

SISTEM REKOMENDASI LAGU METODE COLLABORATIVE FILTERING BERBASIS WEBSITE

MUSIC RECOMMENDATION SYSTEM USING COLLABORATIVE FILTERING WEBSITE-BASED

Muhammad Dhani Fawzan¹, Casi Setianingsih², Ratna Astuti Nugrahaeni³

^{1,2,3} Universitas Telkom, Bandung

dhanifawzan@student.telkomuniversity.ac.id¹, setiacasie@telkomuniversity.ac.id²,

ratnaan@telkomuniversity.ac.id³

Abstrak

Perkembangan teknologi mulai dirasakan. Termasuk dalam ruang lingkup industri musik. Kolaborasi musik dengan teknologi mulai dirasakan, mulai dari penyanyi yang mengembangkan genre *electro* dibantu dengan teknologi, media untuk memutar lagu dan sistem rekomendasi untuk pencarian. Rumah produksi musik, musisi, produser berlomba-lomba mengambil keuntungan seiring berkembangnya industri musik. Semakin maraknya industri musik maka akan terciptanya musik yang bervariasi dan beragam, maka dari itu diperlukan suatu sistem rekomendasi yang dapat memilih lagu yang terbaik dari sekumpulan lagu di beberapa genre sehingga orang dapat menikmati musik yang paling diminati saat ini. Hasil pengujian ini berjalan dengan baik dengan uji alpha sebesar 100% dan nilai RMSE pada sistem sebesar 1,788

Kata Kunci: *Collaborative filtering, Item-based filtering, rating based*

Abstract

Technological developments began to be felt. Included in the scope of the music industry. Music collaborations with technology are begun to rise, starting from singers who develop the electro genre assisted by technology, media to play songs and a recommendation system for searching. Music production houses, musicians, producers are competing to take advantage as the music industry develops. The more widespread the music industry will create varied and diverse music; therefore, we need a recommendation system that can choose the best song from a collection of songs in several genres so that people can enjoy the music that is most in demand today. The results of this test run well with an alpha test of 100% and an RMSE value of 1.788. on the system

Keywords: *Collaborative filtering, Item-based filtering, rating based*

1. Pendahuluan

Para penikmat lagu membutuhkan adanya sistem rekomendasi yang dapat mencari lagu yang sesuai dengan mereka sukai. Penikmat *genre* lagu yang beragam, kalangan umur yang berbeda-beda, *trend* musik yang berbeda-beda setiap zaman, ditambah dengan maraknya studio musik maupun musisi *indie* yang sering mengeluarkan *single* dan banyak hal lainnya menjadi faktor utama dibutuhkan sistem rekomendasi yang dapat memunculkan lagu-lagu terpopuler saat ini

Lagu dapat mengekspresikan perasaan manusia dan memberikan kenyamanan bagi penikmatnya. Maka pentingnya sebuah lagu tidak dapat dipandang sebelah mata lagi. Kemudian diciptakanlah sistem rekomendasi untuk melengkapi kebutuhan manusia tersebut. Metode yang dipakai dari sistem rekomendasi ini adalah *SVD*. Algoritma prediksi yang dimaksud dapat memunculkan hasil pengukuran dari respon variasi pengguna sistem rekomendasi. Metode algoritma ini termasuk golongan *item-based*, sehingga diharapkan sistem rekomendasi ini dapat memenuhi kebutuhan *user* untuk mencari lagu yang sesuai dan membantu promosi para musisi.

2. Landasan Teori

2.1 Collaborative Filtering

Collaborative Filtering adalah menyaring data yang berdasarkan terhadap *user experience* [1]. Lalu mengolahnya kembali untuk menjadi data yang dapat menjadi informasi baru terhadap user selanjutnya (X. Wu, Y. Huang, and S. Wang, 2017) [2]. Ide utamanya adalah untuk mengeksplorasi informasi mengenai perilaku di masa lampau maupun opini dari suatu komunitas pengguna yang kemudian digunakan untuk memprediksi *item* mana yang akan menarik bagi seorang pengguna (Thorat, Poonam B, 2015) [3]. Untuk item disini, penulis memakai dataset music dari Tujuan dari *collaborative filtering* untuk menutupi kelemahan *filter* sebelumnya, yaitu *content filtering*. Hal tersebut terjadi karena jika memakai *content filtering*, pengguna tidak dapat mendapatkan informasi dari pengguna lainya. Tujuan dari *filtering* ini adalah untuk menambahkan opini pengguna sehingga mendapatkan prediksi *item* baru yang mungkin akan diminati pengguna lainnya.

Algoritma yang digunakan beragam dari ruang vector model (VSM) dengan term frequency- inverse document frequency (TF-IDF) hingga model probabilistik dengan naive bayes atau decision tree. Content based tidak membutuhkan profil dari user sehingga tidak memiliki masalah cold start [5]. Collaborative filtering (CF) [6] adalah sebuah model rekomendasi yang menggunakan kedekatan antara user atau item dari pengumpulan data interaksi antara user dan item. Interaksi user-item yang dibangun akan membentuk sebuah matriks user-item. Collaborative Filtering ini nanti terbagi lagi menjadi dua, yaitu Memory Based dan Model-Based. Contoh algoritma yang digunakan pada memory-based collaborative filtering adalah pearson correlation. Algoritma kNearstNeighbors (k-NN) tetapi di algoritma ini terdapat permasalahan overfitting, scalability atau skalabilitas, sparsity atau sparsitas, dan cold start problem pada pengguna baru[7]. Sedangkan dengan model-based collaborative filtering menggunakan Singular Value Decomposition (SVD)[8], NMF, CoClustering dan lainnya dapat mengatasi beberapa permasalahan di atas walaupun mengurangi performa rekomendasi.

2.2 Item-Based Filtering

Item-based Filtering adalah teknik yang digunakan untuk memprediksi item yang mungkin diminati pengguna berdasarkan peringkat yang diberikan untuk item tersebut oleh pengguna lain yang memiliki selera yang sama dengan pengguna target. Item-based filtering menganggap bahwa selera pengguna terhadap suatu item atau barang akan cenderung sama dari waktu ke waktu. Ditambah lagi, pengguna yang menyukai suatu item biasanya juga akan menyukai item lain yang diminati oleh pengguna lain yang juga menyukai item yang sama dengan pengguna tersebut (Sarwar, 2001) [9].

2.3 Singular Value Decomposition

Singular Value Decomposition, merupakan salah satu teknik faktorisasi matriks, SVD pertama kali diperkenalkan oleh Beltrami dan Jordan pada tahun 1870 untuk dekomposisi matriks[10].

$$A = U\Sigma V^T \tag{2.1}$$

Matriks dapat direduksi dengan nilai pangkat k yang dapat membuat nilai matriks Σ menyesuaikan dengan pangkat k dan diubah dengan menjadi nilai tiga matriks faktorisasi yaitu $m \times k$, $k \times k$, dan $k \times n$, dimana $k < r$. SVD menyediakan peringkat k rendah yang tetap mempertahankan dari nilai asli m dengan memperkecil nilai singular k . Disebut dengan Σ_{\square} reduksi matriks karena Σ dilakukan dengan mempertahankan nilai Σ_{\square} . Matriks U dan V juga direduksi menghasilkan matriks U_{\square} dan V_{\square} diperoleh dengan menghapus $(r - k)$ kolom matriks U_{\square} dan matriks V_{\square} diperoleh dengan menghapus $(r - k)$ baris matriks V . Ketika matriks direduksi akan memperoleh matriks M_{\square} .

$$M_{\square} = U_{\square}\Sigma_{\square}V_{\square}^T \tag{2.2}$$

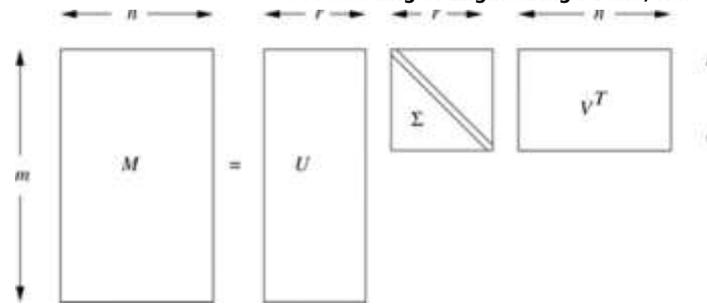
Sarwar [11] menunjukkan bahwa matriks yang lebih rendah dari ruang awal lebih baik karena penyaringan dari nilai-nilai kecil yang merupakan “noise” dalam relasi *user-item*.

Sebagai contoh seperti yang dapat dilihat pada rumus di atas, matriks M memiliki dimensi $m \times n$ dan *ranking* M adalah r . *Ranking* matriks adalah jumlah baris terbesar, atau kolom ekuivalen, yang dapat dipilih dan tidak ada kombinasi linier bukan nol dari baris yang merupakan vector semua-nol. Dengan kata lain, satu set baris atau kolom tersebut adalah independent. Sehingga matriks U , Σ dan V dengan properti berikut dapat dicari. Jika A matriks riil berukuran $m \times n$, maka terdapat matriks ortogonal.

$$U = [u_1, u_2, \dots, u_m] \in \mathbb{R}^{m \times m} \text{ dan } V = [v_1, v_2, \dots, v_n] \in \mathbb{R}^{n \times n}$$

Sedemikian sehingga, $U^tAV = [diag(\sigma_1, \sigma_2, \dots, \sigma_p)] \in \mathbb{R}^{m \times n}$, $p = \min \{m, n\}$ dengan $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_p \geq 0$ [11].

- U adalah matriks kolom-ortonormal $m \times r$ yang berarti masing-masing kolomnya adalah vector satuan dan perkalian titik dari dua kolom mana pun adalah 0
- V adalah matriks kolom-ortonormal $n \times r$. Perhatikan bahwa V selalu dalam bentuk yang dialihkan, sehingga baris V^T ortonormal.
- Σ adalah matriks diagonal, dimana semua elemen yang tidak berada pada diagonal utama adalah 0. Elemen-elemen Σ disebut nilai singular M .



Gambar 2.1 Struktur SVD

2.4 Root Mean Square Error

Root Mean Square Error atau RMSE merupakan persamaan yang digunakan untuk menghitung tingkat akurasi, dan juga merupakan satuan ukur yang digunakan untuk menilai keakuratan kinerja sistem rekomendasi [12]. Secara umum rumus RMSE akan ditunjukkan pada persamaan di bawah ini.

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (Y_i - \hat{Y}_i)^2}{n}} \quad (2.3)$$

Dengan,

- n = banyaknya entri rating
- Y_i = rating sebenarnya pada user- u terhadap item- i
- \hat{Y}_i = rating prediksi pada user- u terhadap item- i

3. Perancangan

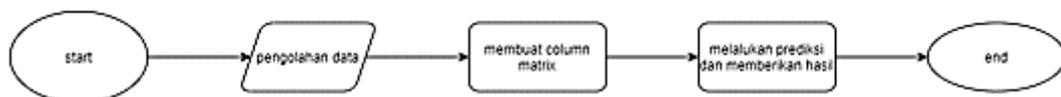
3.1 Gambaran Umum Sistem

Pada perancangan model yang dibangun secara umum, Sebelumnya perlu dibuatnya data menjadi column seperti dibawah ini:

Tabel 3.1 Data Matriks

	L00001	L00002	L00003	L00253
U00002	1	4	1		4
U00003	2	0	2		3
U00004	3	5	3		2
....					
U00031	5	4	2		3

Lalu selanjutnya dilakukan lah perhitungan dengan gambaran flowchart seperti berikut:



Gambar 3.1 Flowchart SVD

Untuk memberikan gambaran dari cara kerja algoritma SVD, diperlukan sampledata dari database aplikasi sistem rekomendasi musik yaitu:

$$\begin{matrix}
 & A & B & C & D & E \\
 U00043 & [1 & 1 & 1 & 0 & 0] \\
 U00044 & [3 & 3 & 3 & 0 & 0] \\
 U00045 & [4 & 4 & 4 & 0 & 0] \\
 U00046 & [5 & 5 & 5 & 0 & 0] \\
 U00047 & [0 & 2 & 0 & 4 & 4] \\
 U00048 & [0 & 0 & 0 & 5 & 5] \\
 U00049 & [0 & 1 & 0 & 2 & 2]
 \end{matrix}
 =
 \begin{bmatrix}
 0.13 & 0.02 & -0.01 \\
 0.41 & 0.07 & -0.03 \\
 0.55 & 0.09 & -0.04 \\
 0.68 & 0.11 & -0.05 \\
 0.15 & -0.59 & 0.65 \\
 0.07 & -0.73 & -0.67 \\
 0.07 & -0.29 & 0.32
 \end{bmatrix}
 \times
 \begin{bmatrix}
 12.4 & 0 & 0 \\
 0 & 9.5 & 0 \\
 0 & 0 & 1.3
 \end{bmatrix}$$

$$\times
 \begin{bmatrix}
 0.56 & 0.59 & 0.56 & 0.09 & 0.09 \\
 0.12 & -0.02 & 0.12 & -0.69 & -0.69 \\
 0.40 & -0.80 & 0.40 & 0.09 & 0.09
 \end{bmatrix}$$

Keterangan:

- A : Lagu *On The Hotline*
- B : Lagu *I Want to Know Your Plans*
- C : Lagu *F. U. R. B.*
- D : Lagu *I Got a Feeling*
- E : Lagu *Dantzig Station*

Dari *matrix* di atas, dapat diambil contoh sampel data yang berisikan *user* beserta lagu yang telah di-*rating* oleh *user*, ketika di kalkulasikan maka akan timbul persamaan di atas. Ini adalah persamaan dari $M = U\Sigma V^T$ (2.3). Disini hanya memakai *row* pertama saja atau yang berisi *orthogonal matrix*. Sedangkan untuk *lambda* sendiri memakai *lambda* yang terbesar saja dikarenakan *lambda* yang bernilai rendah maka akan dianggap sebagai *noise*.

$$\text{User Q} = [5 \quad 0 \quad 0 \quad 0 \quad 0] \times \begin{bmatrix} 0.56 & 0.12 \\ 0.59 & -0.02 \\ 0.56 & 0.12 \\ 0.59 & -0.69 \\ 0.09 & -0.69 \end{bmatrix} = [2.8 \quad 0.6]$$

$$\text{User D} = [0 \quad 4 \quad 5 \quad 0 \quad 0] \times \begin{bmatrix} 0.56 & 0.12 \\ 0.59 & -0.02 \\ 0.56 & 0.12 \\ 0.59 & -0.69 \\ 0.09 & -0.69 \end{bmatrix} = [5.2 \quad 0.4]$$

Pada matriks ini, menunjukan di uji kelayakan *algoritma SVD*. dimana matriks *user q* dan *user d* adalah musik yang telah di-*rating* oleh *user q* dan *user d*. sedangkan matriks disebelahnya adalah matriks *V* yang berisi *music to concept* similarities dari hasil kalkulasi *SVD*. Selanjutnya, untuk mencari kemungkinan *user q* menyukai musik "*On The Hotline*", maka dapat melakukan perkalian antara matriks *user q* dengan *v*. Sehingga didapat hasil berikut:

$$\text{User Q} = [5 \quad 0 \quad 0 \quad 0 \quad 0] = [2.8 \quad 0.6]$$

$$\text{User D} = [0 \quad 4 \quad 5 \quad 0 \quad 0] = [5.2 \quad 0.4]$$

Similarity $\neq 0$

Ketika melakukan perbandingan *user q* dan *user d*. Walaupun, keduanya tidak memiliki *common lines* yang sama. Tetapi ketika dikalkulasikan dengan *algoritma SVD*, maka ada kemungkinan lagu "*On The Hotline*" akan direkomendasikan. Dari sini juga dapat ditarik kesimpulan bahwa, walaupun ada beberapa data kosong. tetapi *algoritma* ini tetap dapat menjalankan fungsinya sebagai sistem rekomendasi.

4. Implementasi dan Pengujian

4.1 Pengujian Nilai Parameter Dengan Metode *RMSE*

Pertama-tama, yang perlu dilakukan adalah membuat model akurasi dimana model ini membutuhkan nilai dari data aktual dan data hasil prediksi. Kita dapat mengambil sample data. Lalu menerapkan rumus *RMSE* Untuk memudahkan perhitungan maka kita akan memakai gambar :

	id	predictions
106	L00116	3.401457
162	L00172	3.257356
35	L00045	3.251210
225	L00235	3.240616
61	L00071	3.228131
229	L00239	3.208289
206	L00216	3.203611
238	L00248	3.175617
77	L00087	3.156341
237	L00247	3.108660

Gambar 4.1 Gambar Hasil Lagu

Tabel 4.1 Perhitungan RMSE

Kode Lagu	Actual Data	Hasil rekomendasi	Different	Squared
L00116	0	3.401457	3.401457	11,569910
L00172	0	3.257356	3.257356	10,61036811
L00045	0	3.251210	3.251210	10,57036646
L00235	0	3.240616	3.240616	10,50159206
L00071	0	3.228131	3.228131	10,42082975
L00239	0	3.208289	3.208289	10,29311831
L00216	0	3.203611	3.203611	10,26312344
L00248	0	3.175617	3.175617	10,08454333
L00087	0	3.156341	3.156341	9,962488508
L00247	0	3.108660	3.108660	9,663766996
			Total	103,940107
			/n	10,3940107
			RMSE	3,223

Diketahui bahwa hasil RMSE untuk contoh sample data dari ini adalah 3,223. Sedangkan Untuk melakukan uji parameter ini yang menggunakan data skala besar, maka akan menggunakan python dan library scipy yang didalamnya terdapat surprise yang berisi data beberapa algoritma yang dipakai di dalam source code SVD. Keuntungan menggunakan RMSE dibandingkan MAE adalah bahwa kesalahan pada RMSE dihitung lebih besar dibandingkan MAE.

5. Kesimpulan dan Saran

5.1 Kesimpulan

Dari Pembahasan yang telah diuraikan sebelumnya, maka peneliti mengambil kesimpulan seperti berikut:

1. Pada pengujian SVD, dapat dimunculkan lagu rekomendasi yang dikalkulasikan dari rating user dengan memberikan 10 lagu teratas mengikuti rekomendasi algoritma.
2. Pada uji algoritma dengan RMSE, SVD memiliki RMSE yang didapatkan adalah 1,77.

5.2 Saran

Tentunya hasil dari tugas akhir yang penulis buat masih jauh dari sempurna, dimana kekurangan dan kesalahan masih terdapat di dalamnya. Adapun hal yang perlu dikembangkan lagi adalah :

1. Menggunakan algoritma gabungan dari content filtering dan collaborative filtering untuk menutupi kelemahan dari cold start dalam tahap pengembangan.
2. Menggunakan algoritma deep learning untuk menutupi kelemahan dari collaborative filtering

Referensi

- [1] Gowda. Asha, Jayaram, and Manjuath, “*Feature Subset Selection Problem using Wrapper Approach in Supervised Learning*,” *International Journal of Computer Applications* 0975 – 8887 ,Vol 1,No.7, 2010.
- [2] X. Wu, Y. Huang, and S. Wang, “A New Similarity Computation Method in Collaborative Filtering based Recommendation System,” *Computation Method in recommendation system*, 2017.
- [3] Thorat, Phoonam B, “Survey on Collaborative Filtering, Content-based Filtering and Hybrid Recommendation System“ *International Journal of Computer Applications*.Vol 110 – No. 4, 2015.
- [4] Z. Sharifi, M. Rezghi, dan M. Nasiri, “New algorithm for recommender systems based on singular value decomposition method,” dalam *ICCKE 2013*, 2013, hlm. 86–91, doi: 10.1109/ICCKE.2013.6682799.
- [5] F. O. Isinkaye, Y. O. Folajimi, dan B. A. Ojokoh, “Recommendation systems: Principles, methods and evaluation,” *Egypt. Inform. J.*, vol. 16, no. 3, hlm. 261–273, Nov 2015, doi: 10.1016/j.eij.2015.06.005.
- [6] B. Sarwar, G. Karypis, J. Konstan, dan J. Reidl, “Item-based collaborative filtering recommendation algorithms,” dalam *Proceedings of the tenth international conference on World Wide Web*.
- [7] Z. Xiong, Y. Cui, Z. Liu, Y. Zhao, M. Hu, and J. Hu, “Evaluating explorative prediction power of machine learning algorithms for materials discovery using k-fold forward cross-validation,” *Comput. Mater. Sci.*, 2020, doi: 10.1016/j.commatsci.2019.109203.
- [8] T. T. Wong, “Performance evaluation of classification algorithms by k-fold and leave-one-out cross validation,” *Pattern Recognit.*, 2015, doi: 10.1016/j.patcog.2015.03.009.
- [9] Pamučar, Dragan, “*Modelling of the fuzzy logical system for offering support in making decisions within the engineering units of the Serbian Army*.” Military Academy, Serbia, 2011
- [10] R. L. Burden and J. D. Faires, *Numerical Analysis* 9th Edition. 2011.
- [11] G. W. S., G. H. Golub, and C. F. Van Loan, “*Matrix Computations.*,” *Math. Comput.*, 1991, doi: 10.2307/2008552.
- [12] X. Su and T. M. Khoshgoftaar, “A Survey of Collaborative Filtering Techniques,” *Adv. Artif. Intell.*, 2009, doi: 10.1155/2009/421425.

