

Rekomendasi Hunian pada Aplikasi Koseeker Menggunakan Hybrid Recommender System dengan Implicit Feedback

Muhammad Arizal Saputro¹, Agung Toto Wibowo²

^{1,2}Fakultas Informatika, Universitas Telkom, Bandung

¹arizalsaputro@student.telkomuniversity.ac.id, ²agungtoto@telkomuniversity.ac.id,

Abstrak

Perkembangan dunia pendidikan menyebabkan meningkatnya jumlah kesadaran orang untuk menuntut ilmu hingga jenjang perguruan tinggi, meningkatnya jumlah mahasiswa setiap tahun menyebabkan peningkatan permintaan hunian sewa sementara. Koseeker adalah platform pencarian hunian sewa untuk mahasiswa yang memiliki banyak data yang tersebar di beberapa daerah, tetapi koseeker tidak memiliki sistem rekomendasi berdasarkan preferensi pengguna. Dengan adanya sistem rekomendasi yang tepat, dapat memudahkan pengguna menemukan hunian sesuai preferensi dan meningkatkan bisnis koseeker. Dalam penelitian ini, dibangun sistem rekomendasi hunian menggunakan Hybrid Recommender System dengan menggabungkan Collaborative Filtering dan Content-Based Filtering berdasarkan data aktivitas user yang bersifat implisit. Dari hasil penelitian ini, model hybrid yang dibangun dapat mengungguli collaborative maupun content-based filtering dengan skor AUC sebesar 0,914.

Kata kunci : sistem rekomendasi, collaborative filtering, hybrid recommender system, content-based filtering, implicit feedback

Abstract

The development of the world of education led to an increase in the number of people's awareness to study to the highest level, the increasing number of students every year causes an increase in demand for rental housing. Koseeker is a rental housing search platform for student that have a lot of data that scattered in several regions, but, Koseeker does not have a recommendation system based on user preferences. With the right recommender system, it can be easier for user to find rent house according to their preference and increase the Koseeker revenue. In this study, a housing recommendation system was built using Hybrid Recommender System by combining Collaborative Filtering and Content-Based Filtering based on user implicit feedback. From the results of this study, the hybrid model that was build can outperform collaborative and content-based filtering with an AUC score of 0,914.

Keywords: recommender system, collaborative filtering, hybrid recommender system, content-based filtering, implicit feedback

1. Pendahuluan

Latar Belakang

Perkembangan dunia pendidikan menyebabkan meningkatnya jumlah kesadaran orang untuk menuntut ilmu hingga jenjang perguruan tinggi. Hal itu pula yang menyebabkan meningkatnya jumlah mahasiswa setiap tahun [1]. Peningkatan tersebut menimbulkan meningkatnya permintaan hunian sewa untuk mahasiswa baik itu indekos, kontrakan, atau apartemen. Terdapat banyak platform yang menyediakan informasi mengenai pencarian hunian sementara bagi mahasiswa, salah satunya adalah Koseeker.

Koseeker merupakan *platform* pencarian hunian sementara untuk mahasiswa yang memiliki banyak data tersebar di beberapa universitas di beberapa daerah seperti Bandung, Jakarta, Semarang dan Yogyakarta. Namun Koseeker masih memiliki kekurangan, dimana koseker belum miliki sistem rekomendasi, oleh karena itu hunian yang ditampilkan tidak berdasarkan preferensi pengguna namun hanya menggunakan *query* biasa tanpa mempertimbangkan preferensi dan minat dari pengguna. Hal tersebut sangat disayangkan sebab dengan adanya sistem rekomendasi yang tepat, dapat memudahkan pengguna menemukan hunian sesuai kriteria dan kemungkinan pengguna untuk melakukan transaksi menjadi lebih besar.

Pada penelitian ini, Sistem Rekomendasi Hunian dibangun untuk mengatasi permasalahan tersebut. Ada beberapa teknik dalam membuat sistem rekomendasi diantaranya adalah collaborative filtering dan content-based

filtering. Collaborative filtering menghasilkan rekomendasi berdasarkan pengguna lain yang memiliki preferensi yang sama, sedangkan content-based filtering menghasilkan rekomendasi berdasarkan item yang sama dengan yang disukai oleh user sebelumnya. Collaborative filtering merupakan teknik yang paling banyak dalam membangun sistem rekomendasi, namun collaborative filtering memiliki kekurangan dimana tidak bisa merekomendasikan item yang belum pernah dirating atau berinteraksi dengan user, oleh karena ini dalam penelitian ini, teknik hybrid dengan menggabungkan collaborative filtering dan content-based filtering digunakan untuk mengoptimalkan Sistem Rekomendasi Hunian yang dibangun.

Topik dan Batasannya

Koseeker memiliki data catatan aktivitas penggunanya dalam bentuk implicit feedback seperti hunian yang sering dilihat, disukai, dipesan dan lain-lain. Berdasarkan penelitian sebelumnya [2], menyimpulkan bahwa algoritma Weighted Alternating Least Square pada metode matrix factorization tepat untuk menangani implicit feedback pada Collaborative Filtering. Pada penelitian ini penulis menggunakan algoritma Weighted Alternating Least Square (W-ALS) [5] pada metode matrix factorization untuk menghasilkan rekomendasi pada teknik collaborative filtering, metode ini dipilih karena W-ALS bertujuan untuk mengoptimalkan collaborative filtering untuk dataset yang bersifat implisit [5].

Content-based filtering pada penelitian ini akan menghasilkan rekomendasi berdasarkan item yang memiliki kesamaan dengan preferensi pengguna. Seperti yang diketahui, data dari koseeker tersebar di beberapa daerah seperti Jakarta, Bandung, dan Yogyakarta oleh karena itu agar hunian yang direkomendasikan sesuai dengan daerah atau kampus pengguna, maka akan dilakukan pengelompokan hunian dengan k-means berdasarkan lokasi hunian sebelum dilakukan content-based filtering.

Pada penelitian ini, rekomendasi hunian yang dihasilkan merupakan penggabungan dari teknik collaborative filtering dan content-based filtering dengan pendekatan *Hybrid Recommender System* menggunakan teknik weighted. Adapun perumusan masalah pada penelitian ini adalah bagaimana cara mengimplementasikan pendekatan *Hybrid Recommender System* dengan menggabungkan collaborative dan content-based filtering untuk merekomendasikan hunian sesuai preferensi pengguna dan bagaimana pengaruh pendekatan tersebut pada akurasi rekomendasi hunian yang diberikan. Batasan masalah pada penelitian ini adalah belum menerapkan pengujian dengan *A/B Testing* secara online untuk mengevaluasi performa sistem rekomendasi, hal ini dikarenakan keterbatasan sumber daya.

Tujuan

Berdasarkan permasalahan yang terdapat pada latar belakang yang sudah dijelaskan sebelumnya, maka tujuan dari penelitian ini adalah untuk membangun sistem rekomendasi hunian untuk aplikasi Koseeker sesuai dengan preferensi pengguna dengan pendekatan *Hybrid Recommender System* dan mengetahui besar akurasi dari sistem rekomendasi yang dihasilkan dengan pendekatan tersebut.

Organisasi Tulisan

Pada penulisan penelitian ini bab 2 berisikan studi literatur yang berhubungan dengan penelitian ini. Pada bab 3 berisikan penjelasan sistem yang dibangun sekaligus cara kerjanya. Pada bab 4 berisikan hasil pengujian dari proses penelitian ini dan pada bab 5 berisikan kesimpulan dari penelitian ini.

2. Studi Terkait

2.1 Recommender System

Recommender System atau sistem rekomendasi adalah alat dan teknik perangkat lunak untuk memberikan saran berupa item yang berguna bagi pengguna [10]. Saran yang diberikan berkaitan dengan pengambilan keputusan, seperti barang yang akan dibeli, musik yang di dengarkan, atau berita apa yang harus dibaca dan lain-lain. Sistem rekomendasi telah digunakan di mana-mana dan digunakan dalam berbagai bidang beberapa tahun terakhir. Banyak website e-commerce seperti amazon telah menggunakan sistem rekomendasi untuk membantu pengguna mereka membuat keputusan pembelian [8]. Sistem rekomendasi umumnya menghasilkan rekomendasi melalui salah satu metode 1) content-based filtering atau 2) collaborative filtering. Content-based filtering menggunakan atribut item untuk menghasilkan rekomendasi yang memiliki atribut serupa. Collaborative filtering membangun model dari perilaku, aktivitas masa lalu atau preferensi pengguna untuk membuat rekomendasi berdasarkan kesamaan dengan pengguna lain [2]. Disamping keduanya memiliki kekuatan dan kelemahan masing – masing, metode collaborative filtering adalah yang paling banyak digunakan , seperti dikutip dalam [7] collaborative filtering memberikan akurasi lebih baik dari pada metode content-based. Hal ini dikarenakan collaborative filtering

mampu memberikan rekomendasi tanpa pengetahuan item [10]. Akan tetapi pada penelitian ini dibangun sistem rekomendasi dengan menggabungkan kedua teknik tersebut untuk menghasilkan rekomendasi yang optimal.

2.2 Collaborative Filtering (CF)

CF merupakan salah satu pendekatan dari sistem rekomendasi berdasarkan interaksi antara user dan item. CF membangun model dari perilaku, aktivitas masa lalu atau preferensi pengguna untuk membuat rekomendasi berdasarkan kesamaan dengan pengguna lain [2]. Ada dua pendekatan utama dalam CF mengidentifikasi pola interaksi antara user dan item yaitu pendekatan berbasis memori dan pendekatan berbasis model [2].

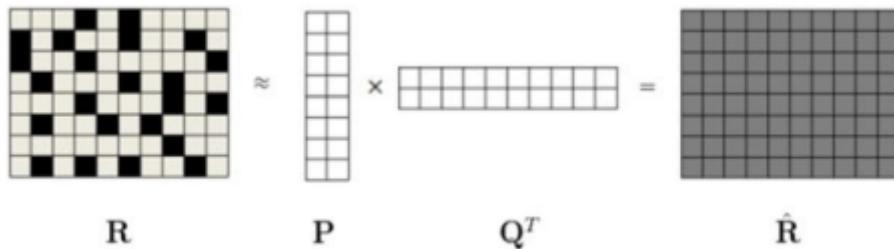
Pendekatan berbasis memori mengambil data rating pengguna untuk menghitung kesamaan antara user dan item untuk menghasilkan rekomendasi. Pendekatan berbasis memori yang terkenal adalah *neighborhood-based* yang menghitung *similarity* terhadap user lain untuk menghasilkan rekomendasi. Meskipun relative sederhana, performa dengan pendekatan ini kurang bagus dengan dataset yang besar dan memiliki tingkat sparsity yang tinggi [10] dimana dalam dunia nyata seringkali dijumpai dataset yang besar dan tingkat *sparsity* yang tinggi. Oleh karena itu pendekatan berbasil model atau dikenal juga pendekatan berbasis Matrix Factorization menjadi sangat populer baru-baru ini karena dapat menangani sparsity lebih baik meningkatkan akurasi daripada pendekatan berbasis memori [2].

2.3 Matrix Factorization

Matrix factorization merupakan teknik pemodelan data untuk collaborative filtering dengan menggunakan matrik. Matrix factorization bekerja dengan cara menguraikan matrik interaksi antara user dan item menjadi dua matrik yang lebih kecil dan ketika matrik tersebut dikalikan kembali akan menghasilkan matrik [7]. Konsep matrix factorization secara matematis dirumuskan

$$R \approx P \times Q^T = R \quad (1)$$

P merepresentasikan hubungan user dan fiturnya dan Q merepresentasikan hubungan item dengan fiturnya. Matrix factorization di ilustrasikan dalam dalam gambar 1.



Gambar 1. ilustrasi konsep matrix factorization. Gambar diambil dari [3]

Dalam Collaborative filtering, model matrix factorization atau biasa disebut *latent factor model* bertujuan untuk menemukan fitur laten yang nantinya akan digunakan memprediksi rating untuk setiap pengguna dan item [7]. Diasumsikan matrik $R_{n \times k}$ dimana r_{ui} adalah rating interaksi dari user u dan item i , yang dapat diuraikan menjadi dua matrik yaitu matrik $P_{m \times k}$ (k adalah fitur laten untuk user) atau p_u dan matrik $Q_{k \times n}$ (k adalah fitur laten untuk item) atau q_i . Untuk mempelajari fitur laten (p_u dan q_i) dapat diperoleh dengan rumus [7]

$$\min_{p_u, q_i} \sum_{u,i} (r_{ui} - q_i^T p_u)^2 + \lambda (\|p_u\|^2 + \|q_i\|^2) \quad (2)$$

Dimana pada persamaan (2) $\lambda \geq 0$ adalah parameter regularisasi yang ditambahkan agar matrik q dan p tidak mengalami overfitting terhadap matrix aslinya. Faktorisasi dapat dicapai dengan menggunakan algoritma Stochastic Gradient Descent atau Alternating Least Square seperti yang dijelaskan di penelitian [7].

Seperti yang diketahui, dalam penelitian ini dataset yang digunakan bersifat implisit, sedangkan dalam penelitian [2] yang membandingkan beberapa algoritma untuk dataset yang bersifat explisit dan implisit menyimpulkan bahwa algoritma Weighted Alternating Least Square menghasilkan akurasi yang lebih bagus untuk dataset yang bersifat implisit dibandingkan metode lainnya. Oleh karena itu pada penelitian ini akan menggunakan algoritma Weighted Alternating Least Square untuk menghasilkan rekomendasi.

2.4 Alternating Least Square (ALS)

Alternating Least Square merupakan algoritma dalam matrix factorization untuk meminimalisir persamaan (2). Sebelumnya p_u dan q_i belum diketahui nilainya, ALS bekerja dengan cara berotasi untuk memperbaiki salah satu p_u atau q_i yang belum diketahui. Secara umum algoritma ALS untuk collaborative filtering dari penelitian [12] sebagai berikut

Tabel 1. Alternating Least Square

Algoritma 1 Alternating Least Square

Inisialisasi matrik Q dengan angka acak kecil

for banyaknya iterasi **do**

 Hitung P dengan meminimalisir rumus (2) atau (3), Q bersifat tetap

 Hitung Q dengan miminalisir rumus (2) atau (3), P bersifat tetap

end

2.5 Weighted Alternating Least Square (W-ALS)

ALS pada umumnya digunakan untuk *feedback* yang bersifat explisit, oleh sebab itu Hu et al [5] mengusulkan W-ALS untuk beradaptasi dengan *feedback* yang bersifat implisit. W-ALS pada dasarnya adalah algoritma yang sangat mirip dengan ALS yang sudah dijelaskan pada bagian 2.4, akan tetapi yang menjadi pembeda adalah fungsi objektifnya. Dalam imlisit data terdapat tingkat ketertarikan user terhadap item, sebagai contoh user yang sering melihat sebuah hunian menunjukkan ketertarikan terhadap hunian tersebut. Oleh karena itu terdapat tambahan variabel baru yang disebut bobot (*Weighted*) pada rumus 2, sehingga menjadi

$$\min_{p_*, q_*} \sum_{u,i} c_{ui} (d_{ui} - q_i^T p_u)^2 + \lambda (\|p_u\|^2 + \|q_i\|^2) \quad (3)$$

Dimana dalam fungsi tersebut terdapat c_{ui} yang merupakan tingkat ketertarikan user terhadap item. Dalam penelitian [5] tingkat ketertarikan dihitung dengan rumus

$$c_{ui} = 1 + \alpha r_{ui} \quad (4)$$

Terdapat variabel α sebagai parameter untuk menghitung , sedangkan d_{ui} dalam persamaan (3) merupakan variabel biner yang mengindikasikan preferensi user u terhadap item i , d_{ui} diturunkan dari

$$d_{ui} = \begin{cases} 1 & r_{ui} > 0 \\ 0 & r_{ui} = 0 \end{cases} \quad (5)$$

Untuk menghasilkan performa sistem rekomendasi yang bagus, pengaturan hyperparameter untuk W-ALS diperlukan, parameter tersebut seperti banyaknya iterasi, alpha α , regularisasi λ , dan banyaknya faktor laten K pada matrix factorization. Banyaknya iterasi adalah seberapa banyak algoritma W-ALS berotasi secara bergantian, α adalah konstanta yang digunakan untuk mengontrol kenaikan tingkat ketertarikan c_{ui} , regularisasi λ merupakan konstanta agar tidak mengalami *overfitting*, sedangkan banyaknya faktor laten K merupakan baris dan kolumn pada matrix dekomposisi yang dibuat.

2.6 Content-Based Filtering

Teknik content-based filtering melibatkan interaksi user dan item, informasi tersebut akan dibuat sebuah model profil pengguna yang kemudian dicocokkan dengan atribut item. Semakin tinggi kecocokan profil pengguna dengan atribut item, maka kemungkinan item tersebut untuk direkomendasikan [6]. Pada penelitian ini sebelum dilakukan content-based filtering dataset hunian akan dilakukan pengelompokan berdasarkan lokasi hunian dengan k-means. Pengelompokan dilakukan agar hunian yang direkomendasikan sesuai dengan daerah atau kampus pengguna.

2.7 K-Means Clustering

K-Means merupakan salah satu metode data clustering non hirarki yang dapat mengelompokkan data ke dalam beberapa cluster berdasarkan kemiripan dari data tersebut [13]. Pada penelitian ini, k-means digunakan untuk

mengelompokkan data hunian berdasarkan lokasi hunian tersebut, sehingga data hunian yang memiliki karakteristik lokasi yang sama dikelompokkan dalam satu cluster dan yang memiliki karakteristik lokasi yang berbeda dikelompokkan dalam cluster yang lain.

2.8 Hybrid Recommender System

Hybrid Recommender System adalah teknik untuk menggabungkan dua atau lebih teknik rekomendasi untuk menghasilkan performa yang lebih baik [4]. Ada beberapa teknik dalam teknik hybrid diantaranya adalah 1) *weighted* dimana nilai beberapa rekomendasi digabungkan berdasarkan parameter tertentu. 2) *switching*, sistem digabungkan secara bergantian. 3) *Mixed*, hasil rekomendasi dari beberapa rekomendasi ditampilkan secara bersamaan. 4) *Feature Augmentation*, output sistem rekomendasi digunakan untuk rekomendasi lainnya. 5) *Meta-level*, model yang dipelajari dari suatu rekomendasi digunakan untuk rekomendasi lainnya. 6) *Cascade*, Suatu rekomendasi memperbaik hasil dari rekomendasi lain.

Pada penelitian ini penulis menggunakan pendekatan hybrid dengan teknik *weighted* dimana dapat dirumuskan dengan persamaan berikut [9]

$$R_{ui} = \sum_f^c \sigma_f R_{ui}^f \quad (6)$$

Dimana σ adalah bobot dari algoritma teknik rekomendasi R_{ui}^f . Namun karena pada penelitian ini hanya menggabungkan dua teknik yang berbeda maka persamaan bisa disimpulkan sebagai berikut

$$R_{ui} = \beta \cdot R^1 + (1 - \beta) \cdot R^2 \quad (7)$$

Dimana β adalah bobot, R^1 adalah collaborative filtering dan R^2 adalah content-based filtering .Pada penelitian kali ini model hybrid yang dibangun adalah menggabungkan teknik collaborative filtering dengan W-ALS dan teknik content-based filtering. Hasil rekomendasi tiap teknik rekomendasi akan dikalikan dengan bobot tertentu kemudian dijumlahkan untuk menghasilkan rekomendasi.

3. Sistem yang Dibangun

3.1 Desain Sistem

Sistem rekomendasi hunian untuk aplikasi Koseeker yang dibangun menggunakan pendekatan *Hybrid Recommender System*, adapun desain sistem dijelaskan pada tabel dibawah ini

Tabel 2. Desain Sistem

Desain Sistem
Input:
Dataset berupa impisit feedback dari aplikasi Koseeker
Dataset hunian dari aplikasi Koseeker
Output:
Daftar rekomendasi hunian berdasarkan preferensi user
Preprocessing Data:
Pembuatan aturan pembobotan terhadap feedback
Clustering data hunian berdasarkan lokasi dengan k-means
Buat data training dan data validasi
Algoritma:
Pembuatan model dengan W-ALS untuk collaborative filtering
Pembuatan model content-based filtering
Gabungkan collaborative dan content-based filtering dengan weighted hybrid
Evaluasi Performa:
Evaluasi dengan AUC dan presisi

Pada tabel 2 merupakan alur dari sistem yang dibangun, dimana sistem akan menghasilkan rekomendasi hunian berdasarkan preferensi pengguna, inputan dari sistem ini berupa data *implicit feedback* dan data hunian dari aplikasi koseeker. Proses rekomendasi dilakukan dengan menggabungkan collaborative dan content-based filtering dengan teknik *weighted*, namun sebelum dilakukan proses hybrid akan dilakukan pra pemrosesan terhadap datatset. Untuk mengevaluasi akurasi sistem rekomendasi yang dibangun akan digunakan metrik AUC dan presisi.

3.2 Dataset

Dataset yang digunakan dalam penelitian ini berasal dari aplikasi koseeker, terdapat dua dataset, pertama adalah data interaksi user dan item, dataset tersebut bersifat implisit yang berisikan data rekaman aktivitas pengguna koseeker selama lima bulan dari bulan januari 2020. Contoh dataset terdapat pada tabel 3.

Tabel 3. Cotoh dataset interaksi user-item koseeker

deve_id	user_id	property_id	event	total
1717f2ab2cd30....	5bee1d8ab45a153	5bee1d8ab45a153	view_property	4
1717fc6324514a...	5bee1d8ab45a154	5d595004dd42852	click_like_button	1
1717fc6324514a...	undefined	5bee1d8ab45a4534	veiw_property	2

Terdapat lima atribut dalam data yaitu *device_id* merupakan id unik setiap perangkat, *user_id* merupakan id user yang sudah terdaftar, *property_id* merupakan id dari hunian atau item, atribut *event* merupakan jenis interaksi antara user dan item, dan atribut total merupakan banyaknya interaksi user dan item. Dataset berisikan 10777 interaksi yang bersifat implisit, terdiri dari 1163 data user dan 612 data hunian. Data tersebut memiliki sparsity sebesar 0,98.

Dataset kedua adalah dataset hunian, contoh dataset terdapat pada tabel berikut ini:

Tabel 4. Cotoh dataset hunian koseeker

property_id	name	keyword	latitude	longitude
5bee1d8ab45a153	Aulia 2	kost yearly male	107.6308023	-6.969659403
5d595004dd42852	Baitul Thalib	kontrakan yearly male	107.6287501	-6.979461915
5bee1d8ab45a4534	Kost Bjs	kost monthly male	107.6351066	-6.9767124

Terdapat lima atribut dimana *property_id* merupakan id dari hunian atau item, *name* nama dari hunian, *keyword* adalah kata kunci atribut hunian, keyword ini yang akan digunakan untuk proses content-based filtering, sedangkan *latitude* dan *longitude* adalah koordinat lokasi hunian, proses pengelompokan berdasarkan lokasi dengan k-means menggunakan atribut korrdinat tersebut. Dataset hunian tersebut berisi 884 data hunian yang tesebar di beberapa daerah.

3.3 Preprocessing

Sebelum melakukan pemodelan W-ALS dibutuhkan sebuah matrik interaksi antara user dan item, oleh karena itu dataset akan dilakukan *preprocessing* untuk menghasilkan matrik tersebut. Dalam dataset terdapat atribut event dimana dalam kolumn tersebut terdapat beberapa jenis interaksi yang berbeda, setiap jenis interaksi memiliki tingkat *confidence* yang berbeda, oleh karena itu dibuatlah aturan pembobotan untuk setiap jenis interaksi sebagai berikut

Tabel 5. Aturan pembobotam

Nama event	Keterangan	Bobot
view_property	User melihat sebuah hunian	1.0
click_booking_button	User menekan tombol pemesanan	1.5
jadwalakan_survey	Akan melakukan survey ke lokasi	1.8
click_contact_landlord	Bertanya kepada pemilik hunian	1.5
click_like_button	Menyukai sebuah hunian	2.0
click_unlike_button	Batal menyukai hunian	-2.0
request_booking	Melakukan pemesanan hunian	3

Atribut bobot pada tabel 5 aturan pembobotan merupakan interpretasi jenis interaksi dan tingkat ketertarikan user terhadap sebuah hunian. Setelah dilakukan beberapa percobaan terhadap bobot setiap interaksi dengan menghitung AUC sebagai pembanding, maka bobot terbaik untuk setiap event adalah seperti pada tabel 5. Kemudian aturan bobot tersebut akan dikalikan dengan setiap baris pada kolumn *total_i* untuk menghasilkan bobot baru *W_i*, bobot baru tersebut yang akan digunakan dalam proses pembuatan model W-ALS. Selain itu dalam kolumn *user_id_i* yang berisikan nilai *undefined* akan diganti dengan *device_id_i*. Hingga akhirnya dataset menjadi seperti berikut ini

Tabel 6. Contoh dataset interaksi baru

user_id	property_id	bobot
5bee1d8ab45a153	5bee1d8ab45a153	4.0
5bee1d8ab45a154	5d595004dd42852	2.0
1717fc6324514a...	5bee1d8ab45a4534	2.0

Clustering dengan k-means dilakukan pada tahap preprocessing ini terhadap dataset hunian berdasarkan koordinat lokasi hunian, hasil cluster ini akan ditambahkan pada atribut keyword dalam dataset hunian untuk content-based filtering. k pada k-mean dipilih sebanyak 15 berdasarkan penyebaran data hunian di tiap daerah. Hasilnya dataset hunian akan menjadi seperti ini pada tabel 7.

Tabel 7. Cotoh dataset hunian baru setelah *clustering*

property_id	name	keyword	latitude	longitude
5bee1d8ab45a153	Aulia 2	cluster0 kost yearly male	107.6308023	-6.969659403
5d595004dd42852	Baitul Thalib	cluster2 kontrakan yearly male	107.6287501	-6.879461915
5bee1d8ab45a4534	Kost Bjs	cluster3 kost monthly male	110.6351066	-6.9767124

3.4 Rekomendasi Hunian

Pada penelitian ini sistem yang dibangun merupakan sistem rekomendasi hunian untuk aplikasi Koseeker. Teknik yang dipakai dalam penelitian ini adalah Collaborative Filtering dan Content-based Filtering. Pada Collaborative Filtering menggunakan teknik W-ALS untuk merekomendasikan item yang memiliki kesamaan dengan pengguna lain, sedangkan pada Content-based filtering akan dibuat profil user kemudian sistem akan merekomendasikan item yang cocok dengan profil user tersebut. Profil user berisi minat pengguna berdasarkan kampus/cluster, jenis hunian, durasi menginap, dan fasilitas hunian yang diminati. Hasil setiap teknik rekomendasi akan digabungkan dengan metode hybrid.

3.5 Rekomendasi Collaborative Filtering dengan W-ALS

Pada tahap ini sistem akan membuat model menggunakan W-ALS dengan inputan berupa matrik item-user, dimana baris pada matrik tersebut adalah atribut item (property_id), kolom adalah atribut pengguna yang menyukai item tersebut (user_id), dan isi dari matriknya adalah atribut bobot interaksi antara user dan item. Maka agar kompatibel, atribut user_id dan property_id dalam dataset yang berupa string harus diubah menjadi numerik. Model tersebut akan menghasilkan fitur laten yang akan digunakan untuk menghasilkan rekomendasi.

Tidak seperti matrix factorization pada dataset ekplisit yang memprediksi rating, model W-ALS untuk dataset yang bersifat implisit tidak memprediksi sebuah rating melainkan menghasilkan sebuah skor yang akan digunakan untuk menghasilkan rekomendasi berdasarkan ranking dari skor tersebut.

Tabel 8. Collaborative Filtering dengan W-ALS

Collaborative Filtering dengan W-ALS

Input:

Faktor laten user (P) dan item (Q) hasil dari model W-ALS

Output:

Daftar rekomendasi hunian

Algoritma:

Hitung skor R dengan rumus $R = P \times Q^T$ untuk setiap user-item iteraksi

Filter item yang sudah pernah berinteraksi dengan setiap user

Sorting item dari skor terbesar untuk dijadikan rekomendasi berdasarkan ranking

3.6 Rekomendasi Content-Based Filtering

Pada tahap ini atribut *keyword* pada dataset hunian akan dibuat menjadi sebuah matrik item-atribut dengan baris adalah item dan kolom adalah atribut pada item tersebut dan isi dari matrik tersebut adalah nilai biner (0/1) dimana

jika item memiliki atribut m maka akan bernilai 1 sedangkan jika tidak maka bernilai 0. profil user juga dibuat berdasarkan iteraksinya dengan item sebelumnya. Algoritma CB terdapat pada table 9,

Tabel 9. Content-based Filtering

Content-based Filtering

Input:

Dataset interaksi user dan item

Dataset hunian yang sudah dilakukan clustering

Output:

Daftar rekomendasi hunian

Algoritma:

Buat profil user berdasarkan interaksinya dengan item

Kalikan profil user dengan matrik item atribut kandidate rekomendasi

Filter item yang sudah pernah berinteraksi dengan setiap user

Sorting item dari skor terbesar untuk dijadikan rekomendasi berdasarkan ranking

3.7 Hybrid Recommender System

Pada tahap ini, dibangun sistem rekomendasi dengan pendekatan *hybrid* dengan menggunakan teknik *weighted*, dimana hasil rekomendasi collaborative filtering dan content-based filtering akan dikalikan dengan bobot tertentu kemudian dijumlahkan untuk menghasilkan rekomendasi.

Tabel 10. Hybrid Recommender System

Hybrid Recommender System

Input:

Hasil skor collaborative filtering dan content-based filtering

Output:

Daftar rekomendasi hunian

Algoritma:

Jumlahkan collaborative dan content-based filtering dengan setiap bobot yang diberikan

Filter item yang sudah pernah berinteraksi dengan setiap user

Sorting item dari skor terbesar untuk dijadikan rekomendasi berdasarkan ranking

3.8 Skenario Pengujian Rekomendasi

Untuk mengukur kualitas sistem rekomendasi, digunakan perhitungan *Area Under Curve* (AUC). AUC dapat digunakan sebagai ukuran tunggal untuk keseluruhan kualitas sistem rekomendasi [11]. Sebagai perbandingan, performa sistem rekomendasi dengan *Hybrid Recommender System* akan dibandingkan dengan performa rekomendasi yang hanya berdasarkan satu teknik rekomendasi.

Untuk menguji seberapa relevan item yang direkomendasikan, dilakukan pengujian *Top N Recommendation* dengan nilai N 5 sampai 60 untuk mengevaluasi performa Top N rekomendasi yang diberikan oleh sistem, penulis menggunakan perhitungan presisi untuk mengukur seberapa banyak item yang relevan direkomendasikan oleh sistem. Pengujian Top N juga akan dilakukan dengan data training yang lebih sedikit. *Confusion matrix* digunakan sebagai dasar evaluasi rekomendasi dengan tabel sebagai berikut

Tabel 11. Confusion Matrix

	Relevan	Tidak Relevan	Total
Direkomendasikan	tp	fp	$tp + fp$
Tidak direkomendasikan	fn	tn	$fn + tn$
Total	$tp + fn$	$fp + tn$	N

Nilai presisi dapat dihitung dengan persamaan [11]

$$presisi = \frac{tp}{tp + fp} \quad (8)$$

Sedangkan nilai AUC merupakan luas area dari grafik ROC (Receiver Operating Characteristics), semakin tinggi nilai AUC maka sistem yang dibangun semakin bagus. AUC dapat dihitung dengan persamaan berikut [11]

$$AUC = \int TPR d(FPR) \quad (9)$$

Dimana TPR merupakan sumbu y dan FPR merupakan sumbu x dari grafik ROC yang dihitung dengan persamaan berikut [11]

$$TPR = \frac{tp}{tp + fn} \quad (10)$$

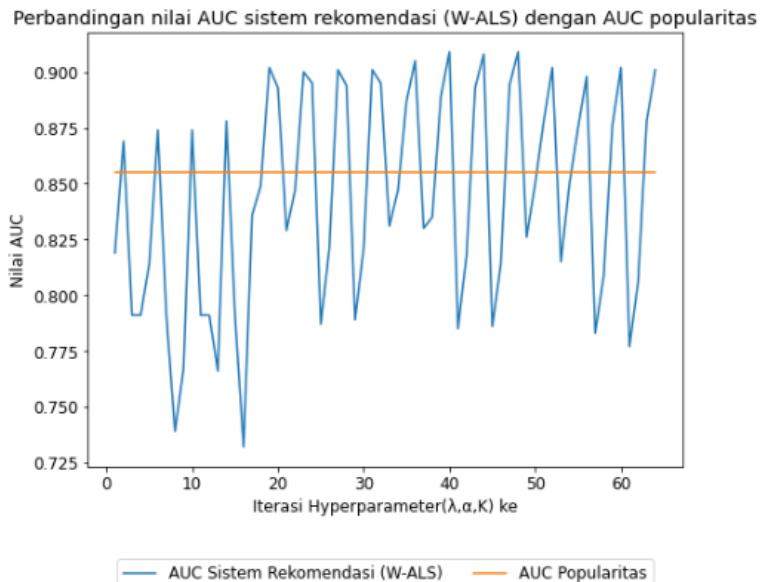
$$FPR = \frac{fp}{tp + tn} \quad (11)$$

Dalam pengujian ini dataset akan dibagi menjadi data training dan data testing. Dalam data training sebagian interaksi antara user dan item akan disembunyikan sebanyak 20% secara acak. Dengan begitu dapat diketahui seberapa banyak item yang direkomendasikan yang sebenarnya disukai oleh user.

4. Evaluasi

4.1 Hasil Pengujian

Setelah dilakukan pengujian *hyperparameter* terhadap dataset dengan menyembunyikan 20% data sebagai data training, skor AUC terbaik untuk teknik collaborative filtering dengan model W-ALS adalah 0.90 dengan parameter terbaik $\alpha = 20$, $\lambda = 100$, $K = 32$ dan iterasi 20 kali. Skor AUC model yang dibangun berhasil mengalahkan AUC dari popularitas item dimana hanya sebesar 0.85. Hasil percobaan dapat dilihat pada gambar 2 dibawah ini

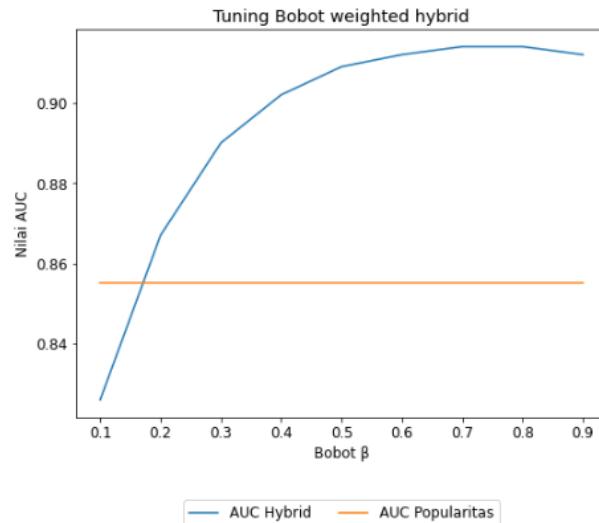


Gambar 2. percobaan hyperparameter model W-ALS

Sedangkan skor AUC terbaik untuk model hybrid yang dibagun adalah 0.914 dengan bobot untuk teknik collaborative filtering adalah 0.7 dan bobot untuk content-based filtering adalah 0.3, bobot tersebut diperoleh setelah dilakukan percobaan dengan beberapa bobot seperti terlihat pada gambar 3. Hasil tersebut lebih baik daripada sistem rekomendasi yang hanya menggunakan satu teknik rekomendasi, hasilnya dapat dilihat pada tabel 12.

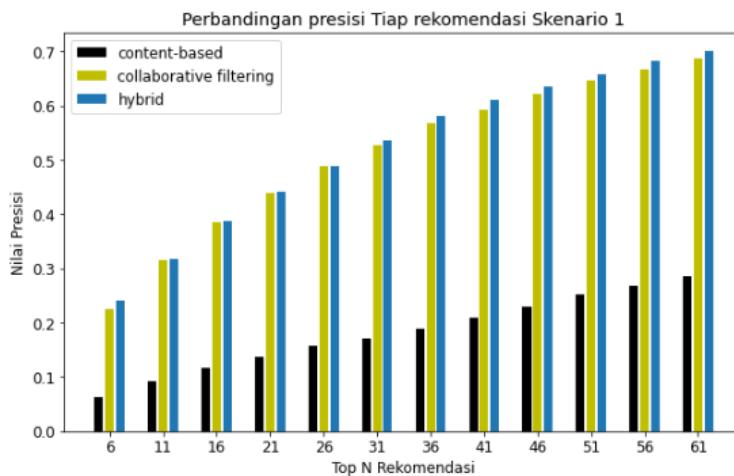
Tabel 12. Skor AUC dan Presisi tiap teknik rekomendasi

Model	AUC	Rata-rata presisi skenario 1	Rata-rata presisi skenario 2
Content-based filtering	0.77	0.18	0.15
Collaborative filtering	0.90	0.51	0.38
Hybrid	0.914	0.52	0.39



Gambar 3. perbobaan dengan nilai bobot berbeda

Untuk pengujian presisi pada *Top N Recommendation* dengan nilai N 5 sampai 60 dibagi menjadi dua skenario. Skenario pertama adalah menghitung nilai presisi dengan menggunakan data training interaksi user-item asli dan skenario kedua menghitung nilai presisi hanya dengan menggunakan setengah dari data training interaksi user-item yang asli sebagai pembandingnya. Pada pengujian ini sebanyak 20% interaksi setiap user pada data training disembunyikan untuk dijadikan data testing, hasilnya dapat dilihat pada gambar 4 dan 5 dibawah ini

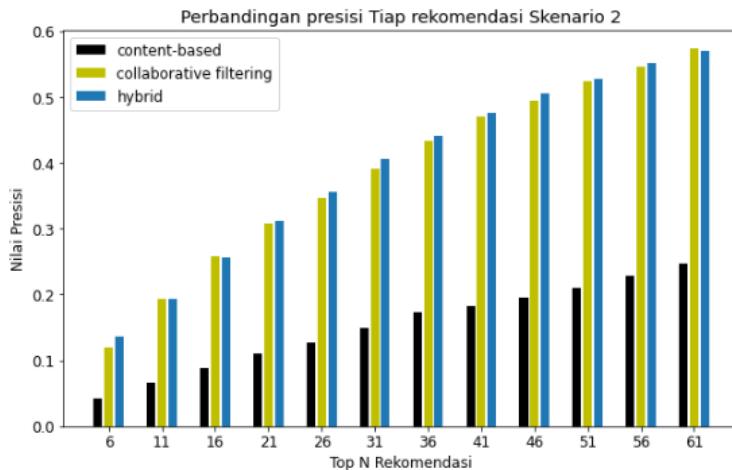


Gambar 4. perbandingan nilai presisi setiap teknik rekomendasi skenario 1

4.2 Analisis Hasil Pengujian

Berdasarkan percobaan pada gambar 2, dapat dilihat bahwa nilai parameter pada teknik collaborative filtering dengan model W-ALS yang dibangun berpengaruh terhadap nilai AUC yang dihasilkan. Dimana pada percobaan tersebut nilai AUC tertinggi adalah 0.90 dengan parameter $\alpha = 20$, $\lambda = 100$, $K = 32$ dan iterasi 20 kali, sedangkan nilai AUC terendah adalah 0.73 dengan parameter $\alpha = 1$, $\lambda = 100$, $K = 64$ dan iterasi 20 kali. Dapat disimpulkan bahwa dengan parameter yang tepat maka performa rekomendasi akan semakin bagus. Model W-ALS yang dibangun berhasil mengalahkan rekomendasi berdasarkan popularitas item dimana AUC berdasarkan popularitas item adalah 0.85.

Berdasarkan percobaan pada tabel 12, nilai AUC untuk teknik content-based filtering hanya sebesar 0.77, hal ini dikarenakan content-based filtering merekomendasikan item yang memiliki atribut yang sama, sedangkan dalam dataset interaksi user-item hanya terdapat 612 item dari total 884 item, itu artinya terdapat 272 item yang belum pernah berinteraksi dengan user. Model *hybrid* yang dibagun dengan menggabungkan collaborati-



Gambar 5. perbandingan nilai presisi setiap teknik rekomendasi skenario 2

ve dan content-based filtering dapat menghasilkan nilai AUC sebesar 0.91, model hybrid berhasil mengalahkan rekomendasi dengan satu teknik saja, hal ini dikarenakan model hybrid yang dibangun melengkapi kekurangan collaborative filtering dan content-based filtering.

Pada pengujian *Top N Recommendation* dengan nilai N 5 sampai 60, rata-rata nilai presisi untuk teknik content-based pada skenario pertama dan kedua yaitu 0.18 dan 0.15, terjadi penurunan namun tidak signifikan, hal ini menunjukkan content-based filtering tidak terlalu terpengaruh terhadap dataset interaksi yang lebih sedikit, karena content-based menghasilkan rekomendasi berdasarkan kesamaan item yang sudah pernah dirating atau berinteraksi, bukan berdasarkan iteraksi oleh user lain. Sedangkan pada teknik collaborative filtering terjadi penurunan nilai rata-rata presisi, dimana rata-rata presisi pada skenario pertama sebesar 0.51 sedangkan rata-rata presisi pada skenario kedua adalah 0.38, hal ini bisa terjadi karena collaborative filtering menghasilkan rekomendasi berdasarkan user lain yang memiliki prefensi yang sama. Hal yang sama terjadi dengan model hybrid yang dibangun, dimana juga mengalami penurunan presisi meskipun masih unggul namun tidak signifikan, hal ini dikarenakan model hybrid yang dibangun merupakan gabungan dari kedua teknik diatas. Dengan begitu dapat disimpulkan bahwa semakin banyak dataset interaksi user item maka performa sistem rekomendasi akan semakin bagus.

5. Kesimpulan

Berdasarkan hasil analisis dan pengujian pada penelitian ini dapat disimpulkan bahwa, model *hybrid* yang dibangun dengan menggabungkan collaborative dan content-based filtering dengan teknik *weighted* dapat digunakan untuk membangun sistem rekomendasi hunian pada aplikasi Koseeker. Dengan dataset interaksi user-item sebanyak 10777, dataset item sebanyak 884, bobot untuk collaborative filtering 0.7 dan content-based filtering 0.3, model hybrid dapat mengalahkan rekomendasi yang hanya menggunakan teknik collaborative atau content-based filtering saja dengan nilai AUC sebesar 0.914, sedangkan AUC untuk teknik collaborative dan content-based filtering adalah 0.90 dan 0.77.

Pada pengujian *Top N Recommendation*, dapat disimpulkan bahwa semakin banyaknya dataset maka sistem rekomendasi dengan model *hybrid* yang dibangun akan semakin bagus. Oleh karena itu saran untuk penelitian selanjutnya adalah dengan menggunakan dataset yang lebih banyak. Pada penelitian ini, bobot pada teknik hybrid adalah bersifat statis dimana bobot untuk menggabungkan teknik collaborative dan content-based filtering sudah ditentukan sebelumnya, saran untuk penelitian selanjutnya adalah bobot bersifat dinamis, dimana setiap user akan memiliki aturan pembobotan yang berbeda-beda bergantung pada aktivitas dan preferensinya. Selain itu dapat juga dilakukan pengujian *A/B Testing* untuk mengevaluasi sistem rekomendasi terhadap pengguna maupun proses bisnis aplikasi Koseeker.

Daftar Pustaka

- [1] Tertinggi sejak 1997, jumlah mahasiswa indonesia 2018 capai 7 juta jiwa. <https://databoks.katadata.co.id/datapublish/2019/09/26/tertinggi-sejak-1997-jumlah-mahasiswa-indonesia-2018-capai-7-juta-jiwa>. Online; Accessed November 2019.

- [2] C. R. Aberger and caberger. Recommender : An analysis of collaborative filtering techniques. 2014.
- [3] M. I. Ardimansyah, A. F. Huda, and Z. K. A. Baizal. Preprocessing matrix factorization for solving data sparsity on memory-based collaborative filtering. In *2017 3rd International Conference on Science in Information Technology (ICSTech)*, pages 521–525, 2017.
- [4] R. Burke. Hybrid recommender systems: Survey and experiments. *User Modeling and User-Adapted Interaction*, 12(4):331–370, 2002.
- [5] Y. Hu, Y. Koren, and C. Volinsky. Collaborative filtering for implicit feedback datasets. In *2008 Eighth IEEE International Conference on Data Mining*, pages 263–272, 2008.
- [6] L. Iaquinta, M. de Gemmis, P. Lops, G. Semeraro, M. Filannino, and P. Molino. Introducing serendipity in a content-based recommender system. pages 168–173, 10 2008.
- [7] Y. Koren, R. Bell, and C. Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009.
- [8] A. Kumar, S. Gupta, S. K. Singh, and K. K. Shukla. Comparison of various metrics used in collaborative filtering for recommendation system. In *2015 Eighth International Conference on Contemporary Computing (IC3)*, pages 150–154, 2015.
- [9] W. Lin, Y. Li, S. Feng, and Y. Wang. The optimization of weights in weighted hybrid recommendation algorithm. In *2014 IEEE/ACIS 13th International Conference on Computer and Information Science (ICIS)*, pages 415–418, 2014.
- [10] F. Ricci, L. Rokach, and B. Shapira. *Recommender Systems Handbook*, volume 1–35, pages 1–35. 10 2010.
- [11] G. Schröder, M. Thiele, and W. Lehner. Setting goals and choosing metrics for recommender system evaluations. 811, 01 2011.
- [12] G. Takács and D. Tikk. Alternating least squares for personalized ranking. 09 2012.
- [13] P.-N. Tan, M. Steinbach, and V. Kumar. *Introduction to Data Mining, (First Edition)*. Addison-Wesley Longman Publishing Co., Inc., USA, 2005.