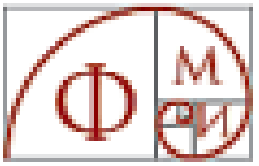
Софийски университет „Св. Кл. Охридски”

Факултет по математика и информатика

Курсов Проект

на тема:

„Препоръчваща система за книги”



Изготвено от: Надежда Францева Ф.Н. 8MI3400357

Курс: „ИИ, I курс“, Учебна година: 2023/2024

Преподаватели: **д-р Милен Чечев**

Съдържание

1 Увод

Проектът има за цел да направи препоръки за книги, базирани на оценки, дадени от потребители. Използвайки алгоритми за колаборативно филтриране, сравнени с хибридния матричен факторизиращ модел LightFM, проектът ще помогне на читателите да намерят най-подходящите за тях книги, осигурявайки по-приятно и обогатяващо прекарване на свободното време.

2 Преглед на областта

За проекта са използвани различни алгоритми за колаборативно филтриране, както и хибридния матричен факторизиращ модел LightFM. Колаборативното филтриране е един от най-широко използваните методи за изграждане на препоръчващи системи, като основната му идея е да прави препоръки въз основа на сходства между потребителите или между елементите (в случая - книги).

В този проект са разгледани и оценени следните алгоритми:

- **Slope One**: Прост и ефективен алгоритъм, който изчислява прогнозни рейтинги на база средните разлики в оценките между елементите.
- **CoClustering**: Алгоритъм, който разделя потребителите и елементите в клъстери, като извършва препоръки въз основа на тези клъстери.
- **SVD (Singular Value Decomposition)**: Алгоритъм, който използва матрична факторизация за намиране на латентни фактори, които описват предпочитанията на потребителите и характеристиките на елементите.
- **SVD++**: Разширение на SVD, което взема предвид не само оценките, но и липсващите оценки на потребителите.
- **LightFM**: Хибриден матричен факторизиращ модел, представящ потребители и елементи като линейни комбинации от латентните фактори на характеристиките на тяхното съдържание. Използва различни техники за максимизиране на точността на препоръките, включително WARP (Weighted Approximate-Rank Pairwise).

Всеки от тези алгоритми беше тестван и сравнени по показатели като средна квадратна грешка (RMSE) и средна абсолютна грешка (MAE). Алгоритъмът с най-ниска грешка бе избран като най-подходящ за препоръчващата система.

3 Проектиране

За реализиране на системата е използван следният модел на данни: https://www.kaggle.com/datasets/zygmunt/goodbooks-10k . Пускат се различните алгоритми, като се изчисляват техните времена за обучение, за обработка на данните, както и техните грешки (*RMSE, MAE*). Резултатите се показват в табличен вид. Правят се няколко експеримента с различна стойност за regularization term. От резултатите се

правят изводи за най-оптимален алгоритъм, като се гледат стойностите на грешките. Избира се въпросният алгоритъм и се извеждат най-добрите предложения за книги, които да бъдат препоръчани на всеки от потребителите.

4 Реализация, тестване/експерименти

4.1 Използвани технологии, платформи и библиотеки

Проектът е реализиран под формата на *Colab Notebook* - https://colab.research.google.com/drive/1iNdDvF4sxdkC8wWyGNPG990oPSoiFa_5?usp=sharing
Използвани са данни от *Kaggle*.
За предварителна обработка се използват *Pandas* и *Numpy*.
За анализ на алгоритмите се използва библиотеките *Surprise и LightFM*.

4.2 Реализация/Провеждане на експерименти

В *Colab Notebook*-а, където е реализиран проекта, върху данните от *Kaggle* за оценки от потребители за различни типове места се прави предварителна обработка посредством *Pandas* и *Numpy*. Направено е сравнение между 4 алгоритъма - *SlopeOne, CoClustering, SVD, SVDpp*. Критериите за сравнение са относителната квадратична грешка, абсолютната грешка, както и времето за изпълнение. Във времето за изпълнение се включват времената за обучение (трениране) и тези за упражняване на наученото върху тестовия набор от данни. (Табл. 1)

Collaborative Filtering Algorithms	RMSE	MAE	Time
SlopeOne	0.906	0.691	0:02:31
CoClustering	0.873	0.668	0:04:19
SVD	0.844	0.66	0:02:54
SVDpp	0.837	0.651	0:10:49

Табл. 1 – резултати от сравнение на алгоритмите (default reg term)

Обучителното и тестовото множество са направени на принципа на крос валидацията от цялото множество. Експерименти се провеждат на база на различни стойности на *regularization term – default* (0.02), 0.1, 0.5 и 0.85.

Collaborative Filtering Algorithms	RMSE	MAE	Time
SlopeOne	0.906	0.691	0:02:15
CoClustering	0.873	0.668	0:03:42
SVD	0.839	0.658	0:02:43
SVDpp	0.835	0.654	0:10:47

Табл. 2 – резултати от сравнение на алгоритмите (reg term = 0.1)

Collaborative Filtering Algorithms	RMSE	MAE	Time
SlopeOne	0.906	0.691	0:02:16
CoClustering	0.873	0.667	0:03:45
SVD	0.85	0.676	0:02:42
SVDpp	0.848	0.674	0:10:39

Табл. 3 – резултати от сравнение на алгоритмите (reg term = 0.5)

Collaborative Filtering Algorithms	RMSE	MAE	Time
SlopeOne	0.906	0.691	0:02:15
CoClustering	0.874	0.669	0:03:45
SVD	0.864	0.69	0:02:38
SVDpp	0.862	0.688	0:10:30

Табл. 4 – резултати от сравнение на алгоритмите (reg term = 0.85)

От получените резултати Collaborative Filtering Algorithms разбираме, че за конкретните данни най-добре би сработил *SVD++* алгоритъма. Използвайки него, можем да изведем най-добри препоръки за всеки от потребителите. Примери за някои от тях :

```
24383 ['Gates of Fire: An Epic Novel of the Battle of
Thermopylae', 'Batman: The Killing Joke', 'The Invention of
Hugo Cabret', 'Avatar: The Last Airbender (The Promise, #1)',
'Avatar: The Last Airbender (The Promise, #2)']
```

Резултатите могат да се прочетат като:
На потребител с ID **24383** би било най-подходящо да се предложи следната книга: **Gates of Fire: An Epic Novel of the Battle of Thermopylae (Огнените порти: Епичен роман за битката при Термопилите)**, по-малко, но доста вероятно е той да хареса препоръки за книгите: **Batman: The Killing Joke (Батман: Убийствената шега)** и **The Invention of Hugo Cabret (Изобретението на Хюго Кабре)**, **Avatar: The Last Airbender (Аватар: Последният повелител на въздуха) 1 и 2** също биха представлявали интерес за него.

По-късно добавихме и хибридния матричен факторизиращ модел **LightFM и** получихме следните препоръки:

```
Препоръчани книги за потребител с ID 24383:
1. Jasper Jones
2. The Taming of the Queen (The Plantagenet and Tudor Novels,
#11)
3. Hope Was Here
4. The King's Buccaneer (Krondor's Sons, #2)
```

```
5. The Element: How Finding Your Passion Changes Everything
6. Palace of Stone (Princess Academy, #2)
7. Bloody Jack (Bloody Jack, #1)
8. The Pleasure of My Company
9. The River Why
10. A Little Something Different
```

Използвайки него, получаваме следните стойности на грешките:

```
RMSE: 4.412217503544822
MAE: 4.190378279120978
```

LightFM е хибриден модел, който използва както съдържателна информация, така и колаборативно филтриране. Ако не е налична достатъчно съдържателна информация, това може да повлияе на ефективността му.

5 Заключение

От направеното проучване можем да заключим, че за конкретния набор от данни, най-добре работи алгоритъмът *SVD++*. Той дава най-малка стойност на *RMSE* и *MAE* грешките. Най-малка грешка означава най-голяма вероятност препоръката на системата да се хареса на потребителя.

Като идея за по-нататъшно развитие е намирането на още корпуси от данни, както и прилагането на алгоритми за конкретизиране на препоръката до точно определена книга.

6 Използвана литература

[1]https://www.kaggle.com/gspmoreira/recommender-systems-in-python-101
[2]https://learn.fmi.uni-sofia.bg/pluginfile.php/498129/mod_resource/content/1/Recommender.Systems_Handbook.pdf

[3]https://learn.fmi.uni-sofia.bg/pluginfile.php/498130/mod_resource/content/1/practical_rec_sys_2019.pdf

[3]https://surprise.readthedocs.io/en/stable/index.html

[4]https://making.lyst.com/lightfm/docs/home.html