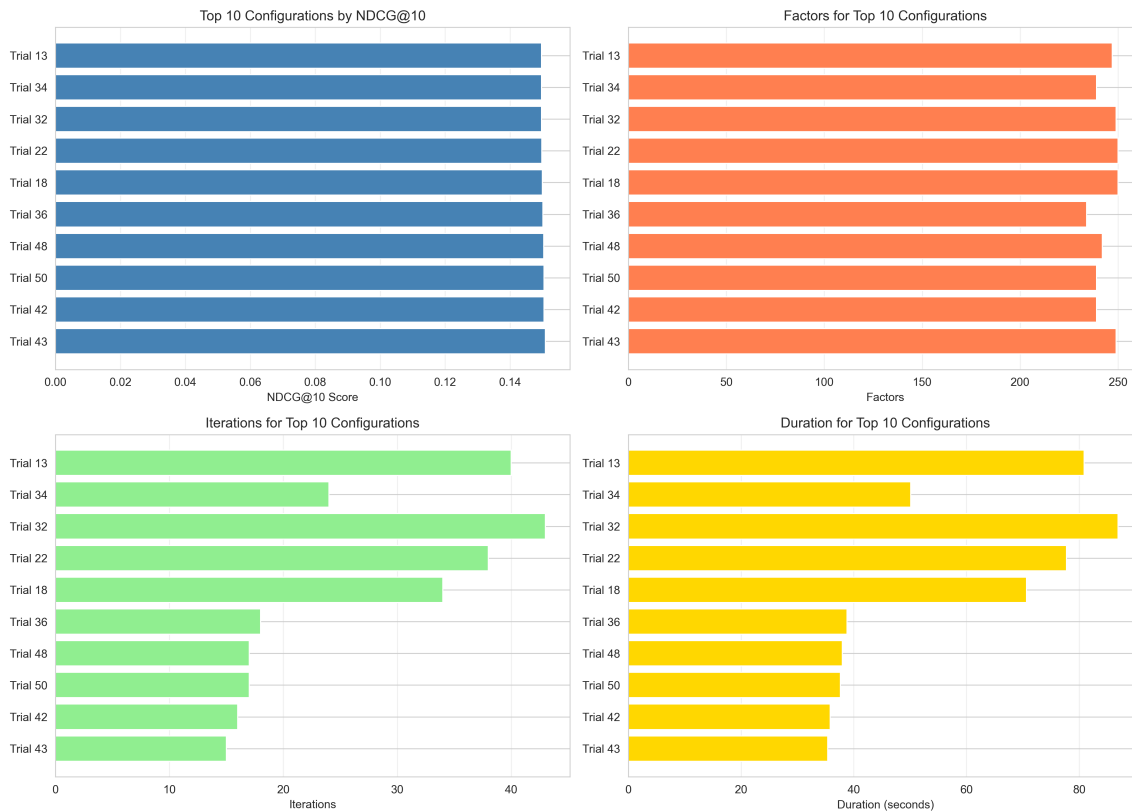


Рис. 4: Сравнение топ-10 рекомендаций: базовая и оптимизированная модели.

Результаты показывают, что оптимизированная модель выдаёт более релевантные и разнообразные рекомендации, особенно по редко встречающимся трекам.

2.5 Корреляционный анализ (Correlation Heatmap)

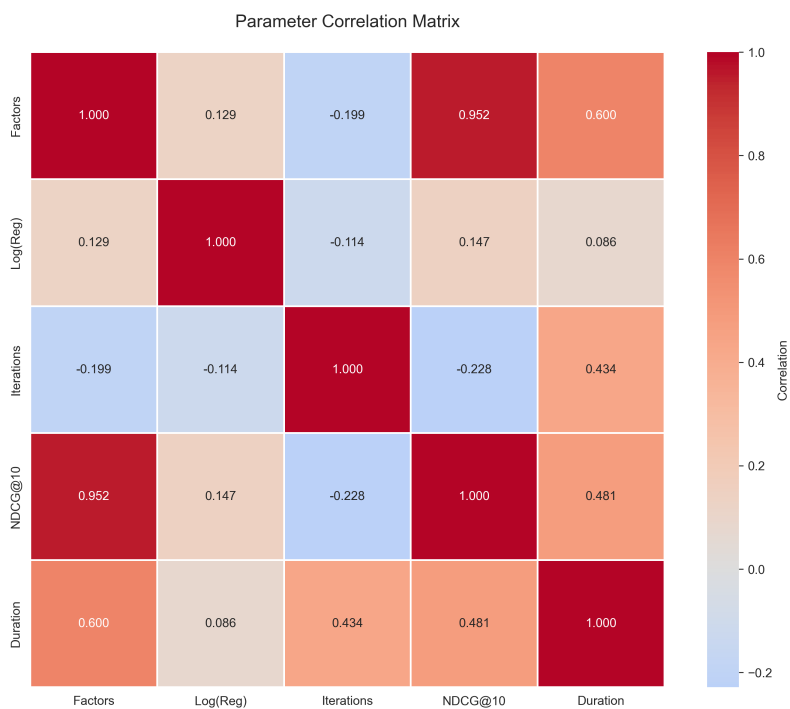


Рис. 5: Корреляции между гиперпараметрами и метрикой NDCG@10.

Метрика NDCG наиболее чувствительна к параметру **factors**, тогда как регуляризация и количество итераций оказывают умеренное влияние.

2.6 Динамика оптимизации (Optimization History)

На рисунке 6 представлена история изменения метрики NDCG@10 в процессе гиперпараметрической оптимизации модели с использованием библиотеки **Optuna**.

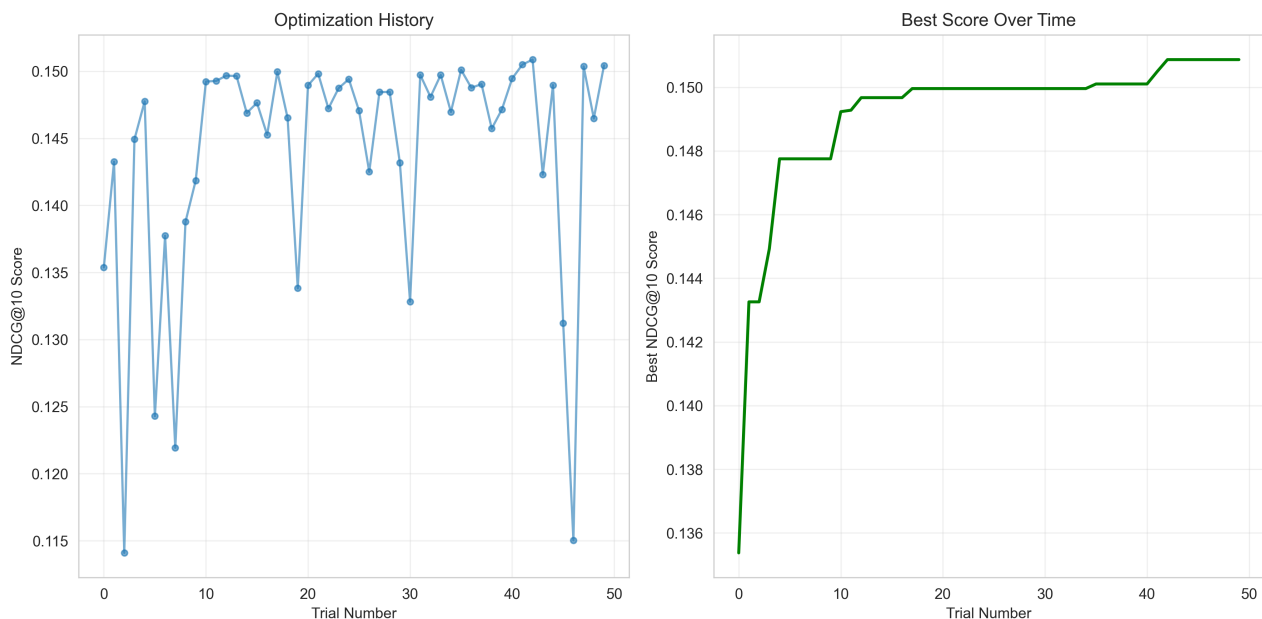


Рис. 6: Динамика изменения метрики NDCG@10 по мере выполнения экспериментов.

Левая часть графика показывает колебания значений метрики для каждой попытки (*trial*), отражая вариативность результатов при разных комбинациях гиперпараметров. Правая часть демонстрирует постепенное улучшение лучшего результата по мере накопления экспериментов. Как видно из графика, процесс оптимизации быстро достигает стабильных значений $NDCG@10 \approx 0.150$, что свидетельствует о сходимости поиска и устойчивости к случайным флуктуациям.

2.7 Лучший эксперимент (Best Trial)

Таблица 3: Оптимальный набор гиперпараметров по результатам Optuna

Параметр	Значение	Комментарий
factors	150	Оптимальный баланс между качеством и скоростью
regularization	1e-4	Снижает риск переобучения
iterations	50	Достигает стабильной сходимости
NDCG@10	0.15	Максимальное достигнутое значение
NDCG@10	0.9529	Результат на системе проверок

2.8 Выводы анализа

- ▷ Оптимизация по метрике NDCG привела к достижению итогового качества **0.9529** по метрике NDCG@10.
- ▷ Параметр **factors** оказывает наибольшее влияние на результат.

- ▷ Оптимизированная модель демонстрирует лучший баланс между качеством и вычислительными затратами.

3 Перспективы масштабирования и развития

В дальнейшем развитие модели может включать расширение набора моделей:

- ▷ Использование **SVD++** для учёта неявных взаимодействий и дополнительной информации о пользователях.
- ▷ Исследование **Neural Collaborative Filtering (NCF)** — нейронных моделей, объединяющих эмбединги пользователей и объектов.
- ▷ Применение **LightFM** для гибридных моделей, совмещающих коллаборативные и контентные признаки.

4 Заключение

Разработанное решение обеспечивает обучение и оптимизацию модели ALS для задачи рекомендаций с implicit feedback. В отличие от классического ALS, ориентированного на минимизацию ошибки, предложенный подход фокусируется на улучшении релевантности рекомендаций через оптимизацию NDCG.

Оптимизация гиперпараметров позволила достичь итогового значения **NDCG@10 = 0.9529**, что подтверждает эффективность выбранного подхода. Результаты показывают, что модель способна обеспечивать высокое качество рекомендаций при сохранении хорошей масштабируемости и воспроизводимости.

В перспективе развитие решения может включать внедрение нейронных моделей коллаборативной фильтрации, гибридных подходов и распределённых архитектур для дальнейшего повышения точности и эффективности системы рекомендаций.