Terbit *online* pada laman : http://teknosi.fti.unand.ac.id/

Jurnal Nasional Teknologi dan Sistem Informasi

| ISSN (Print) 2460-3465 | ISSN (Online) 2476-8812 |

Artikel Penelitian

Sistem Rekomendasi Pembelian *Smartphone* berbasis Algoritma *K-Means* dan *Singular Value Decomposition*

Ivan Zuhdiansyah [a,*], *Ardytha Luthfiarta* [b]

[a,b]*Universitas Dian Nuswantoro, Jalan Imam Bonjol No. 207, Kota Semarang Kode Pos. 50131, Indonesia*

INFORMASI ARTIKEL

Sejarah Artikel:
Diterima Redaksi: 14 Maret 2024
Revisi Akhir: 08 Mei 2024
Diterbitkan *Online*: 09 Mei 2024

KATA KUNCI

Sistem Rekomendasi,
Collaborative Filtering,
K-Means Clustering,
Singular Value Decomposition (SVD)

KORESPONDENSI

E-mail: zuhdiansyahivan@gmail.com*

A B S T R A C T

Perkembangan teknologi informasi yang pesat, memberi dampak pada ketesedian informasi yang berlimpah. Hal ini menjadikan suatu masalah yang disebut kelebihan informasi, menyebabkan pengguna internet sulit memahami dan membuat keptusan. *E-commerce* merupakan salah satu yang terdampak dari kelebihan informasi, dengan banyaknya produk dan pengguna baik dari penjual maupun pembeli yang ada. Sistem rekomendasi adalah bagian penting dari *e-commerce* yang menjadi salah satu cara menangani kelebihan informasi, dengan memberikan rekomendasi produk kepada pembeli agar membantu menentukan pilihan. Dalam sistem rekomendasi memiliki permasalahan *scalability*, dimana banyaknya produk yang tersedia membuatnya menjadi tidak efektif dan efisien dalam memberikan rekomendasi kepada pembeli. Maka, penelitian ini mengusulkan metode sistem rekomendasi yang dikombinasikan teknik *clustering*. Menggunakan algoritma *K-Means* untuk mengelompokkan produk, kemudian algoritma *Singular Value Decomposition (SVD)* untuk membuat rekomendasi di dalam *cluster* yang terbentuk. Hasil keluaran model yaitu, rekomendasi produk dan prediksi *rating* yang diberikan pembeli dari produk yang direkomendasikan. Evaluasi model mendapatkan nilai *dbi* sebesar 0,703 untuk *clustering*, nilai rata-rata terbaik *MAE* 0.8150 dan *RMSE* 1.1781 untuk rekomendasi yang dihasilkan. Kesimpulan yang didapat bahwa metode ini dapat menangani masalah *scalability* dan memberikan rekomendasi yang akurat dengan nilai evaluasi yang lebih baik dibandingkan penelitian sebelumnya.

1. PENDAHULUAN

Saat ini kita hidup dalam era jaringan internet yang terus meningkat dan berkembang dalam dekade terakhir, dengan banyak informasi di semua lingkup kehidupan. David Bawden dan Lyn Robinson[1] menuturkan, kelebihan informasi perlu ditanggapi lebih serius dari sebelumnya. Kata 'kelebihan' diklaim memiliki dua makna, yaitu sebagai masalah utama di zaman ini dan sama sekali bukan masalah. Hal ini disebut sebagai faktor penting seperti dalam berbelanja *online*. Fenomena kelebihan informasi dikenal dengan berbagai nama, seperti *Information Overload, Information Overabundance*, *Information Fatigue*, *Information Anxiety*, *Informtion Pollution*, *Information Violence* dan *Information Assault*. Tidak ada definisi tunggal yang diterima secara umum, namun definisi terbaik yang dapat dipahami adalah situasi yang muncul ketika terdapat begitu banyak informasi yang relevan dan berpotensi berguna sehingga hal tersebut justru menjadi penghalang dan bukan menjadi bantuan. Menurut penelitian [2], yang terpenting adalah ketersediaan informasi produk yang berlebihan berdampak negatif terhadap kecemasan informasi, dan efek stres media sosial terhadap keputusan pembelian *online* serta dampak terjadinya kekhawatiran.

E-commerce menjadi salah satu yang terdampak dari kelebihan informasi, dengan banyaknya produk yang tersedia, membuat pengaruh terhadap kepuasan dan kinerja pengguna saat akan melakukan transaksi. Penelitian [3] menjelaskan, *Electronic Commerce* atau *e-commerce* adalah beragam aktivitas seperti pemasaran, pengembangan, pengiriman, penjualan dan pengeluaran untuk layanan maupun produk dengan *platform*

https://doi.org/10.25077/TEKNOSI.v10i1.2024.45-53

online. *E-commerce* menjadi sesuatu hal yang tidak dapat dipisahkan dari bisnis karena berbagai alasan, termasuk kemudahan penggunaan, aksebilitas universal, variasi yang luas, kemudahan pengelolaan produk dari vendor yang berbeda, cara pembayaran yang terpercaya dan kenyamanan bertransaksi. Fasilitas tersebut meningkatkan kehidupan pengguna ke tingkat kualitas yang tingi, sehingga *e-commerce* memainkan peran penting dalam pengalaman bisnis dan pengguna saat ini. *E-commerce* menghasilkan banyak sekali informasi, maka sistem rekomendasi dimanfaatkan sebagai solusi dari permasalahan kelebihan informasi.

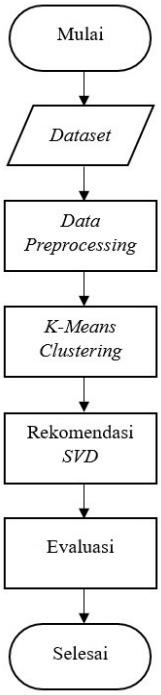
Lebih lanjut, penelitian [3] menjelaskan sistem rekomendasi mempelajari pengalaman dan pendapat dari perilaku pelanggan, kemudian memberi rekomendasi *item* atau produk yang menurutnya paling relevan diantara kemungkinan hasil, sistem rekomendasi juga menyediakan fasilitas untuk meningkatkan adaptasi aplikasi untuk setiap pengguna. Dalam tinjauan [4], sistem rekomendasi dikembangkan untuk mengurangi sebagian masalah kelebihan informasi yang dihasilkan di internet. Selain itu, sistem rekomendasi menerapkan teknik analisis data untuk membantu pengguna menemukan *item* yang ingin dibeli dengan menghasilkan skor prediksi atau daftar *top-N* yang direkomendasikan. Teknik klasik sistem rekomendasi seperti *Collaborative Filtering* masih memainkan peran dominan hampir di semua jenis penerapan, namun memiliki keterbatasan yang salah satunya adalah *scalability.* Masalah yang sama juga ditemui oleh [5], *Collaborative Filtering* membutuhkan data latih dalam jumlah besar yang menyebabkan masalah *scalability.* Di era *Big Data,* semakin banyak *item* dan pengguna ditambahkan menjadikan *scalability* sebagai permasalahan umum dalam sistem rekomendasi. Dua pendekatan umum yang digunakan untuk mengatasi masalah tersebut adalah, *Dimensionality Reduction* dan menggunakan *Clustering-based* untuk menemukan pengguna dalam kelompok kecil alih-alih menemukan pengguna dalam keseluruhan *dataset.*

Pendekatan *clustering-based* digunakan pada penelitian [6], dengan memakai algoritma *K-Means* dan *KNN*. Memakai data ulasan *smartphone* yang diambil dari situs Kimovil dan menggunakan metode *Collaborative Filtering. K-Means clustering* ditujukan untuk mengelompokkan *item smartphone,* kemudian algoritma *KNN* digunakan untuk membuat rekomendasi pada *cluster* yang terbentuk. Hasil evaluasi model mendapatkan nilai *MAE* 1.1047 dan *RMSE* 1.7579. Namun penelitian ini tidak memakai pendekatan *Dimensionality Reduction*. Sedangkan pada penelitian [7], menggunakan algoritma *Singular Value Decomposition (SVD)* dan *KNN*. Menggunakan *dataset* dari sistem informasi mahasiswa *University of Gondar* dan sistem katalog *online.* Model yang diusulkan diuji dalam sistem rekomendasi berbasis model, dengan algoritma *SVD* menghasilkan skor *RMSE* 0.1991 dan skor *RMSE* 0.1623 untuk *dataset* yang dioptimalkan. Sementara untuk model *KNN* dengan *dataset* yang sama, menghasilkan nilai *RMSE* 1.0535. Hal ini menunjukkan bahwa kinerja model *Matrix Factorizatioin* lebih baik dibandingkan model *Neighbor-based.* Namun penelitian ini tidak menggunakan pendekatan *Dimensionality Reduction* dan *Clustering-based* secara bersamaan, melainkan membandingkan keduanya.

Berdasarkan masalah *scalability* dan pendekatan yang digunakan untuk menanganinya yang dipaparkan oleh [4], [5]. Maka, dalam penelitian metode yang digunakan adalah *Collaborative Filtering* dengan algoritma *Singular Value Decomposition* (SVD) sebagai pendekatan *Dimensionality Reduction*, dikombinasikan algoritma *K-Means* sebagai pendekatan *Clustering-based.* Algoritma *K-Means* dipilih mengacu pada penelitian [6], guna mengelompokkan *item* atau produk, kemudian algoritma *SVD* dipilih mengacu pada penelitian [7] guna membuat rekomendasi di dalam *cluster* yang terbentuk.

2. METODE

Metode dalam penelitian ini melalui empat proses yaitu, dimulai dengan data *preprocessing, K-Means clustering,* rekomendasi *Singular Value Decomposition (SVD)* dan terakhir evaluasi. Visualisasi metode dapat dilihat pada Gambar 1.



Gambar 1. *Flowchart* metode penelitian

2.1. Dataset

Dataset yang akan digunakan dalam penelitian ini berasal dari situs *e-commerce* berbasis *website* yang sama seperti penelitian [6]. Cara mendapatkan data di situs tersebut adalah dengan *web scraping* menggunakan *Data Miner*. Menurut [8], *web scraping* merupakan cara mendapatkan data dalam situs *web* secara otomatis tanpa menyalin secara manual. Bertujuan guna mencari informasi atau data tertentu yang kemudian mengumpulkan hasilnya di *web* baru. Berfokus mendapatkan data melalui pengambilan dan ekstrasi. *Data Miner* adalah ekstensi peramban *Google Chrome* yang membantu *scraping* data dari halaman *web* lalu menyimpannya ke dalam dokumen *.csv* atau *spreadsheet excel. Data Miner* dapat *scraping* satu halaman dan mengekstrak data dari beberapa halaman seperti hasil pencarian, produk dan harga, informasi kontak, email, nomor telepon dan

https://doi.org/10.25077/TEKNOSI.v10i1.2024.45-53

banyak lagi. Kemudian *Data Miner* mengonversi data yang diambil menjadi dokumen berformat *.csv* atau *spreadsheet excel* yang dapat diunduh.

Dataset yang akan digunakan memiliki atribut yaitu, *username, smartphone, rating* dan *review* dari ulasan produk *smarthphone.* Contoh format *dataset* tersedia pada Tabel 1.

Tabel 1. Contoh format *dataset*

Username	Smartphone	Rating	Review
pakmobile	poco x3 pro	10.0	the best
nbelala	poco x3 nfc	8.5	cool!
gadgetin	iphone 13	9.5	best phone ever

2.2. Data Preprocessing

Proses ini bertujuan agar data yang terkumpul sesuai dan dapat digunakan untuk proses selanjutnya. Apabila data tidak sesuai kebutuhan seperti data tidak lengkap atau didapati nilai kosong, maka akan berdampak terjadinya *error.* Berikut urutannya :

2.2.1. Data Cleaning

Data cleaning merupakan proses mendeteksi data yang salah atau bermasalah dan memperbaikinya atau menghapusnya dari kumpulan data. Secara umum, hal ini berfungsi untuk mengidentifikasi dan mengganti data yang tidak lengkap, tidak akurat atau tidak relevan. Meskipun teknik yang digunakan berbeda-beda, langkah-langkah dasar yang diikuti adalah hapus data yang tidak relevan atau duplikat, perbaikan kesalahan struktural dan nilai yang hilang dapat diisi menggunakan nilai *mean, median* atau mode terkait fitur model [9].

2.2.2. Data Integration

Dalam buku [10], *data integration* terdiri dari penggabungan data dari beberapa penyimpanan data. Proses ini dilakukan secara hati-hati agar terhindar redundansi dan inkonsistensi dalam kumpulan data yang dihasilkan. Operasi umum yang dilakukan dalam integrasi data adalah identifikasi, penyatuan variabel dan domain, analisis korelasi atribut, duplikasi tupel, dan deteksi konflik dalam nilai data dari sumber yang berbeda.

2.2.3. Data Transformation

Dalam buku [10] juga menjelaskan *data transformation,* adalah data dirubah atau dikonsolidasikan sehingga hasil proses dapat diterapkan atau lebih efisien. Bagian tugas dalam transformasi data adalah *smoothing,* kontruksi fitur, agregasi atau peringkasan data, normalisasi, dikritisasi dan generalisasi. Sebagian besar tugas tersebut akan dipisahkan sebagai tugas independen, karena transformasi data seperti pembersihan data disebut sebagai rangkaian *preprocessing* secara umum. Tugas-tugas yang memerlukan pengawasan manusia dan lebih tergantung pada data adalah teknik transformasi data klasik, seperti pembuatan laporan dan menggabungkan atribut yang sudah ada.

2.2.4. Data Reduction

Data reduction merupakan proses dengan tujuan mengurangi volume data asli dan merepresentasikan dalam volume yang jauh lebih kecil. Hal ini memastikan integritas data sekaligus mengurangi data. Ketika data tersebut sangat penting, maka perlu untuk menguranginya. Mungkin sulit mempelajari informasi yang diinginkan, dan memerlukan waktu lama untuk memproses kueri yang rumit [9].

2.2.5. Tf-idf Weighting

Menurut [11], *TF-IDF* adalah gabungan dari dua istilah, yaitu *Term Frequency (TF)* dan *Inverse Dcoument Frequency (IDF).* Istilah *TF* diterapkan untuk menghitung berapa kali suatu *term* ada dalam sebuah dokumen. *IDF* digunakan untuk menghitung berapa banyak istilah yang muncul untuk seluruh dokumen. *IDF* melakukan perhitungan untuk suku-suku dengan frekuensi kemunculan paling sedikit jika diberi nilai lebih tinggi, namun lebih sering muncul lebih rendah. Konsekuensinya, *TF-IDF* akan melakukan proses pembobotan pada setiap *item* untuk mendapatkan nilai atau kata kunci penting dalam sekumpulan kata. Pembobotan *Tf-idf* didefinisikan sebagai berikut :

$$w_{(x,y)} = tf_{(x,y)} * log\left(\frac{N}{df_x} - 1\right) \quad (1)$$

2.3. K-Means Clustering

Algoritma *K-Means* digunakan dalam mengelompokkan *item* atau produk, sebelum pembuatan rekomendasi. *K-Means* menurut [12], sering digunakan untuk menyelesaikan masalah pengelompokan dan termasuk kedalam *Unsupervised Learning*. *K-Means clustering* memiliki keunggulan yang pertama, komputasi lebih efisien ketika variabelnya besar dengan *globular cluster* dan K kecil, menghasilkan *cluster* yang lebih ketat daripada *Hierarchical clustering*. Kedua, mudah dalam implementasi dan interpretasi hasil *clustering* menjadi daya tarik algoritma ini. Beberapa penerapan algoritma *K-Means*, yaitu untuk klasifikasi dokumen, segmentasi pelanggan, analisis data *rideshare*, pengelompokan otomatis, analisis detail catatan panggilan dan deteksi penipuan asuransi.

Langkah-langkah *clustering* menggunakan algoritma *K-Means* menurut [13] adalah :

1. Menentukan banyak (k), yaitu jumlah *cluster* terhadap kumpulan data yang tersedia.
2. Tentukan k yang menjadi *centroid,* bisa dilakukan secara acak.
3. Hitung jarak setiap data terhadap *centroid* dengan rumus jarak, bisa menggunakan *Euclidean Distance* dengan rumus pada Persamaan 2.

$$d(x_i, \mu_j) = \sqrt{\sum(x_i, \mu_j)^2} \quad (2)$$

4. Mengelompokkan setiap data berdasarkan kedekatan dengan *centroid,* lalu perbarui nilai *centroid* dengan lokasi dari pusat *cluster* menggunakan Persamaan 3.

$$\mu_j(t+1) = \frac{1}{N_{sj}} \sum_{\forall s \in sj} x_j \quad (3)$$

5. Lakukan langkah 2-4 sampai anggota masing-masing *cluster* tidak ada yang berubah.

2.4. *Rekomendasi SVD*

Singular Value Decomposition (SVD) merupakan algoritma yang dipakai untuk pembuatan sistem rekomendasi, juga termasuk dalam algoritma *Dimensionality Reduction.* Menurut penelitian yang dilakukan [14], dijelaskan bahwa algoritma *SVD* memberikan ilustrasi yang tepat dari kumpulan data yang direpresentasikan sebagai matriks dengan jumlah dimensi. Namun, semakin sedikit jumlah dimensi (komponen) yang dipilih, semakin kurang presisi ilustrasi *SVD*. Dengan *SVD*, nilai singular k terbesar dipilih berdasarkan Persamaan 4 berikut :

$$X \rightarrow N.S.Z^T \tag{4}$$

Salah satu keunggulan dari *SVD* adalah dapat menangani matriks yang jarang (*sparse*) dengan cukup efisien.

2.5. *Evaluasi*

Dalam tahap ini evaluasi bertujuan untuk mengetahui baik tidaknya metode yang diusulkan, lalu melakukan analisis terhadap sistem yang dibuat agar dapat menarik kesimpulan. Uji evaluasi yang dipakai penelitian ini :

2.5.1. *Davies Bouldin Index*

Evaluasi yang digunakan untuk mengetahui kualitas dari proses *clustering,* adalah dengan mengukur nilai *DBI.* Menurut [15], *Davies Bouldin Index* bertujuan guna jarak antara satu *cluster* dengan yang lain menjadi maksimal, juga mencari nilai guna jarak antar data dokumen didalam *cluster* yang sama menjadi minimal. Hal serupa juga diungkapkan oleh [16], evaluasi *Davies Bouldin Index* akan menghasilkan nilai paling optimum yakni nilai terkecil yang dihasilkan Persamaan 5.

$$DBI = \frac{1}{k}\sum_{i=1}^{k} max_{j \neq i}\left\{\frac{\bar{d}_i + \bar{d}_j}{d_{i,j}}\right\} \tag{5}$$

Semakin kecil nilai *DBI* dan tidak negatif, maka semakin bagus *cluster* yang dihasilkan dari pengelompokan dengan algoritma yang dipakai.

2.5.2. *Mean Absolute Error*

Saat ini menurut [17], indeks pengukuran utama dari algoritma rekomendasi adalah akurari prediksi dan akurasi klasifikasi. Akurasi prediksi merupakan ukuran kemiripan antara *rating* yang diprediksi dengan *rating* sebenarnya. Indikator evaluasi akurasi prediksi salah satunya adalah *Mean Absolute Error (MAE). MAE* digunakan untuk mengukur kemiripan antara *rating* prediksi dan *rating* sebenarnya, semakin kecil nilai *MAE* maka semakin tinggi keakuratan algoritma yang diusulkan. Rumus *MAE* dapat dilihat sebagai berikut.

$$MAE = \frac{\sum_{i=1}^{n}|p_{u,i} - r_{u,i}|}{|n|} \tag{6}$$

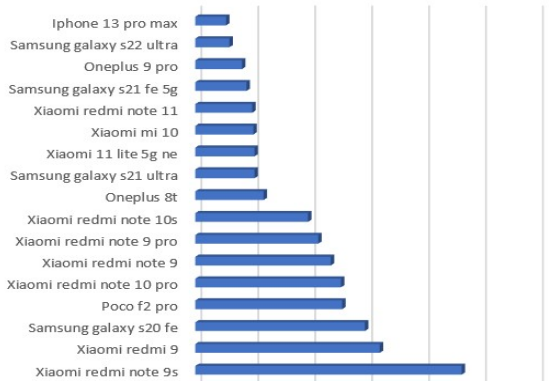
2.5.3. *Root Mean Square Error*

Menurut penelitian [18], *RMSE* mengukur akar kuadrat *MSE,* yaitu perbedaan rata-rata akar kuadrat dari peringkat sebenarnya dan prediksi. Akar kuadrat di sekitar *MSE* menerjemahkan ukuran *RMSE* dalam satuan pada skala yang sama. Nilai *RMSE* yang lebih rendah menentukan akurasi prediksi yang lebih baik yang dihasilkan sistem rekomendasi.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(|A_i - P_i|)^2} \tag{7}$$

3. HASIL

3.1. *Dataset*



Gambar 2. Hasil *scraping*

Dataset yang dipakai bersumber dari https://www.kimovil.com, yaitu data 20 produk dari 25 produk *smartphone* terlaris teratas pada tanggal 29 April 2023. Data-data tersebut merupakan ulasan masing-masing produk *smartphone,* dengan atribut *username* (nama pengguna atau pembeli), *smartphone* (nama produk), *rating* (nilai yang diberikan pengguna setelah membeli dengan *range* 0-10) dan review (ulasan mengenai produk yang dibeli). Data diperoleh melalui proses *scraping* dan disimpan ke dalam format dokumen *spreadsheet excel*. Proses *scraping* dilakukan pada masing-masing *web page* produk *smartphone,* sehingga mendapatkan 20 dokumen *spreadsheet excel* dengan jumlah total 2775 baris.

3.2. *Data Preprocessing*

3.2.1. *Data Cleaning*

Q25

	A	B	C	D	E
1	username	rating	review		
2		8.7			
3					
4					
5					
6					
7	supmaselko	5.5			
8	begin1	10			
9	dikyhadiansyah12345	8.5			
10	Zipperok	10			
11	fabiolT	8.1			
12	simosadam	10			
13	i_am_albertico	10	Great cell phone cannon		
14	patinojuanpablo85	8.9	In general all its features are good except the technology		
15	javiervalero88	10	Very complete terminal for a great price		

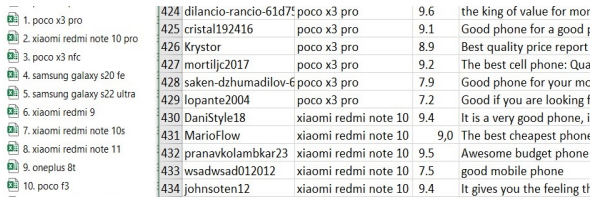
Gambar 3. *Data cleaning*

Tahap awal ini dilakukan guna membersihkan *record* data setelah proses *scraping.* Dapat dilihat dalam Gambar 3, terdapat *noise* seperti nilai kosong pada baris 2 kolom *username, rating, review* sampai baris 6 kolom *username, rating, review* dan nilai

https://doi.org/10.25077/TEKNOSI.v10i1.2024.45-53

tidak lengkap pada baris 7 kolom *review* sampai baris 12 kolom *review*. Kemudian *record* tersebut dihapus dan tambahkan kolom setelah *username* untuk nama *smartphone* pada setiap dokumen.

3.2.2. *Data Integration*



Gambar 4. *Data integration*

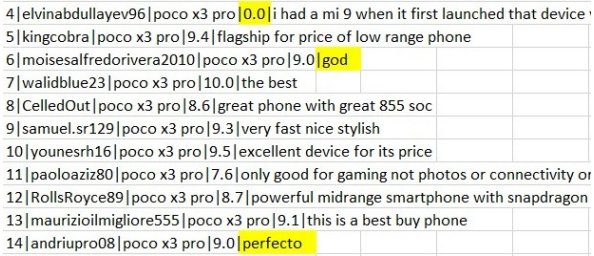
Masing-masing dokumen produk *smartphone* digabungkan menjadi satu, agar menjadi satu dokumen yang utuh. Dokumen yang menjadi satu ini dapat disebut sebagai *dataset* mentah, karena masih belum sesuai dan siap digunakan. Setelah disatukan, kolom *rating* dibenahi dengan penulisan menggunakan titik. Terdapat penulisan nilai *rating* yang tidak benar dengan menggunakan koma seperti nilai 9,0 baris 431 dalam Gambar 4.

3.2.3. *Data Transformation*

Gambar 5. *Data transformation*

Setelah proses integrasi, *dataset* disimpan ke dalam format *.csv* dengan separator ‘ | ’ seperti Gambar 5. Kemudian ditransformasi dalam bentuk tabel dengan kolom *username, smartphone, rating* dan *review.* Dilanjutkan merapikan data di kolom *review,* dengan merubahnya menjadi huruf kecil semua dan menghapus karakter yang dirasa tidak perlu seperti koma, titik, tanda kurung dan sebagainya.

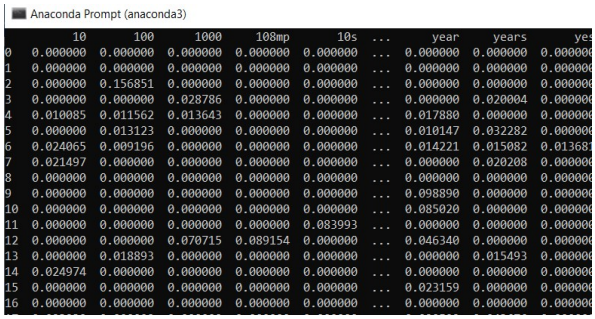
3.2.4. *Data Reduction*



Gambar 6. *Data reduction*

Dataset disimpan kembali dalam format *.csv* setelah proses transformasi, hal ini dilakukan guna melihat kembali data-data yang perlu dikurangi dengan manghapusnya atau dibenahi. Dalam Gambar 6 terdapat nilai *rating* 0.0 pada baris 4, terdapat satu atau lebih kata yang tidak sesuai dengan konteks seperti baris 6 dan 14. Apabila didapati nilai kosong satu atau lebih pada baris tertentu, maka baris tersebut akan dihapus. Tahap ini menghasilkan *dataset* yang siap pakai dengan jumlah 2570 baris.

3.2.5. *Tf-idf Weighting*

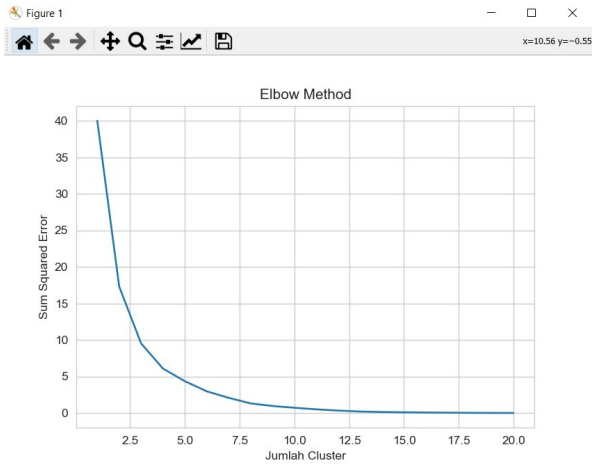


Gambar 7. *Tf-idf weighting*

Pembobotan dilakukan pada kolom *review* yang bernilai teks, menjadi besaran nilai angka yang akan mengganti nilai teks tersebut. Pertama, membaca *dataset* dan rubah ke dalam bentuk tabel. Kedua, kelompokkan data berdasarkan kolom *smartphone,* sehingga setiap baris kolom *review* menjadi sebuah *list.* Ketiga, memecah *list* menjadi per kata atau tokenisasi, hapus kata *stopwords.* Terakhir, lakukan pembobotan *Tf-idf* seperti pada Gambar 7.

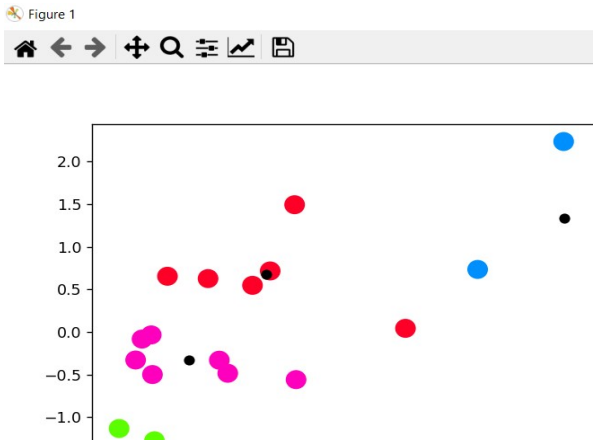
3.3. *K-Means Clustering*

Proses *clustering* memakai atribut *smartphone, rating, review* dari *dataset,* dimulai dengan menentukan nilai k menggunakan *elbow method*. Menurut [19], *elbow method* adalah metode terkenal untuk memperkirakan jumlah *cluster* yang diperlukan sebagai parameter awal dalam algorima *K-Means.*



Gambar 8. *Elbow method*

Dari Gambar 8, nilai k yang dipilih untuk *K-Means clustering* adalah 4. Visualisasi *cluster* yang terbentuk dapat dilihat pada Gambar 9 dengan *point* hitam sebagai titik pusat setiap *cluster* dan anggota masing-masing *cluster* dapat dilihat pada Tabel 2.



Gambar 9. *K-Means cluster*

Tabel 2. *K-Means cluster*

Cluster 0	***Cluster 1***	***Cluster 2***	***Cluster 3***
Xiaomi MI 10	Oneplus 8t	Poco X3 NFC	Xiaomi Redmi 9
Iphone 13 Pro Max	Poco F2 Pro	Poco F3	Oneplus 9 Pro
Xiaomi Redmi Note 11	Xiaomi Redmi Note 9s	Poco x3 pro	Xiaomi Redmi Note 9
Samsung Galaxy s22 Ultra	Xiaomi Redmi Note 10 Pro		Samsung Galaxy S21 Ultra
	Xiaomi Redmi Note 10s		Xiaomi 11 Lite 5g NE
	Samsung Galaxy S20 FE		Samsung Galaxy S21 FE 5g
			Xiaomi Redmi Note 9 Pro

3.4. Rekomendasi SVD

Proses rekomendasi memakai atribut *username, smartphone, rating* dari *dataset* dan berjalan di dalam *cluster.* Apabila seorang pengguna membeli produk yang berada di *cluster 1*, maka hasil rekomendasi produk adalah nilai prediksi *rating* tertinggi diantara produk-produk di *cluster 1* selain produk yang dibeli. Kondisi lain, apabila seorang pengguna membeli produk lebih dari satu dan produk-produk tesebut berada di dalam *cluster* yang berbeda, maka sistem rekomendasi membuat rekomendasi di masing-masing *cluster* yang terdapat produk yang dibeli oleh pengguna tadi. Kondisi ini terjadi karena, dalam penelitian ini tidak memakai atribut waktu yang menunjukkan kapan pengguna melakukan pembelian.

3.5. Evaluasi

3.5.1. Evaluasi Clustering

Uji evaluasi *clustering* memakai *Davies Bouldin Index* menghasilkan nilai 0.703 dari 4 *cluster* yang terbentuk, hasil nilai ini terbilang relatif kecil yang menandakan bahwa proses *clustering* bagus.

3.5.2. Evaluasi Rekomendasi

Dari sistem rekomendasi yang dibuat dalam penelitian ini, dilakukan uji metrik evaluasi *MAE* dan *RMSE* dengan *cross validation* 5 kali seperti penelitian [7]. Kemudian dengan *dataset* yang sama, dilakukan pembuatan sistem rekomendasi dengan metode yang dipakai oleh [6] dan juga, dilakukan uji evaluasi yang sama sebagai pembanding. Nilai evaluasi yang dihasilkan dapat dilihat pada Tabel 3 dan 4.

Tabel 3. Nilai rata-rata *MAE*

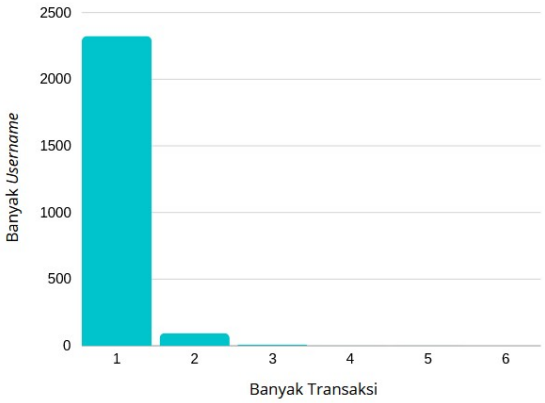
	K-Means SVD	***K-Means KNNBasic***
Cluster 0	0.8150	0.9291
Cluster 1	0.9085	0.9476
Cluster 2	0.9083	0.9115
Cluster 3	1.1955	1.2696

Tabel 4. Nilai rata-rata *RMSE*

	K-Means SVD	***K-Means KNNBasic***
Cluster 0	1.1781	1.3050
Cluster 1	1.3595	1.4023
Cluster 2	1.3874	1.3894
Cluster 3	1.6510	1.7137

4. PEMBAHASAN

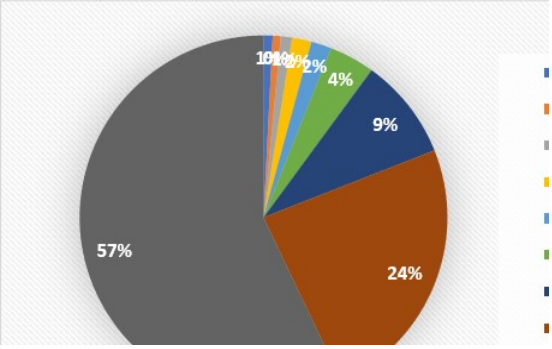
4.1. Dataset



Gambar 10. Banyak transaksi *username*

Dataset yang digunakan terdiri 2435 *username* dari total transaksi 2570. Dapat dilihat pada Gambar 10, transaksi paling banyak yang dilakukan adalah 6 kali oleh satu *username* dan transaksi paling sedikit yang dilakukan adalah satu kali oleh 2325 *username.* Hal ini menjadikan kondisi 94.7% *sparse* dalam *dataset,* angka presentase ini didapat dari rasio perbandingan antara nilai kosong dan terisi dalam matriks interaksi *username-smartphone.*

https://doi.org/10.25077/TEKNOSI.v10i1.2024.45-53



Gambar 11. Persentase *rating*

Untuk nilai *rating* dalam *dataset*, nilai paling banyak diberikan berkisar 9.1-10.0 dengan presentase pada Gambar 11. Hal ini dapat dikatakan bahwa pembeli merasa senang atau puas atas transaksi meski baru sekali membeli. Dari *feedback* ini, menjadikan pihak *e-commerce* memberikan penawaran produk melalui sistem rekomendasi yang bertujuan menarik pengguna agar memakai situs ini dalam membeli suatu produk lagi.

4.2. Data Preprocessing

Pada tahap *data cleaning* sampai menjadi *dataset* mentah dalam *data integration,* dilakukan secara manual di *Microsoft Excel.* Lalu di tahap selanjutnya, data ditransformasi dengan pemrograman *python* melalui *library pandas,* merapikan data di kolom *review* dengan *case folding* dan menghapus karakter yang tidak perlu seperti tanda baca. Diteruskan tahap *data reduction,* melihat kembali data yang perlu dikurangi atau dibenahi seperti nilai *rating* yang dipakai yaitu dari 1.0-10.0. Tahap *data reduction* dilakukan secara manual di *Microsoft Excel.*

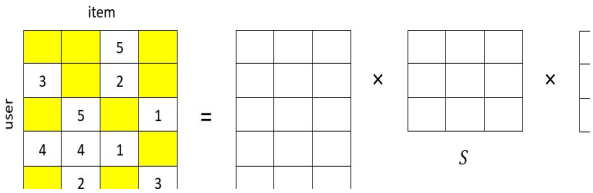
Pembobotan *Tf-idf* disyaratkan agar dapat dilakukan proses *clustering* yang membutuhkan data bernilai angka, sedangkan dalam kolom *review* data bernilai teks. Pembobotan dilakukan melalui pemrograman *pyhton* dengan *library scikit-learn* dan menurut [20], *stopwords* adalah kata umum berfrekuensi tinggi dan tidak memiliki arti, pekerjaan ini dapat dilakukan menggunakan *library nltk*.

4.3. K-Means Clustering

Proses *clustering* dijalankan melalui pemrograman *python* dengan memakai *library scikit-learn.* Parameter *K-Means* yang digunakan dalam pemrograman adalah *n_clusters=3* dan *random_state=0.* Selain parameter yang disebutkan, parameter lain yang digunakan adalah sesuai dengan *default* dari *library scikit-learn.* Menurut [21], *scikit-learn* adalah paket *machine learning* paling komprehensif dan *open-source.* Paket ini mencakup empat topik utama *machine learning,* yaitu tranformasi data, *supervised learning, unsupervised learning,* serta evaluasi dan seleksi model.

4.4. Rekomendasi SVD

Proses rekomendasi dilakukan dengan bahasa pemrograman *python* dan *library* yang dipakai adalah *scikit surprise. Library scikit surprise* dibuat oleh Nicolas Hug[22], ditujukan untuk membangun sistem rekomendasi yang menangani data *rating* eksplisit. Menurut [23], proses *SVD* sama seperti model *machine learning* yang langkahnya mencakup inisiasi model dan *fitting* pada *trainset,* sehingga menghasilkan prediksi *testset.* Pada Gambar 12, dapat dilihat matriks R terdapat nilai kosong, untuk mengisi nilai tersebut perlu dilakukan prediksi. Algoritma *SVD* memprediksi nilai tersebut menggunakan teknik seperti *Gradient Descent,* kemudian matriks R difaktorisasi menjadi tiga matriks.



Gambar 12. Rekomendasi *SVD*

Tiga matriks tadi yaitu, U, S, V^T, lalu matriks tersebut dikalikan dan membentuk matriks R yang lengkap, dimana sel yang tadinya kosong terisi dengan nilai prediksi. Kolom dengan nilai prediksi tertinggi setiap baris merupakan *item* yang akan direkomendasikan pada baris tersebut, prediksi nilai *rating* yang akan diberikan oleh *user* adalah nilai prediksi tertingi pada setiap baris itu sendiri. Lebih jelas dapat diperhatikan matriks R pada Gambar 12, *user* di baris pertama berinteraksi dengan *item* ketiga, setelah proses *SVD* nilai kosong pada baris pertama terisi nilai prediksi. Kolom dengan nilai prediksi tertinggi akan menjadi *item* yang direkomendasikan kepada *user* pertama, dan prediksi *rating* yang akan diberikan *user* baris tersebut adalah nilai prediksi tertinggi itu sendiri.

4.5. Evaluasi

4.5.1. Evaluasi Clustering

Berdasarkan dari *cluster* yang terbentuk, setiap anggota dalam masing-masing *cluster* memiliki kemiripan, banyaknya transaksi untuk sebuah produk *smartphone* menentukan letak produk tersebut berada didalam *cluster* yang mana. Pada *cluster* 1 berisi 4 anggota produk *smartphone* dengan keseluruhan transaksi berjumlah 1267 dari 1230 *username.* Sedangkan pada *cluster* 2, beranggotakan 9 produk *smartphone* dengan total transaksi berjumlah 407 dari 398 *username.* Analisis tersebut berlaku untuk transaksi yang memiliki nilai atribut yang lengkap berupa *smartphone, rating* dan *review.* Apabila *username* atau pengguna hanya memberikan nilai *rating* saja dan tidak memberi *review* setelah melakukan transaksi, maka data tersebut tidak dapat diproses melainkan akan dihapus.

4.5.2. Evaluasi Rekomendasi

Setelah beberapa kali menjalankan program, nilai rata-rata *MAE* dan *RMSE K-Means SVD* selalu lebih kecil dibanding *K-Means KNN* untuk setiap *cluster,* meski pada *cluster* 2 nilai *RMSE* hampir sama namun tetap lebih kecil *K-Means SVD.* Nilai yang dihasilkan menunjukan bahwa *K-Means SVD* lebih baik, meski terbilang masih kurang bagus karena lebih dari 1. Nilai tersebut disebabkan oleh kondisi dataset yang *sparse,* dimana sebagian besar *username* atau pengguna baru melakukan transaksi satu kali. Akibatnya ketika dibuat matriks interaksi pengguna-produk dan *rating* sebagai nilai, banyak sel tidak bernilai atau kosong.

Semakin banyak transaksi yang dilakukan setiap pengguna, maka semakin bagus nilai *MAE* dan *RMSE*, atau dengan kata lain semakin bagus rekomendasi yang dihasilkan.

Terdapat kondisi, apabila ada *cluster* hanya beranggotakan 2 produk saja, maka dapat disimpulkan rekomendasi untuk produk 1 adalah produk 2 dan sebaliknya. Rekomendasi pada kondisi ini dapat dibuat secara langsung maupun melalui algoritma *SVD.*

5. KESIMPULAN

Dari penelitian yang sudah dilaksanakan, sistem rekomendasi dapat dijalankan dengan metode *Collaborative Filtering model-based* dengan algoritma *SVD,* dan dikombinasikan algoritma *K-Means clustering* untuk mengelompokkan produk sebelum rekomendasi dibuat. Sistem dapat menghasilkan rekomendasi produk dan prediksi *rating* dari pengguna terhadap produk yang direkomendasikan. Dengan metode ini, proses rekomendasi berjalan lebih efektif dan efisien, karena produk sudah dikelompokkan sesuai dengan kemiripannya. Jadi, proses rekomendasi berjalan dalam *cluster* alih-alih berjalan dalam keseluruhan *dataset,* hal ini bisa dikatakan dapat menjadi solusi masalah *scalability*. Prinsip dasar sistem rekomendasi adalah memberikan rekomendasi berdasarkan preferensi pengguna ataupun produk. Dengan produk yang dikelompokkan berdasarkan kemiripannya dan kemudian dibuat rekomendasi didalam kelompok-kelompok tersebut, menjadikan hasil rekomendasi lebih akurat dibandingkan metode yang sama tanpa *K-Means clustering*. Evaluasi yang dicapai setelah dibandingkan dengan penelitian [6] menunjukkan metode ini lebih baik, menghasilkan nilai rata-rata terbaik *MAE* 0.8150 dan *RMSE* 1.1781 pada *cluster* 0. Nilai rata-rata *RMSE* terbilang masih tinggi, hal ini disebabkan oleh kondisi *sparse* dalam *dataset.* Dari penelitian ini dapat dikatakan model *Matrix Factorization* lebih baik dibanding model *Neighborhood-based,* selaras dengan yang dikatakan oleh [7]. Untuk penelitian berikutnya dapat dilakukan penyaringan *dataset* dengan minimal transaksi yang dilakukan pengguna.

DAFTAR PUSTAKA

[1] D. Bawden and L. Robinson, *Information Overload: An Introduction*. 2020.

[2] B. A. Al-Youzbaky and R. D. Hanna, "The Effect of Information Overload, and Social Media Fatigue on Online Consumers Purchasing Decisions: The Mediating Role of Technostress and Information Anxiety," *J. Syst. Manag. Sci.*, vol. 12, no. 2, pp. 201–226, 2022, doi: 10.33168/JSMS.2022.0209.

[3] P. M. Alamdari, N. J. Navimipour, M. Hosseinzadeh, A. A. Safaei, and A. Darwesh, "A Systematic Study on the Recommender Systems in the E-Commerce," *IEEE Access*, vol. 8, pp. 115694–115716, 2020, doi: 10.1109/ACCESS.2020.3002803.

[4] F. Alyari and N. Jafari Navimipour, "Recommender systems: A systematic review of the state of the art literature and suggestions for future research," *Kybernetes*, vol. 47, no. 5, pp. 985–1017, 2018, doi: 10.1108/K-06-2017-0196.

[5] D. Roy and M. Dutta, "A systematic review and research perspective on recommender systems," *J. Big Data*, vol. 9, no. 1, 2022, doi: 10.1186/s40537-022-00592-5.

[6] F. A. Prayoga and K. Kusnawi, "Smartphone Recommendation System Using Model-Based Collaborative Filtering Method," *J. Tek. Inform.*, vol. 3, no. 6, pp. 1613–1622, 2022, doi: 10.20884/1.jutif.2022.3.6.413.

[7] E. Ahmed and A. Letta, "Book Recommendation Using Collaborative Filtering Algorithm," *Appl. Comput. Intell. Soft Comput.*, vol. 2023, 2023, doi: 10.1155/2023/1514801.

[8] D. D. A. Yani, H. S. Pratiwi, and H. Muhardi, "Implementasi Web Scraping untuk Pengambilan Data pada Situs Marketplace," *J. Sist. dan Teknol. Inf.*, vol. 7, no. 4, p. 257, 2019, doi: 10.26418/justin.v7i4.30930.

[9] K. Maharana, S. Mondal, and B. Nemade, "A review: Data pre-processing and data augmentation techniques," *Glob. Transitions Proc.*, vol. 3, no. 1, pp. 91–99, 2022, doi: 10.1016/j.gltp.2022.04.020.

[10] S. Garcia, J. Luengo, and F. Herrera, *Data Preprocessing in Data Mining. Intelligent Systems Reference Library. 2015*, vol. 10. 2015.

[11] M. Johari and A. Laksito, "The Hybrid Recommender System of the Indonesian Online Market Products using IMDb weight rating and TF-IDF," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 5, no. 5, pp. 977–983, 2021, doi: 10.29207/resti.v5i5.3486.

[12] S. Ray, "Introduction to Machine Learning and Different types of Machine Learning Algorithms," *Proc. Int. Conf. Mach. Learn. Big Data, Cloud Parallel Comput. Trends, Prespectives Prospect. Com. 2019*, pp. 35–39, 2019.

[13] F. Indriyani and E. Irfiani, "Clustering Data Penjualan pada Toko Perlengkapan Outdoor Menggunakan Metode K-Means," *JUITA J. Inform.*, vol. 7, no. 2, p. 109, 2019, doi: 10.30595/juita.v7i2.5529.

[14] F. Anowar, S. Sadaoui, and B. Selim, "Conceptual and empirical comparison of dimensionality reduction," *Comput. Sci. Rev.*, vol. 40, p. 100378, 2021, doi: 10.1016/j.cosrev.2021.100378.

[15] S. Ramadhani, D. Azzahra, and T. Z, "Comparison of K-Means and K-Medoids Algorithms in Text Mining based on Davies Bouldin Index Testing for Classification of Student's Thesis," *Digit. Zo. J. Teknol. Inf. dan Komun.*, vol. 13, no. 1, pp. 24–33, 2022, doi: 10.31849/digitalzone.v13i1.9292.

[16] H. Santoso, H. Magdalena, and H. Wardhana, "Aplikasi Dynamic Cluster pada K-Means BerbasisWeb untuk Klasifikasi Data Industri Rumahan," *MATRIK J. Manajemen, Tek. Inform. dan Rekayasa Komput.*, vol. 21, no. 3, pp. 541–554, 2022, doi: 10.30812/matrik.v21i3.1720.

[17] W. Fu, J. Liu, and Y. Lai, "Collaborative filtering recommendation algorithm towards intelligent community," *Discret. Contin. Dyn. Syst. - Ser. S*, vol. 12, no. 4–5, pp. 811–822, 2019, doi: 10.3934/dcdss.2019054.

[18] S. G. K. Patro *et al.*, "A Hybrid Action-Related K-Nearest Neighbour (HAR-KNN) Approach for Recommendation Systems," *IEEE Access*, vol. 8, pp. 90978–90991, 2020, doi: 10.1109/ACCESS.2020.2994056.

[19] A. J. Onumanyi, D. N. Molokomme, S. J. Isaac, and A. M. Abu-Mahfouz, "AutoElbow: An Automatic Elbow Detection Method for Estimating the Number of Clusters in a Dataset," *Appl. Sci. 2022, Vol. 12, Page 7515*, vol. 12, no. 15, p. 7515, Jul. 2022, doi: 10.3390/APP12157515.

[20] M. Wang and F. Hu, "The application of nltk library for python natural language processing in corpus research," *Theory Pract. Lang. Stud.*, vol. 11, no. 9, pp. 1041–1049, 2021, doi: 10.17507/tpls.1109.09.

[21] J. Hao and T. K. Ho, "Machine Learning Made Easy: A Review of Scikit-learn Package in Python Programming Language," *J. Educ. Behav. Stat.*, vol. 44, no. 3, pp. 348–361, 2019, doi: 10.3102/1076998619832248.

[22] N. Hug, "Surprise: A Python library for recommender systems," *J. Open Source Softw.*, vol. 5, no. 52, p. 2174, 2020, doi: 10.21105/joss.02174.

[23] A. A. Patoulia, A. Kiourtis, A. Mavrogiorgou, and D. Kyriazis, "A Comparative Study of Collaborative Filtering in Product Recommendation," *Emerg. Sci. J.*, vol. 7, no. 1, pp. 1–15, 2023, doi: 10.28991/ESJ-2023-07-01-01.

NOMENKLATUR

tf	frekuensi *term* (x) dalam suatu dokumen (y)
N	jumlah seluruh dokumen yang ada
df	jumlah semua dokumen yang mengandung suatu *term*
d	titik dokumen
x_i	data kriteria
μ_j	*centroid* pada *cluster* ke-j
$\mu_j(t+1)$	*centroid* baru dalam iterasi ke $(t+1)$
N_{sj}	banyak data dalam *cluster* sj
X	matriks asli yang didekomposisi menjadi tiga matriks
N	matriks $(n \times k)$, kolom vektor satuan ortogonal
S	matriks diagonal $(k \times k)$
Z	matriks ortogonal $Z^T.Z = Z.Z^T = I$ berukuran $k \times d$
k	jumlah *cluster*
d_i	jarak rata-rata antar data dalam *cluster i* sementara
d_j	jarak rata-rata antar data dalam *cluster j* sementara
$p_{u,i}$	prediksi *rating* produk
$r_{u,i}$	*rating* pengguna sebenarnya
I	kumpulan produk yang digunakan memprediksi skor
P	peringkat pengguna yang diprediksi
A	peringkat aktual pengguna
n	jumlah total produk dalam daftar rekomendasi

BIODATA PENULIS

Ivan Zuhdiansyah
Mahasiswa Program Studi Teknik Informatika S-1 Fakultas Ilmu Komputer Universitas Dian Nuswantoro, fokus peminatan pada *Knowledge Discovery in Database and Information Retrieval*.

Ardytha Luthfiarta, M. Kom
Dosen Program Studi Teknik Informatka S-1 Fakultas Ilmu Komputer Universitas Dian Nuswantoro, fokus penelitian di bidang *Data Mining, Information Retrieval, Machine Learning, Deep Learning, NLP.*