

Aplikasi Marketplace Penyewaan Barang Menggunakan Algoritma Slope One Sebagai Rekomendasi Kepada Pengguna

Akbar Serdano¹, Muhammad Ibrahim², Vicky Romadon³

¹Fakultas Sains dan Teknologi, Universitas Sumatera Utara Medan, Indonesia,

²Bank Rakyat Indonesia (BRI) Pusat Medan, Indonesia,

³Development Back End Medan, Indonesia

¹akbar.serdano@uinsu.ac.id, ²ibrahimmuhammad11@gmail.com, ³vickyromadon46@gmail.com

Abstrak— Menyewa barang merupakan solusi untuk mendapatkan barang yang diinginkan tanpa harus dibeli. Dikarenakan alasan tertentu banyak orang yang memilih untuk menyewa barang daripada harus membelinya. Namun penyewa terkadang kesulitan dalam menemukan tempat menyewa barang yang dicari serta pemilik barang kekurangan tempat untuk menyewakan barang yang dimilikinya. Untuk membantu penyewa dan pemilik barang maka perlu dibuat suatu Marketplace yang dilengkapi dengan sistem rekomendasi produk. Sistem rekomendasi tersebut menggunakan algoritma Slope One. Hasil dari aplikasi menunjukkan bahwa algoritma Slope One cukup sesuai memberikan rekomendasi. Kecenderungan yang ada yaitu semakin banyak rating diberikan terhadap produk maka nilai MAE yang dihasilkan semakin kecil.

Kata Kunci— Marketplace, Recommender System, Slope One, MAE.

Abstract— Renting goods is a solution to get the desired item without having to be purchased. Due to certain reasons many people choose to rent goods rather than have to buy them. But the problem is that tenants are hard to find the place to rent the item they are looking for and the owner of the item lacks a place to market the goods he owns. To help tenants and property owners, it is necessary to create a Marketplace equipped with a product recommendation system. The system uses Slope One algorithm. The results of the application show that the Slope One algorithm is quite appropriate to provide recommendations. The existing trend is that the more rating is given to the product, the smaller the MAE value.

Keywords— Marketplace, Recommender System, Slope One, MAE.

I. PENDAHULUAN

Sewa-menyewa berasal dari kata “sewa” dan “menyewa”. Kata “sewa” berarti pemakaian sesuatu dengan membayar uang sedangkan kata “menyewa” berarti memakai dengan membayar uang sewa[1]. Berdasarkan hasil kuesioner yang telah dilakukan didapatkan hasil berupa, sebagian orang yang memiliki barang tidak terpakai memilih untuk tidak menjualnya dan berkeinginan untuk menyewakan barang tersebut. Begitu juga sebaliknya dikarenakan alasan tertentu sebagian orang memilih untuk menyewa barang yang diinginkannya daripada membeli barang tersebut. Oleh karena itu Marketplace penyewaan barang menjadi suatu alternatif untuk mempertemukan antara pencari barang dan pemilik barang. Saat ini ada beberapa Marketplace yang bergerak dalam bidang sewa menyewa seperti Rentongo, Kwipped namun aplikasi Marketplace tersebut hanya melayani model bisnis B2B (Business To Business) sehingga perorangan yang tidak menggunakan model B2B tidak dapat melakukan transaksi. Selain itu pada umumnya pemilik barang takut untuk menyewakan barangnya

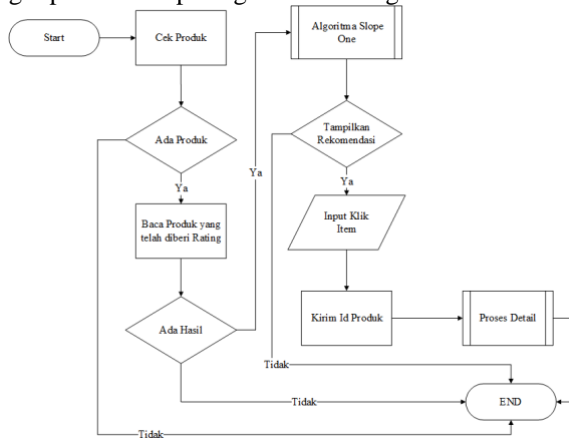
dikarenakan pemilik barang khawatir barangnya hilang atau terjadi kerusakan saat disewakan. Pengguna biasanya juga akan menggunakan lokasi terdekat dari pengguna sebagai pertimbangan dalam mencari barang yang akan pengguna sewa. Selain itu, pilihan produk yang beragam juga terkadang membuat pengguna kesulitan dalam menentukan pilihan. Pengguna akan lebih mudah membuat pilihan jika sistem dapat memberikan rekomendasi kepada pengguna berdasarkan rating atau ulasan dari produk tersebut. Sistem rekomendasi menyediakan sebuah informasi berupa daftar item-item yang sesuai keinginan pengguna[2].

Untuk memberikan solusi dari permasalahan di atas maka akan dikembangkan sebuah aplikasi Marketplace penyewaan barang yang dapat digunakan oleh perorangan untuk melakukan proses sewa menyewa barang yang akan dilengkapi dengan fitur seperti kemampuan sistem men-generate surat pernyataan yang dapat digunakan oleh pemilik barang sebagai bukti, bahwa barang miliknya disewa oleh orang yang bersangkutan. Fitur filtering produk

berdasarkan lokasi terdekat dari pengguna juga akan disematkan pada aplikasi ini agar pengguna mendapatkan pilihan produk berdasarkan jarak terdekat dari pengguna. Selain itu, aplikasi juga akan dilengkapi suatu sistem rekomendasi untuk memberikan rekomendasi produk yang relevan berdasarkan popularitas dari suatu produk. Sistem rekomendasi yang digunakan pada aplikasi ini adalah algoritma *Slope One*. Algoritma *Slope One* memberikan prediksi berdasarkan nilai hasil pencarian dari item-item yang dibandingkan. Keunggulan algoritma *Slope One* dibandingkan algoritma rekomendasi lainnya adalah algoritma *Slope One* mudah untuk diimplementasi, efisien saat melakukan query, tidak memerlukan banyak requirement dikarenakan rekomendasi berdasarkan *rating* dari setiap item, dan cukup akurat[3].

II. METODE PENELITIAN

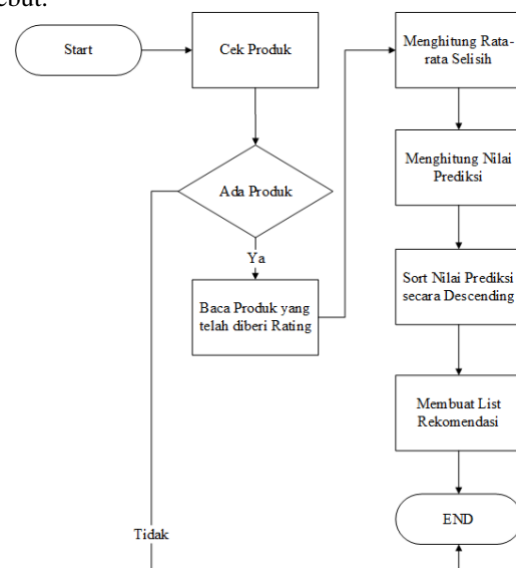
Metode penelitian yang digunakan dalam penelitian ini mencakup cara kerja algoritma *Slope One* dalam fitur rekomendasi produk untuk memberikan urutan *list* produk terbaik berdasarkan *rating*, yang diberikan pengguna. Penjelasan lebih lanjut mengenai alur proses yang dilakukan dalam aplikasi sistem rekomendasi ini dijelaskan dengan *flowchart*, seperti yang diperlihatkan pada gambar 1 dan gambar 2.



Gambar 1. *Flowchart* Sistem Rekomendasi Produk pada saat Membuka Aplikasi *Marketplace* Penyewaan Barang

Pada gambar di atas, aplikasi ini akan digunakan oleh *user* dan pada saat *user* membuka aplikasi penyewaan barang, maka aplikasi akan langsung melakukan pengecekan apakah produk tersedia pada. Jika produk tersedia, maka aplikasi akan langsung membaca produk yang tersedia pada aplikasi penyewaan barang. Setelah melakukan request, aplikasi akan menerima hasil berupa JSON (JavaScript Object Notation). JSON akan di-parsing dan dimasukkan ke dalam *arraylist*. Setelah data di-parsing,

data akan diproses dengan algoritma *Slope One*, sehingga diperoleh *arraylist* rekomendasi. Setelah perhitungan rekomendasi selesai dilakukan, akan ditampilkan halaman utama dari aplikasi. Jika salah satu item di-klik oleh *user*, maka detail dari produk tersebut akan ditampilkan. Detail berupa deskripsi produk, foto-foto, harga, ulasan, dan *rating*. Selain itu, aplikasi juga bisa mencari produk berdasarkan nama dan radius dari lokasi sekitar. Jika ada hasil, maka akan ditampilkan *list* dengan nama yang dicari dan jika di-klik maka akan dimunculkan detail dari produk tersebut.



Gambar 2. *Flowchart* Algoritma *Slope One* pada saat Membuka Aplikasi *Marketplace* Penyewaan Barang

Pada gambar di atas menunjukkan alur algoritma *Slope One* yang diterapkan dalam aplikasi. Algoritma *Slope One* digunakan untuk memberikan rekomendasi produk berdasarkan tingkat popularitas. Proses perhitungan algoritma *Slope One* dimulai dengan mengecek produk yang ada pada aplikasi. Langkah berikutnya adalah menghitung rata-rata selisih dari nilai setiap produk dengan nilai produk lainnya yang dimasukkan ke dalam *arraylist*. Setelah mendapatkan rata-rata selisih dari setiap produk yang ada, selanjutnya menghitung nilai prediksi untuk mendapatkan poin *Slope One*. *Arraylist* akan diurutkan sesuai dengan poin *Slope One* secara *descending*[4].

Contoh implementasi algoritma *Slope One* pada sistem rekomendasi penyewaan barang dengan ilustrasi 6 buah data produk dan 6 pengguna sesuai dengan tabel 1.

Tabel 1. *Rating* Enam Buah Produk Oleh Enam Pengguna

	A	B	C	D	E	F
Dina	0	5	0	0	0	0
Ahmad		0	3	3	4	2
Budi	0	3	0	0	0	3
Erika	1	2	0	2	0	4
Septina	0	2	2	4	0	4
Ani	1	?	3	?	?	5

Tabel 2. Nilai Rata-rata *Rating* Antar Produk

	A	B	C	D	E	F
A	0	1	2	1,5	3	2,66667
B	-1	0	0	1	0	1,33333
C	-2	0	0	1	1	1
D	-1,5	-1	-1	0	1	0,33333
E	-3	0	-1	-1	0	-2
F	-2,66667	-1,33333	-1	-0,33333	2	0

1.

Berdasarkan tabel 1 akan dicari urutan rekomendasi untuk pengguna Ani dari tiga produk yang belum di *rating*. Langkah-langkah perhitungan urutan rekomendasi dari tiga produk menggunakan Algoritma *Slope One* yaitu sebagai berikut:

1. Menghitung Rata-rata Selisih *Rating* Antar Produk Berdasarkan Data *Rating*

Untuk menghitung rata-rata selisih *rating* antar produk berdasarkan data *rating* yang ada pada tabel 1 menggunakan rumus sebagai berikut. [5]

$$dev_{j,i} = \sum_{u \in S_{j,i}(x)} \frac{u_j - u_i}{card(S_{j,i}(x))} \quad (1)$$

Contoh penghitungan rata-rata selisih *rating* menggunakan produk A dan produk C.

Keterangan:

u adalah Ahmad dan Ani

u_j = *rating* untuk produk A

u_i = *rating* untuk produk C

$card(S_{j,i}(X))$ = jumlah pelanggan yang memberi *rating* pada item i dan item j

Nilai rata-rata selisih *rating* antar produk A dan produk C adalah:

$$\frac{1 - 3}{2} + \frac{1 - 3}{2} = \frac{-4}{2} = -2$$

Tabel 2 memperlihatkan nilai rata-rata selisih *rating* antar produk berdasarkan nilai *rating* yang diberikan oleh pengguna. Nilai pada tabel 2 menunjukkan semakin besar nilai secara absolut berarti perbedaan *rating* antar *user* semakin berbeda.

2. Menghitung Nilai Prediksi Berdasarkan Data Rata-rata Selisih *Rating*

Untuk menghitung nilai prediksi terhadap data rata-rata selisih pada tabel 1 dan tabel 2 menggunakan rumus sebagai berikut.

$$P_{(u_j)} = \frac{1}{card(R_j)} \sum_{i \in R_j} (dev_{j,i} + u_i) \quad (2)$$

Nilai prediksi yang akan dicari adalah nilai prediksi produk-produk yang akan direkomendasikan kepada pengguna Ani, yaitu nilai prediksi produk B, produk D, dan produk F. Sehingga dari nilai prediksi ini dapat dibuat rekomendasi untuk pengguna Ani.

Nilai Prediksi produk B, dibantu oleh pengguna Budi yang telah memberi *rating* produk F. Pengguna Erika yang telah memberi *rating* produk A dan produk F. Pengguna Septina yang telah memberi *rating* produk C dan produk F.

$$\begin{aligned} P(B) &= \frac{1}{5} \times ((-1,36666 + 5) + (1 + 1) + (-1,36666 + 5) + (0 + 3) \\ &\quad + (-1,36666 + 5)) \\ &= \frac{1}{5} \times (3,6666 + 2 + 3,6666 + 3 + 3,6666) \\ &= 3,2 \end{aligned}$$

Nilai Prediksi produk D, dibantu oleh pengguna Ahmad yang telah memberi *rating* produk A, produk C, dan produk F. Pengguna Erika yang telah memberi *rating* produk A dan produk F. Pengguna Septina yang telah memberi *rating* produk C dan produk F.

$$\begin{aligned} P(C) &= \frac{1}{6} \times ((1,5 + 1) + (1 + 3) + (-0,3333 + 5) + (1,5 + 1) + (-0,3333 + 5) \\ &\quad + (1 + 3) + (-0,3333 + 5)) \\ &= \frac{1}{6} \times (2,5 + 4 + 4,6666 + 2,5 + 4,6666 + 4 + 4,6666) \\ &= 3,8571 \end{aligned}$$

Nilai Prediksi produk E, dibantu oleh pengguna Ahmad yang telah memberi *rating* produk A, produk C, dan produk F.

$$\begin{aligned}
 P(E) &= \frac{1}{3} \times ((3 + 1) + (1 + 3) + (2 + 5)) \\
 &= \frac{1}{3} \times (4 + 4 + 7) \\
 &= 5
 \end{aligned}$$

Tabel 3. Nilai Prediksi Produk B, D, E untuk pengguna Ani

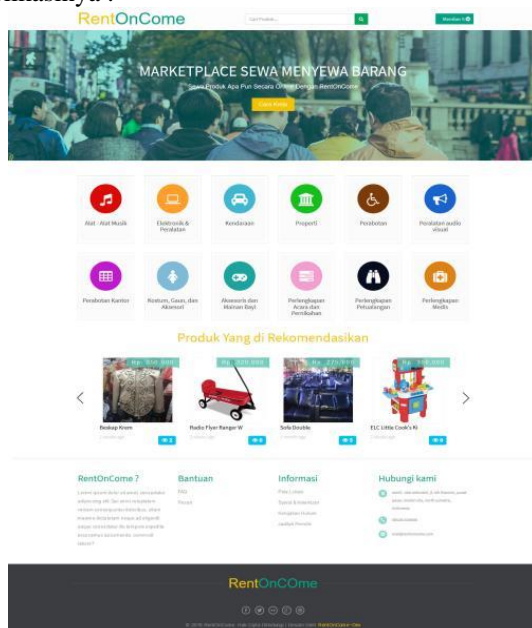
	Nilai Prediksi
B	3,2
D	3,8571
E	5

3. Urut Nilai Prediksi Berdasarkan *Descending*
 Dari nilai prediksi pada tabel 3, urutan *list* rekomendasi produk untuk pengguna Ani, yaitu:
 - a. Produk E
 - b. Produk D
 - c. Produk B

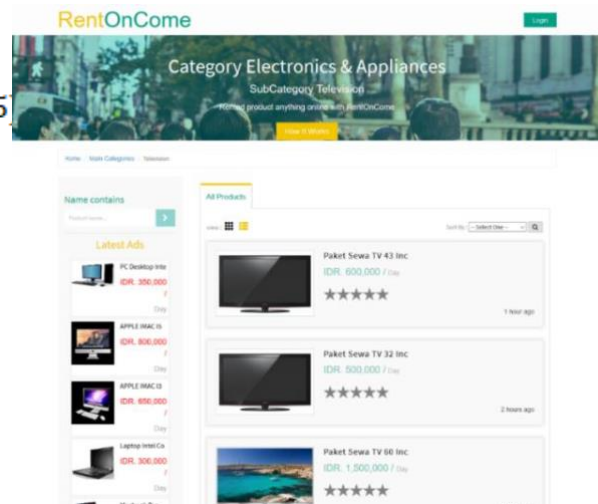
III. HASIL DAN PEMBAHASAN

A. HASIL

Hasil yang didapatkan berupa aplikasi *Marketplace* sewa menyewa barang menggunakan algoritma *Slope One* sebagai rekomendasi, berikut hasil tampilan aplikasinya :



Gambar 3. Tampilan Home Beserta Rekomendasi



Gambar 4. Tampilan Daftar Produk Sesuai Kategori

B. PEMBAHASAN

Pengujian yang dilakukan terhadap Algoritma *Slope One* menggunakan MAE (*Mean Absolute Error*) untuk menentukan seberapa baik algoritma yang diimplementasikan ke dalam aplikasi *Marketplace* penyewaan barang. *Mean Absolute Error* adalah metode untuk mengukur tingkat akurasi atau besar error hasil prediksi *rating* dari sistem terhadap *rating* yang sebenarnya yang *user* berikan terhadap suatu item. MAE diperoleh dengan menghitung nilai *absolute* dari N pasang *rating* asli dan prediksi, kemudian menghitung rata-ratanya[6]. Untuk menghitung MAE maka digunakan rumus sebagai berikut :

$$MAE = \frac{1}{n} \sum_{i=1}^n |f_i - y_i| \quad (3)$$

Keterangan :

f_i = merupakan hasil prediksi

y_i = nilai *rating* sebenarnya

n = jumlah data

Rekomendasi produk yang diberikan oleh sistem didasarkan pada nilai prediksi *rating* produk. Untuk dapat melihat sejauh mana keakuratan rekomendasi yang diberikan, maka perlu dilakukan pengujian terhadap nilai prediksi yang dihasilkan. Pengujian menggunakan *Mean Absolute Error* (MAE) untuk menghitung rata-rata error dari nilai prediksi yang dihasilkan. Pengujian dilakukan terhadap 78 produk yang diambil dari tabel product dengan total tenant sebanyak 10 orang yang diambil dari tabel *user* serta *rating* produk yang diambil dari tabel *ratings*. *Rating*

diberikan secara acak oleh setiap tenant beserta jumlah produk yang di *rating* bervariasi. Nilai MAE pada tahapan pengujian menggunakan persamaan (3) dimana nilai prediksi *rating* produk yang dihasilkan, dikurangkan dengan nilai *rating* yang sebenarnya. Nilai *rating* yang sebenarnya didasarkan pada rata-rata *rating* dari keseluruhan tenant yang telah merating item tersebut.

Untuk memudahkan dalam menarik kesimpulan dari hasil pengujian yang dilakukan, maka hasil pengujian disajikan dalam bentuk tabel. Hasil pengujian terdapat pada tabel 4.

Tabel 4. Hasil Pengujian MAE

User	Jumlah Item Yang Dirating	Jumlah Item Yang Diprediksi	MAE
User Id 1	27	51	0.951
	48	30	0.142
User Id 4	20	58	0.424
	58	20	0.384
User Id 5	30	48	0.767
	40	38	0.509

Berdasarkan pengujian terhadap *user id 1*, *user id 4*, dan *user id 5* diatas, dapat disimpulkan bahwa banyaknya *rating* yang digunakan sebagai bahan prediksi mempengaruhi keakuratan prediksi. Semakin banyak *rating* yang digunakan untuk memprediksi maka keakuratan prediksi semakin baik. Hal ini terlihat dari hasil MAE yang relatif menurun jika jumlah *rating* semakin banyak.

IV. KESIMPULAN

Berdasarkan pengujian yang dilakukan maka diperoleh kesimpulan yang didapat adalah sistem rekomendasi yang diterapkan pada aplikasi *Marketplace* sewa menyewa barang menggunakan algoritma *Slope One* yang telah dilakukan dengan menggunakan metode MAE didapatkan hasil semakin banyak *rating* yang diberi maka nilai MAE yang dihasilkan akan relatif kecil atau mendekati 0 yang berarti algoritma *Slope One* memiliki keakuratan yang cukup baik.

REFERENSI

- [1] D. Alwi, Hasan, "Kamus Besar Bahasa Indonesia Edisi Ketiga," *Jakarta PT. Balai Pustaka*, pp. 17–49, 2005.
- [2] L. Sebastia, I. Garcia, E. Onaindia, and C. Guzman, "E-Tourism: A tourist recommendation and planning application," *Int. J. Artif. Intell. Tools*, vol. 18, no. 5, pp. 717–738, 2009, doi: 10.1142/S0218213009000378.

- [3] T. JIANG and W. LU, "Improved Slope One Algorithm Based On Time Weight," 2013, doi: 10.2991/icsee.2013.575.
- [4] D. Lemire and A. Maclachlan, "Slope one predictors for online rating-based collaborative filtering," in *Proceedings of the 2005 SIAM International Conference on Data Mining, SDM 2005*, 2005, pp. 471–475, doi: 10.1137/1.9781611972757.43.
- [5] E. Vozalis and K. Margaritis, "Analysis of Recommender Systems' Algorithms," *Hercma*, pp. 1–14, 2003.