

PENGEMBANGAN VIRTUAL FASHION ROOM BERBASIS WEB

Gunawan¹⁾, Fandi Halim²⁾, Erwin Lawrence³⁾

^{1,2} Program Studi S-1 Sistem Informasi, STMIK Mikroskil

³ Program Studi S-1 Teknik Informatika, STMIK Mikroskil

Jl. Thamrin No. 140, Medan, 20212

Telp.: (+6261)4573767, Fax: (+6261) 4567789

E-mail: gunawan@mikroskil.ac.id¹⁾, fandi@mikroskil.ac.id²⁾, lawrencemail@rocketmail.com³⁾

Abstrak

Butik merupakan salah satu media bagi pecinta fashion untuk mengikuti perkembangan fashion. Sebelum memutuskan untuk membeli, pecinta fashion biasanya mencoba, membandingkan, serta mencari informasi, baik dari majalah, penjual, ataupun konsumen lainnya, sehingga sangat menyita waktu dan kebanyakan dari mereka tidak memiliki waktu ke butik. Pengembangan sistem memperbolehkan pelanggan untuk mencoba produk secara virtual dengan adanya penerapan virtual simulation serta sistem rekomendasi produk menggunakan algoritma slope one scheme.

Kata kunci: virtual, sistem rekomendasi, slope one scheme, fashion

Abstract

Boutique is one of the media for fashion lovers to keep on fashion trends. Before decide to buy, they usually try, compare, and search for products information, either from magazine, seller, or other consumers, hence, it is quite a time consuming and most of the customers do not have time to go to boutique. System development allow customers to try the products virtually with the use of virtual simulation and products recommendation system using slope one scheme algorithm.

Keywords: virtual, recommendation system, slope one scheme, fashion

1. PENDAHULUAN

Memilih produk *fashion* tentu saja merupakan hal yang sangat rumit dan membutuhkan waktu yang cukup banyak, apalagi dengan variatifnya produk. Banyak faktor yang harus dipertimbangkan dalam kegiatan tersebut, seperti kesesuaian antara produk *fashion* yang dipilih, produk yang sedang diminati pelanggan pada umumnya, dan harga produk dari *vendor* yang berbeda. Jika terdapat sebuah sistem yang dapat membantu merekomendasikan produk yang sesuai dengan pilihan produk yang telah ditentukan oleh pelanggan sebelumnya, maka tentu dapat mengurangi kesulitan pemilihan produk tanpa mengurangi preferensi pelanggan. Jika sistem rekomendasi ini diimplementasikan ke dalam *website* untuk mendukung kegiatan bertransaksi secara *online*, maka tentu saja kemudahan pelanggan dalam memilih produk akan lebih banyak lagi. Akan tetapi, bertransaksi secara *online* juga memiliki hambatan tersendiri.

Umumnya pelanggan yang membeli secara *online* harus menggunakan imajinasi untuk membayangkan kesesuaian antara beberapa produk *fashion* yang telah dibeli. Berdasarkan hal ini, maka dikembangkan sebuah *website* yang dapat melayani pelanggan dalam membeli produk *fashion* sekaligus mengkombinasikan beberapa produk *fashion* yang telah dipilih secara virtual dengan menekankan pada fitur rekomendasi produk berdasarkan hubungan antara ketertarikan/penilaian seorang pembeli dengan pembeli lainnya terhadap suatu produk. Misalnya, pelanggan A membeli item x, y, z dan pelanggan B membeli item y. Saat B ingin membeli lagi, maka sistem akan merekomendasikan item x dan z.

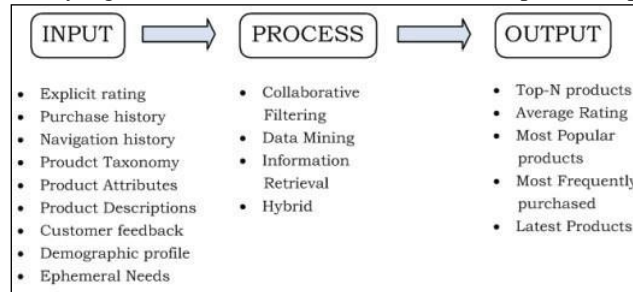
Pada tulisan ini akan dibahas pengembangan sistem rekomendasi pada *website* penjualan produk *fashion* yang disebut dengan *Virtual Fashion Room* (VFR) dengan beberapa fitur tambahan seperti mencoba produk *fashion* secara virtual dan bertransaksi menggunakan *website*. Metodologi pengembangan sistem yang akan digunakan mengacu kepada metodologi *Framework for the Application of System Thinking* (FAST) dengan strategi *Rapid Application Development* (RAD) yang menekankan pada *prototyping*.

2. REKOMENDASI PRODUK

Sistem rekomendasi sangat berguna bagi suatu perusahaan. Tanggapan *user* terhadap suatu produk dapat digunakan untuk merekomendasikan produk lain. Adapun beberapa hal yang umumnya perlu diperhatikan dalam merekomendasikan suatu produk antara lain [1]:

1. *Scale*, sebuah toko *online* dapat memiliki *database* yang berisi jutaan akun pelanggan dan jutaan produk item yang berbeda.
2. *Speed*, dibutuhkan algoritma yang dapat menghasilkan rekomendasi yang tepat.
3. *Diversity*, pelanggan yang baru biasanya memiliki informasi yang terbatas, sedangkan pelanggan yang sudah lama memiliki informasi yang lebih lengkap.
4. *Alteration*, setiap interaksi pelanggan memiliki informasi yang bernilai dan algoritma tersebut harus mampu merespon secepat mungkin terhadap informasi baru tersebut.

Secara umum, faktor-faktor yang terlibat dalam sistem rekomendasi dapat dilihat pada Gambar 1.



Gambar 1. Generic Input-Process-Output Model Recommender Systems [1]

2.1 Collaborative Filtering

Teknik *collaborative filtering* merupakan teknik yang banyak digunakan untuk menghasilkan rekomendasi produk. Ide utama dari teknik ini adalah mencari sejumlah pelanggan yang memiliki ketertarikan dan preferensi, kemudian menggunakan data ini untuk memberikan rekomendasi. Hasil dari sistem rekomendasi beragram tergantung dari data personalisasi itu digunakan. Umumnya, sistem rekomendasi menampilkan top_n produk yang dihasilkan dengan memberikan penilaian terhadap produk-produk yang didasarkan pada kriteria yang telah didefinisikan atau ditetapkan sebelumnya. *Collaborative filtering* kadang-kadang disebut sebagai *user-based filtering* yang memiliki aspek lain [1].

2.2 Item to Item

Pandangan lain dalam memberikan sistem rekomendasi pada produk-produk katalog yaitu berdasarkan item. Teknik berbasis item menganalisis hubungan antara pelanggan dengan item dan kemudian menggunakan hubungan tersebut untuk memberikan rekomendasi. Pertama, persamaan atau perbedaan antara item yang dicari, kemudian hasil evaluasi tersebut dihubungkan terhadap pelanggan atau sekelompok pelanggan. Daripada membandingkan pelanggan dengan pelanggan lain, *item to item collaborative filtering* membandingkan setiap item yang dibeli atau di-rating oleh pelanggan terhadap item yang mirip dan kemudian menggabungkan item-item yang mirip tersebut ke dalam daftar rekomendasi. Skalabilitas dari teknik berbasis item biasanya cukup baik, karena teknik ini hanya bergantung pada seberapa banyak item yang telah dibeli, di-rating, atau dikunjungi. Selain itu, performansi dari teknik ini tinggi karena perhitungan yang banyak terhadap item-item yang mirip dapat dilakukan secara *offline* dan skala perhitungan *online* dalam mencari item yang mirip untuk preferensi pelanggan tidak bergantung pada ukuran katalog atau total pelanggan [1].

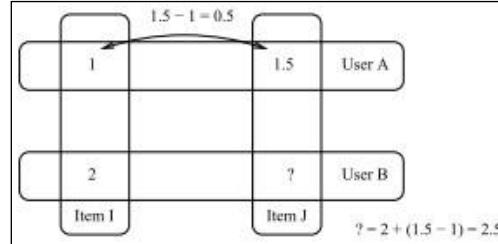
2.3 Online Rating Collaborative Filtering

Teknik *collaborative filtering* dengan cara *rating* secara *online* meminta data yang terdiri dari pasangan item dan *rating* dari seorang pelanggan. Respon dari permintaan data ini adalah sekelompok prediksi data yang terdiri dari pasangan item dan *rating* untuk item yang belum pernah di-rating oleh pelanggan tersebut. Teknik *collaborative filtering* yang baik harus mencakup kelima hal berikut ini [2]:

1. *Easy to implement and maintain*: semua *aggregated data* mudah diinterpretasi oleh *programmer* biasa dan algoritma mudah diimplementasikan dan diuji.
2. *Updatable on the fly*: penambahan data *rating* yang baru akan mengubah semua prediksi yang ada secara instan.
3. *Efficient at query time*: waktu *query* pasti cepat dan hemat biaya penyimpanan.
4. *Expect little from first visitor*: seorang pengguna dengan banyak *rating* seharusnya menerima rekomendasi yang valid.
5. *Accurate within reason*: skema ini sangat kompetitif dibandingkan dengan skema lain yang dapat melakukan prediksi secara akurat. Keakuratan juga tidak selalu memerlukan pengorbanan yang besar dalam kesederhanaan atau skalabilitas.

2.4 Slope One Scheme Algorithm

Algoritma *slope one scheme* memperhitungkan informasi dari pengguna lain yang melakukan *rating* terhadap item yang sama dan item lain yang dinilai oleh pengguna yang sama, namun algoritma ini juga bergantung pada data yang masuk, baik dalam *user array* maupun *item array* dan merupakan informasi yang penting untuk memprediksikan *rating*. Banyak pendekatan berasal dari data yang masuk. Secara khusus, hanya *rating* oleh pengguna yang telah memberi *rating* beberapa item yang sama dengan *predictee user* (pengguna yang ingin diprediksi nilai *rating* yang akan diberikannya) dan hanya *rating* item yang juga telah di-*rating* oleh *predictee user* termasuk ke dalam prediksi berdasarkan *rating* dari *slope one scheme*.



Gambar 2. Basis of Slope One Scheme [2]

Notasi-notasi berikut ini menjelaskan *slope one scheme*.

1. $evaluation_u$ adalah *rating* yang diberikan oleh *user* yang dinotasikan dengan *array* u , dimana u_i adalah *rating* yang diberikan *user* untuk item i .
2. $S(u)$ adalah himpunan bagian dari semua item, yang terdiri atas item-item yang sudah di-*rating* di $evaluation_u$.
3. χ adalah kumpulan dari semua $evaluation_u$ dalam *training set*.
4. $Card(S)$ adalah jumlah elemen yang terdapat dalam himpunan S .
5. \bar{u} adalah rata-rata dari *rating* yang terdapat dalam $evaluation_u$.
6. $S_i(\chi)$ adalah himpunan semua $evaluation_u \in \chi$ yang mengandung item i ($i \in S(u)$).
7. $P(u)$ adalah prediksi yang merepresentasikan sebuah vektor dimana setiap komponen adalah prediksi yang berhubungan dengan suatu item. Prediksi-prediksi bergantung secara implisit pada *training set* χ .

Secara formal, diberikan 2 *evaluation array* v_i dan w_i dengan $i = 1, \dots, n$. Kemudian dapat mulai mencari prediktor terbaik dengan formula $f(x) = x + b$ untuk memprediksi w dari v dengan meminimalkan $\sum_i (v_i + b - w_i)^2$. Berdasarkan turunan dari b dan andaikan turunan mendekati 0, maka akan diperoleh $b = \frac{\sum_i w_i - v_i}{n}$.

Dengan kata lain, konstanta b harus dipilih menjadi rata-rata (*average difference*) antara dua buah *array*. Hasil ini menghasilkan skema berikut ini.

Diberikan sebuah *training set* χ dan dua item j dan i dengan *rating* u_j dan u_i masing-masing dalam evaluasi beberapa pengguna (dinotasikan sebagai $u \in S_{j,i}(\chi)$), kemudian mempertimbangkan deviasi rata-rata dari item i yang berhubungan dengan item j sebagai berikut:

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

Perhatikan bahwa setiap *user evaluation* u tidak mengandung baik u_j maupun u_i tidak termasuk ke dalam penjumlahan. Matriks simetris didefinisikan dengan $dev_{j,i}$ dapat dihitung dengan cepat dan *up to date* setiap saat data baru dimasukkan. Mengingat bahwa $dev_{j,i} + u_i$ adalah prediksi untuk u_j yang diberikan u_i , prediktor yang masuk akal mungkin adalah rata-rata semua seperti prediktor seperti ini.

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

Dimana $R_j = \{i | i \in S(u), i \neq j, card(S_{j,i}(\chi)) > 0\}$ adalah kumpulan dari semua item yang relevan. Terdapat sebuah pendekatan yang dapat menyederhanakan perhitungan dari prediktor ini. Untuk kumpulan data (*dataset*) yang cukup padat dimana semua pasangan dari item yang telah di-*rating*, yaitu dimana $card(S_{j,i}(\chi)) > 0$ untuk hampir semua i, j paling banyak $R_j = S(u)$ untuk $j \notin S(u)$ dan $R_j = S(u) - \{j\}$ karena $j \in S(u)$.

Karena $\bar{u} = \frac{\sum_i u_i}{card(S(u))} \approx \frac{\sum_{i \in R_j} u_i}{card(R_j)}$ untuk kebanyakan j , maka formula prediktor untuk *slope one scheme* dapat disederhanakan menjadi

$$P^{S1}(u)_j = \bar{u} + \frac{1}{card(R_j)} \sum_{i \in R_j} dev_{j,i} \quad (3)$$

Perlu diketahui bahwa implementasi dari *slope one scheme* tidak tergantung dari bagaimana *user* melakukan *rating* item yang dibelinya, tetapi hanya pada rata-rata *rating* dari *user* tersebut dan yang penting pada item-item mana yang telah di-*rating* oleh *user* yang lain [2].

2.5 Mengukur Tingkat Akurasi

Mean Absolute Error (MAE) adalah salah satu persamaan yang sering digunakan untuk mengukur akurasi antara nilai yang diprediksi dengan nilai yang sebenarnya. Persamaan ini akan menghitung rata-rata selisih antara nilai prediksi dan nilai yang sebenarnya.

$$MAE = \frac{1}{n} \sum_{i=0}^n |p_i - f_i| \quad (4)$$

Dimana n adalah jumlah data yang dihitung, p_i adalah nilai prediksi pada data ke- i , dan f_i adalah nilai sebenarnya pada data ke- i .

3. KEPUTUSAN DALAM PEMBELIAN PRODUK FASHION

Pencarian informasi (*information search*) adalah proses dimana konsumen mensurvei lingkungannya untuk mencari data yang diperlukan untuk membuat sebuah keputusan. Pencarian informasi konsumen biasa terjadi sebelum pembelian (*pre-purchase search*) atau saat membeli (*ongoing search*). *Pre-purchase search* meningkatkan produk dan *market knowledge*, karena konsumen mencari informasi yang spesifik sesaat setelah dia menyadari sebuah masalah. Sedangkan konsumen yang menikmati *browsing* karena merasa senang, atau karena mereka sekedar ingin tahu yang terjadi di pasar terlibat dalam *ongoing search*, dimana meningkatkan kemungkinan terjadinya pembelian tak terencana (*impulse buying*). Sebuah pilihan harus dibuat dari alternatif yang tersedia, apalagi konsumen modern memiliki berbagai pilihan dari beragam merk (contoh: rokok) atau variasi berbeda dari merk yang sama (contoh: lipstik). Proses memilih produk ada yang sederhana dan cepat, namun ada juga yang rumit sampai melibatkan perhatian dan proses kognitif. Pilihan yang dibuat dapat dipengaruhi oleh hasil pencarian informasi, informasi yang muncul saat pembelian, atau kepercayaan terhadap merk yang mereka lihat dari iklan [3].

4. HASIL

Pengembangan sistem mengikuti tahapan metodologi FAST dengan strategi RAD sebagai berikut.

4.1 Scope Definition

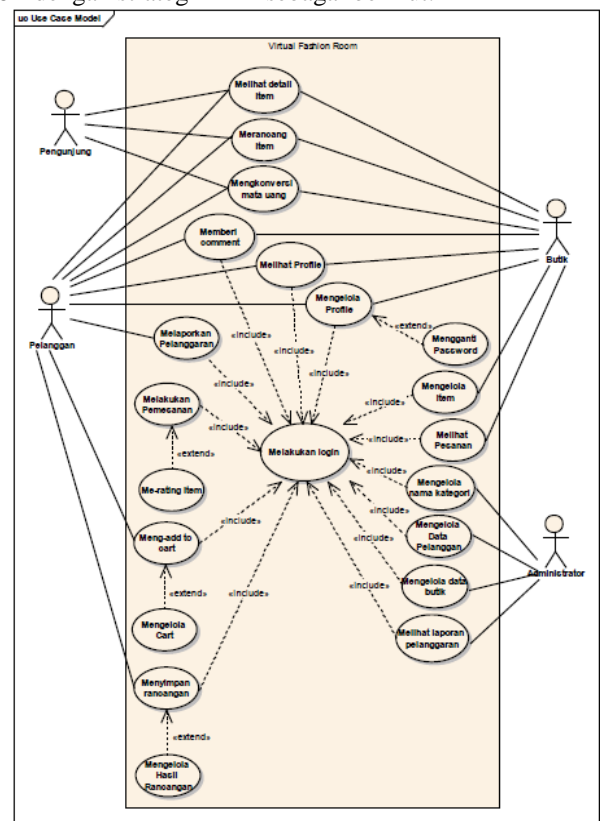
Ruang lingkup pengembangan sistem ini meliputi:

1. Media penjualan khusus untuk produk *fashion* wanita.
2. Perangkat lunak hanya meliputi proses pemesanan pakaian.
3. Perangkat lunak dilengkapi dengan sistem rekomendasi produk yang menggunakan algoritma *Slope One Scheme*.
4. Produk yang direkomendasikan digolongkan ke dalam tiga tipe, yaitu *popular item*, *item to item*, dan *related item*.

4.2 Problem Analysis

Rumusan masalah dalam penulisan ini mencakup:

1. *Customer* sulit mencari produk yang paling disukai di antara produk yang ditawarkan oleh berbagai butik secara *online* karena kurangnya rekomendasi produk.
2. *Customer* sulit dalam memadukan produk *fashion* yang sudah dibeli dari berbagai butik apabila berbelanja secara *online*.



Gambar 3. Use Case Diagram Virtual Fashion Room

4.3 Requirement Analysis

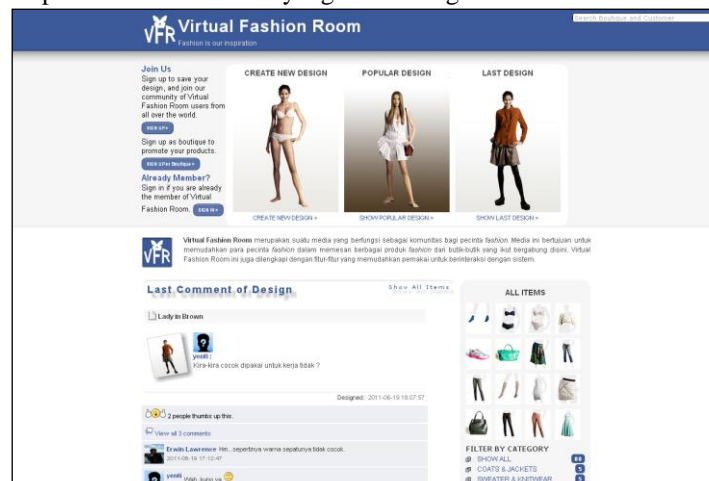
Mendefinisikan persyaratan yang harus dimiliki sistem. Terdapat empat *user* yang dapat menggunakan sistem ini, yakni administrator, pengunjung, pelanggan, dan butik. Masing-masing *user* memiliki hak akses yang berbeda-beda terhadap fitur yang terdapat di dalam sistem. Adapun fitur-fitur yang terdapat di dalam sistem dapat dilihat pada Gambar 3.

4.4 Decision Analysis

Pada tahap *decision analysis* dihasilkan arsitektur sistem baru. Sistem akan dibangun untuk dapat berjalan pada sisi *client* dan sisi *server*. Berdasarkan kedua sisi tersebut kemudian diputuskan *tools* dan sumber daya yang akan digunakan di *kelimallayer* sistem (*Presentation Layer*, *Presentation Logic Layer*, *Application Layer*, *Data Manipulation Layer*, dan *Data Layer*). Pada *Presentation Layer* digunakan Opera sebagai *web browser*. Untuk *Presentation Logic Layer* digunakan HTML5, CSS3, JQuery, dan JavaScript. PHP digunakan di *Application Logic Layer*, sedangkan untuk *Data Manipulation Layer* dan *Data Layer* menggunakan MySQL.

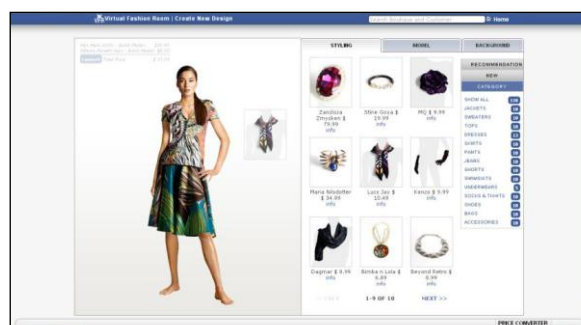
4.5 System Review

VFR dikembangkan atas tiga bagian utama, yaitu halaman pengunjung untuk pengguna yang belum melakukan *login*, halaman administrator, serta halaman *registered user* yang dapat dilihat oleh butik dan pelanggan. Gambar 4 merupakan tampilan awal dari sistem yang dikembangkan.



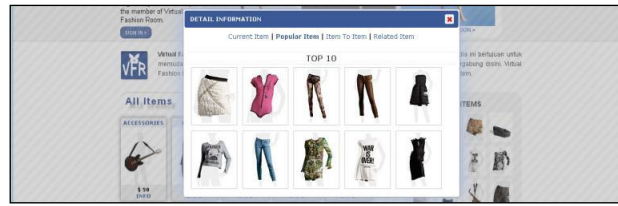
Gambar 4. Tampilan Awal VFR

Halaman *Create New Design* (Gambar 5) dapat digunakan untuk mendesain *fashion* berdasarkan beberapa produk yang berasal dari beberapa butik yang berbeda. Produk yang telah dipilih akan otomatis dihitung jumlah harganya oleh sistem. Untuk melakukan desain, pengguna melakukan *drag and dropp* pada objek dengan menggunakan item yang ditampilkan.



Gambar 5. Tampilan Fitur Create New Design pada VFR

Sistem juga menyediakan fitur rekomendasi item (Gambar 6). Rekomendasi item yang disediakan adalah berdasarkan *popular item*, *item to item*, dan *related item* dari item yang pertama kali dipilih. Untuk memudahkan pencarian terdapat menu filter item yang telah direkomendasikan berdasarkan kategori item.



Gambar 6. Tampilan Fitur Rekomendasi Popular Item

Pelanggan dapat memberikan *rating* terhadap barang yang telah dibelinya (Gambar 7). *Rating* ini yang digunakan sebagai dasar untuk menghasilkan sistem rekomendasi produk.

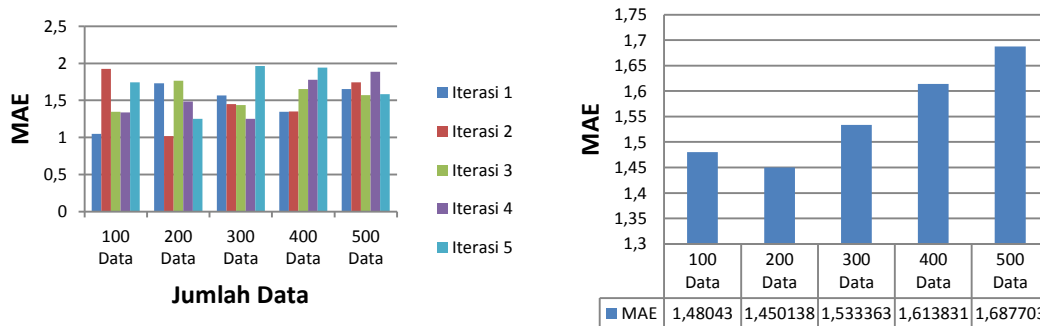


Gambar 7. Tampilan Fitur Rating Per Item Produk

5. PEMBAHASAN

Pengujian akurasi antara nilai yang diprediksi dengan nilai yang sebenarnya menggunakan persamaan *Mean Absolute Error* (MAE) dimana nilai MAE yang dihasilkan akan menunjukkan seberapa jauh nilai kesalahan antara nilai data uji dengan nilai data *training*. Pengujian dilakukan sebanyak 5 kali, dimana pada iterasi ke-*i* digunakan sebagai data uji, sementara sisanya digunakan sebagai data *training*. Nilai *rating* yang diberikan merupakan hasil *generate* secara random yang merepresentasikan *rating* yang diberikan pelanggan terhadap barang yang telah dibelinya.

Hasil pengujian pada Gambar 8 menunjukkan rata-rata nilai MAE terendah dicapai pada saat jumlah data 200 data, sedangkan nilai MAE tertinggi dicapai pada saat jumlah data sebanyak 500 data. Hal ini menunjukkan bahwa nilai MAE cenderung menaik ketika jumlah data yang diproses semakin banyak.



(a) (b)

Gambar 8. Grafik MAE Dengan Perbedaan Jumlah Data (a) dan Rata-Rata MAE Terhadap Jumlah Data (b)

6. KESIMPULAN DAN SARAN

Dengan adanya *websiteVFR* memungkinkan pelanggan dapat melakukan akses informasi dan kegiatan transaksi, mencoba *fashion* secara *virtual*, mengetahui *fashion* yang sedang tren, serta memudahkan dalam pencarian *fashion* dari sekian banyak produk yang ditawarkan karena tersedianya sistem rekomendasi produk. Sistem rekomendasi dalam VFR adalah berdasarkan hubungan antara ketertarikan/penilaian seorang pembeli dengan ketertarikan/penilaian pembeli lainnya terhadap suatu produk. Sistem rekomendasi produk ini sangat bergantung terhadap banyaknya penilaian (*rating* item) dari pembeli. Untuk penelitian selanjutnya, diharapkan dapat dikembangkan fitur rekomendasi produk bukan hanya berdasarkan item yang dipilih, akan tetapi juga berdasarkan desain dari para pelanggan.

7. DAFTAR RUJUKAN

- [1] Helsinki University of Technology, 2007. *Slope One Predictor on Consumer Data*. Finlandia: Independent Research Project in Applied Mathematic.
- [2] Lemire, D. and A. Maclachlan, 2005. *Slope One Predictors for Online RatingBased Collaborative Filtering*.
- [3] Solomon, M. R. and N. J. Rabolt, 2004. *Consumer Behaviour in Fashion*. London: Pearson Education, Inc.