

ANIME

Recommender System

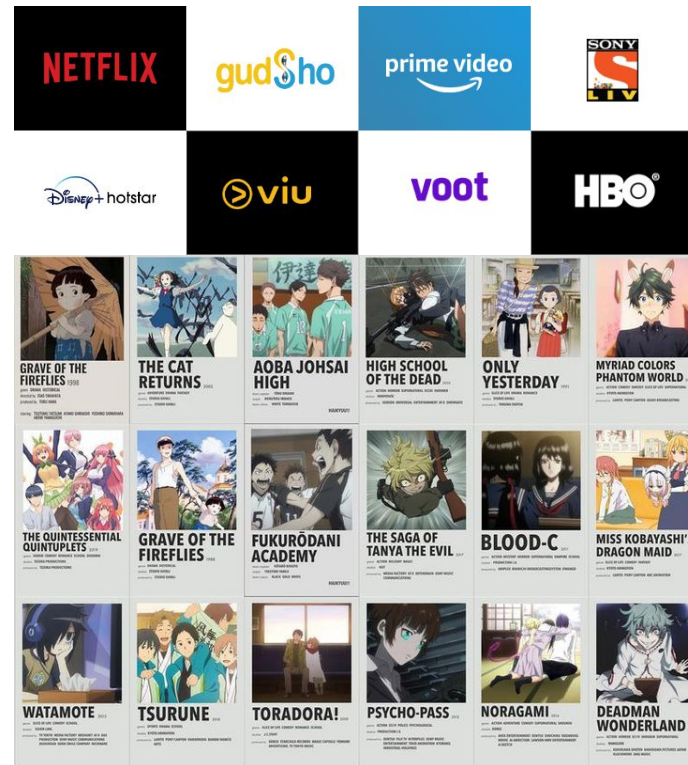
Recommender System
Juniarto Kurniawan - Batch 9

Outline

- Introduction/Background
- Analysis
- Conclusion
- References

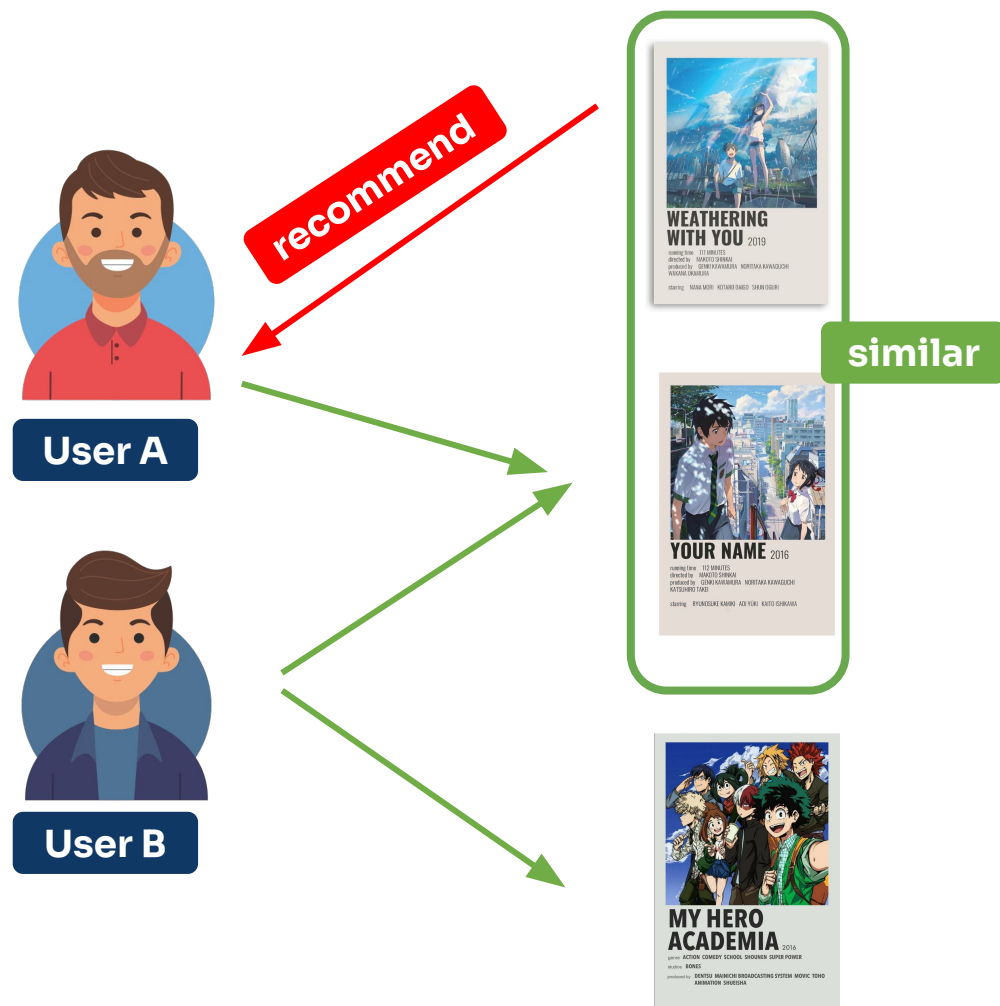
Introduction

Business Problem



- Pilihan banyaknya anime baik judul maupun genre banyak tersedia dalam layanan OTT tersebut. Namun perlu diingat bahwa user memiliki keterbatasan waktu untuk mencari anime yang cocok dengan seleranya pada saat ingin menontonnya.
- Solusi yang digunakan adalah memberikan rekomendasi yang tepat
- Pilihan rekomendasi yang tepat tersebut dapat memberikan kenyamanan dan kemudahan bagi user untuk menonton anime sehingga dapat meningkatkan engagement pada layanan OTT untuk tetap berlangganan dan pada ujungnya dapat meretain atau bahkan meningkatkan revenue perusahaan OTT.

Item Based – Collaborative Filtering

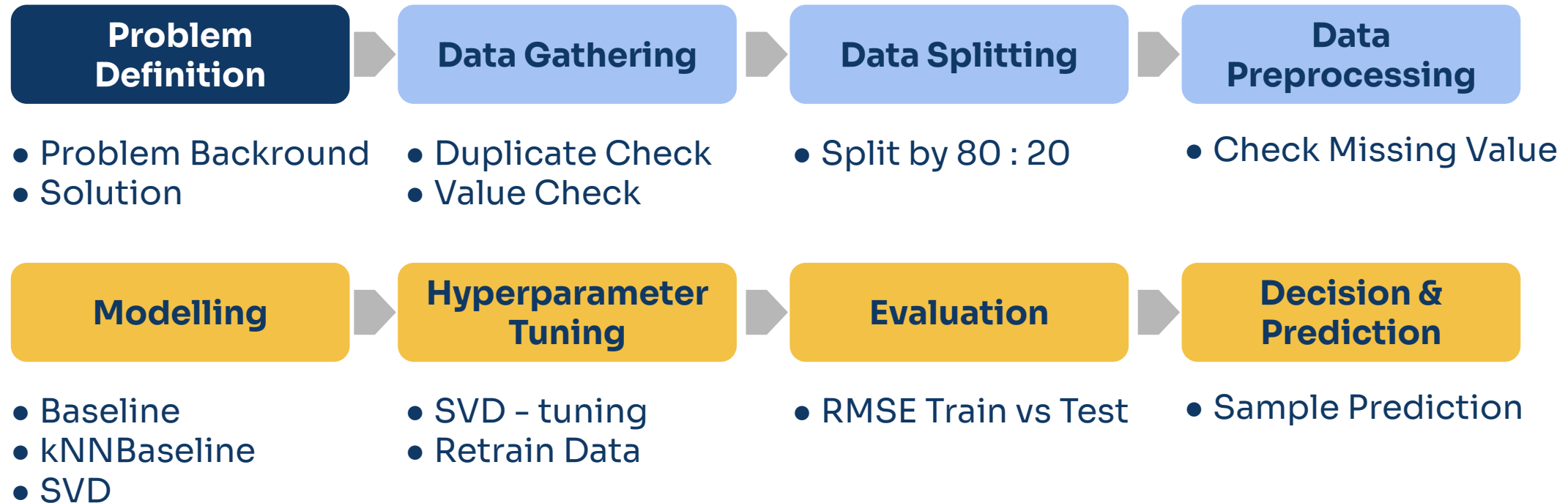


Sebagai contoh anime Your Name telah di rating oleh User A dan B, Anime Your Name ini mempunyai kemiripan dengan Anime Weathering with You dan anime tersebut belum di tonton oleh user A sehingga user A mendapat Rekomendasi tontonan anime tersebut.

Item-based ini akan kita gunakan dalam pemodelan kNN

Analysis

Workflow



Data Gathering

MyAnimeList

Data rating

- anime_id
- user_id
- rating

Data Anime

- anime_id
- name
- genre
- dll

- Data yang sebenarnya sebanyak 7,8jt namun karena keterbatasan hardware komputasi maka akan digunakan 20% data yang diambil secara random.
- Isian rating terdapat angka “-1” yang artinya bahwa user menonton anime tersebut namun tidak memberikan rating. Oleh sebab itu data tersebut dapat dieliminasi terlebih dahulu sehingga data net menjadi sebanyak 1,26 jt.

	user_id	anime_id	rating
0	1	20	-1
1	1	24	-1
2	1	79	-1
3	1	226	-1
4	1	241	-1

anime_id	name	genre
32281	Kimi no Na wa.	Drama, Romance, School, Supernatural
5114	Fullmetal Alchemist: Brotherhood	Action, Adventure, Drama, Fantasy, Magic, Mili...
28977	Gintama°	Action, Comedy, Historical, Parody, Samurai, S...
9253	Steins;Gate	Sci-Fi, Thriller
9969	Gintama'	Action, Comedy, Historical, Parody, Samurai, S...

Data Splitting

Dalam mempermudah proses kalkulasi untuk data splitting hingga modelling kita gunakan package Surprise, nantinya package tersebut membuat matrix Utility serta membagi data menjadi Full data, Trainset, dan Testset.

```
# Load library for deep copy
import copy
# Split the data dengan porsi 20% data test dan random state 42
full_data, train_data, test_data = train_test_split(utility_data,
                                                    test_size = 0.2,
                                                    random_state = 42)
```

Data Train

1.014.232

80%

Data Test

253.559

20%

Data Preprocessing

Missing value yang butuh mendapat perhatian adalah user_rating, anime_id, dan user_id. Berdasarkan analisa tidak terdapat missing data pada data train maupun data test.

```
# Check Missing Value  
rating.isna().sum()
```

```
user_id      0  
anime_id     0  
rating       0  
dtype: int64
```

Modelling

Pada pemodelan ini digunakan 3 metode yaitu:

- Baseline Mean Prediction untuk model awal
- kNN dengan pearson similarity dan item-based collaborative
- Singular Value Decomposition

Berdasarkan pemodelan di atas dapat diketahui bahwa model yang cukup baik adalah model SVD oleh karena itu selanjutnya kita lakukan parameter tuning untuk memperoleh hasil yang optimal.

Model	RMSE
Baseline	1.572
kNN	1.289
SVD	1.281

Hyperparameter Tuning

Hyperparameter Candidate yang akan digunakan untuk tuning model SVD adalah sebagai berikut :

- Learning Rate (γ) terdiri dari beberapa nilai : [0.5,0.05,0.005]
- Jumlah Latent Factor beberapa nilai : [50,200]
- Regularization Strength beberapa nilai : [0.2,0.01,0.02]

Model	RMSE
SVD	1.281
SVDtuned	1.242

Hasil SVD setelah dilakukan tuning menjadi lebih baik daripada sebelum dilakukan tuning dengan best parameter : {'lr_all': 0.05, 'n_factors': 200, 'reg_all': 0.2}

Evaluation

Retrain Data

Setelah itu lakukan retrain data dengan menggunakan best parameter.

Evaluation

Lakukan evaluasi dengan membandingkan data train dengan data test.

```
# Hitung Prediksi
test_pred = model_best.test(test_data)
test_rmse = accuracy.rmse(test_pred)
test_rmse
```

```
RMSE: 1.2347
1.2347260606386212
```

```
# Summary RMSE
summary_test_df = pd.DataFrame({'Model' : ['SVD'],
                                  'RMSE-Tuning': [tuning_svd.best_score['rmse']],
                                  'RMSE-Test': [test_rmse]})

summary_test_df
```

SVDtuned

RMSE

Train

1.242

Test

1.234

Decision & Prediction

Decision

Berdasarkan RMSE tersebut maka model menghasilkan prediksi test yang baik sehingga kita dapat menggunakannya untuk memprediksi User.

Prediction

Untuk membuat prediksi langkah awal yang dilakukan adalah mencari unrated Anime untuk user yang terpilih. Setelah itu membuat prediksi dengan estimator dari model terbaik yang telah terpilih dan membuat prediksi ratingnya.

	user_id	anime_id	predicted_rating
8277	100	31540	9.773838
3537	100	4353	9.755825
2921	100	3269	9.743974
6850	100	18689	9.713090
2901	100	3231	9.643351

Decision & Prediction

Selanjutnya kita tambahkan keterangan anime_id berupa nama, dan genre dengan mengambil database "anime"

Dari gabungan dua data tersebut dapat kita resume, misalnya ingin menampilkan 5 Anime yang cocok untuk user 100.

```
# Generate 5 recommendation for user 100
get_top_highest_unrated(estimator=model_best,
                        k=5,
                        userid=100,
                        rating=rating,
                        anime=anime)
```

	user_id	anime_id	predicted_rating	name	genre
8277	100	31540	9.773838	Sekkou Boys	Comedy, Music
3537	100	4353	9.755825	Gakuen Heaven: Hamu Hamu Heaven	Comedy, Drama, Romance, School, Shounen Ai
2921	100	3269	9.743974	.hack//G.U. Trilogy	Action, Fantasy, Game, Sci-Fi
6850	100	18689	9.713090	Diamond no Ace	Comedy, School, Shounen, Sports
2901	100	3231	9.643351	Gunslinger Girl: Il Teatrino	Action, Drama, Military, Sci-Fi

Conclusion

Conclusion

Konklusi yang dapat diambil adalah, berdasarkan perbandingan 3 model pada Collaborative Filtering menghasilkan bahwa model SVD mempunyai nilai RMSE yang terkecil. Selain itu dengan model SVD yang telah dituning juga menghasilkan nilai yang kecil. Namun ada hal yang perlu diperhatikan bahwa SVD memerlukan review user untuk dapat menentukan rekomendasi, sehingga hal ini tidak terlalu fit untuk user baru.

Namun model ini dapat dijadikan apabila ingin maintain user dan dapat memberikan potensi apa bila user mereview dengan aktual dan akurat.

Selanjutnya dengan hardware yang memadai dapat dilakukan pemodelan pada total 7,8 juta data dan berbagai variasi hyperparameter. Sehingga hasil dapat lebih baik dan dapat mencerminkan data yang sebenarnya.

Reference

[Anime Recommendations Database | Kaggle](#)

[collaborative-filtering-python/collaborative-filtering-memory-based.ipynb at master · klaudia-nazarko/collaborative-filtering-python · GitHub](#)

[https://conferences.ittelkom-pwt.ac.id/index.php/centive/article/download/16/16](#)

Thank You
