



Hak cipta dan penggunaan kembali:

Lisensi ini mengizinkan setiap orang untuk menggubah, memperbaiki, dan membuat ciptaan turunan bukan untuk kepentingan komersial, selama anda mencantumkan nama penulis dan melisensikan ciptaan turunan dengan syarat yang serupa dengan ciptaan asli.

Copyright and reuse:

This license lets you remix, tweak, and build upon work non-commercially, as long as you credit the origin creator and license it on your new creations under the identical terms.

BAB II

LANDASAN TEORI

2.1 Sistem Rekomendasi

Sistem rekomendasi merupakan sebuah perangkat lunak yang bertujuan untuk membantu pengguna dengan cara memberikan rekomendasi kepada pengguna ketika pengguna dihadapkan dengan jumlah informasi yang besar. Rekomendasi yang diberikan diharapkan dapat membantu pengguna dalam proses pengambilan keputusan. (Ricci, Rokach, Shapira, & Kantor, 2011)

Sistem rekomendasi memiliki 3 metode yaitu, *Content Based*, *Collaborative Filtering* dan *Hybrid Recommendation* (Sartika, 2014)

a. *Content Based*.

Content Based adalah metode rekomendasi dengan menganalisa deskripsi item untuk mengidentifikasi item yang memiliki ketertarikan khusus dari seorang *user*. Deskripsi didapat dari timbal balik *user* yang disediakan setiap item atau *user* dapat mengisi sendiri melalui profile masing-masing (Pazzani & Billsus, 2007)

b. *Collaborative Filtering*

Collaborative Filtering adalah metode rekomendasi dengan mempertimbangkan opini seseorang tentang item tersebut (Ekstrand, Riedl, & Konstan, 2011). Opini bisa merupakan *rating* item yang diberikan oleh *user*.

c. *Hybrid Recommendation*

Metode ini adalah metode rekomendasi yang menggabungkan beberapa metode yang ada menjadi satu, misalnya menggabungkan metode *Content Based* dan

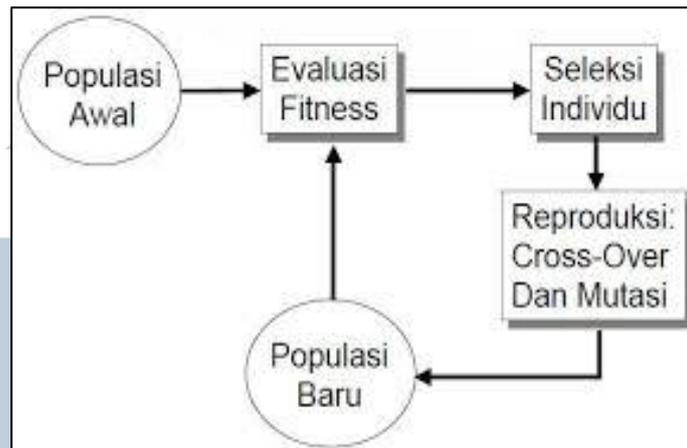
Collaborative Filtering secara bersamaan pada sebuah sistem rekomendasi untuk mencapai hasil rekomendasi yang diinginkan (Ekstrand, Riedl, & Konstan, 2011).

Dalam kasus ini, penulis memilih metode *content based* karena sistem rekomendasi yang akan dikembangkan ini secara *default* menggunakan harga sebagai ketertarikan *user*.

2.2 Algoritma Genetika

Algoritma Genetika adalah metode pencarian yang didasarkan pada prinsip seleksi alam dan genetika, dan biasanya digunakan untuk memproduksi solusi-solusi yang memiliki kualitas tinggi bagi optimisasi dan problem pada pencarian, dan menggunakan operator alam seperti mutasi, kawin silang (*crossover*) dan seleksi (Mitchell, 1996). Algoritma ini menggambarkan variabel keputusan dari suatu masalah sebagai suatu *string* yang merupakan urutan alfabet yang panjangnya terbatas (Sastry, 2005). Algoritma genetika bekerja dengan suatu populasi *string* dan melakukan proses pencarian nilai optimal secara paralel dengan menggunakan operator genetika. Algoritma genetika akan melakukan rekombinasi antar individu. Algoritma genetika memiliki elemen dasar berupa *string* yang tersusun dari rangkaian *substring* (gen), yang masing-masing merupakan kode dari parameter dalam ruang solusi dimana suatu *string* (kromosom) menyatakan kandidat solusi.

Kumpulan *string* dalam populasi berkembang dari generasi ke generasi melalui operator genetika. Pada setiap iterasi, individu-individu kromosom dalam populasi itu akan di evolusi dan di seleksi untuk menentukan populasi pada generasi berikutnya. Populasi ini akan terus berulang sampai menemukan suatu parameter dengan nilai yang paling optimal sesuai dengan yang diinginkan.



Gambar 2.1. Siklus Algoritma Genetika (Goldberg, Sastry, & Kendall, 2005)

2.2.1 Metode Seleksi Natural Selection

Natural Selection adalah proses seleksi yang melibatkan sekumpulan individu pada sebuah populasi, dimana individu yang memiliki nilai fitness tinggi akan memiliki keuntungan untuk bertahan, dan individu yang memiliki nilai fitness tertinggi dari sebuah populasi tersebut akan dipilih (Mijwel, 2016). Dalam kasus ini, yang akan dipilih adalah dua individu yang memiliki nilai fitness tertinggi, dan akan dilakukan proses Reproduksi

2.2.2 Metode Single Point Crossover

Crossover adalah operator algoritma genetika yang membutuhkan parameter dua kromosom. Dua buah kromosom tersebut disebut kromosom induk.

Operator ini akan menghasilkan dua buah kromosom baru. Ada beberapa jenis *crossover* yang sering digunakan dalam algoritma genetika antara lain metode *single point crossover*. Metode ini merupakan bagian dari Algoritma Genetika dengan menentukan *crossover point* (gen tertentu). Kromosom baru pertama berisi gen pertama sampai gen *crossover point* dari kromosom induk pertama ditambah

dengan gen dari *crossover point* sampai gen *crossover point* dari induk kedua. Kromosom baru kedua berisi gen pertama sampai gen *crossover point* dari induk kedua ditambahkan dengan gen dari *crossover point* sampai gen dari kromosom induk pertama.

Random position $k = 3$				Chromosome length $l = 8$				
Parent 1	0.32	0.22	0.34	0.89	0.23	0.76	0.78	0.45
Parent 2	0.12	0.65	0.38	0.47	0.31	0.56	0.88	0.95
swapping all the genes between 4 and 8				↓				
Offspring 1	0.32	0.22	0.34	0.47	0.31	0.56	0.88	0.95
Offspring 2	0.12	0.65	0.38	0.89	0.23	0.76	0.78	0.45

Gambar 2.2 Proses *crossover* dengan *single point crossover* (Mendes, 2013)

Dari ilustrasi di atas maka contoh penerapan metode *single point crossover* adalah sebagai berikut:

Orang Tua 1: 0 0 0 | 1 1 1 1 1 1 1 1 1 1

Orang Tua 2: 1 1 1 | 0 0 0 0 0 0 0 0 0 0

Setelah proses *crossover* turunan yang dapat dihasilkan adalah dari kedua parent diatas adalah:

Anak 1: 0 0 0 | 0 0 0 0 0 0 0 0 0 0

Anak 2: 1 1 1 | 1 1 1 1 1 1 1 1 1 1

2.2.3 Metode Uniform Mutation

Metode mutasi yang mengubah satu buah nilai gen yang dipilih dengan nilai acak antara batas atas dan batas bawah yang ditentukan (Academia Sinica, 2004). *Mutation rate* akan ditentukan diawal proses mutasi, dan berdasarkan bilangan yang diacak akan dapat menentukan gen-gen yang akan dimutasi.

2.3 Komputer

Sistem Komputer adalah elemen-elemen yang terkait untuk menjalankan suatu aktifitas dengan menggunakan computer. Elemen dari sistem komputer terdiri dari Manusia (*Brainware*), perangkat lunak (*software*), dan perangkat keras (*hardware*) (Palit & Mogot, 2012)

a. *Processor*

Processor atau yang biasa dibilang dengan CPU(*Central Processing Unit*) adalah otak dari sebuah sistem komputer, *processor* yang akan memerintah kerja kepada *hardware* dan *software*. (Higher National Computing, 2018)

b. *Motherboard*

Motherboard adalah *Printed Circuit Board* utama pada sebuah sistem komputer, dan merupakan tempat terjadinya komunikasi antara komponen-komponen yang terhubung. (TechTarget, 2018)

c. Kartu Grafis

Kartu Grafis merupakan papan sirkuin khusus yang biasanya dipasang di *PCI Express* pada computer moderen, kartu grafis juga memiliki memori, yang disebut VideoRAM, kartu grafis biasanya untuk mengolah gambar 2D, 3D atau bahkan untuk melakukan *General Purpose Graphic Processing*(*GPGPU Computing*). (TechTarget, 2018)

d. Memori

Memori atau yang biasa disebut RAM adalah tempat penyimpanan sementara yang akan digunakan *processor* untuk menyimpan perintah, dan data-data terkait ketika perintah sedang berjalan. (Higher National Computing, 2018)

e. *Power Supply*

Power Supply adalah komponen yang mengalirkan tenaga ke paling tidak satu buah arus, dan menyediakan listrik untuk komponen komputer. (TechOpedia, 2018)

f. *Hard Disk*

Hard Disk adalah media penyimpanan berupa piringan magnet yang menyimpan data pada sebuah area piringan *hard disk*. (Higher National Computing, 2018)

g. *Monitor*

Monitor adalah perangkat *Electronic Visual Computer Display* yang menyediakan sebuah layer, dijamin moderen monitor menggunakan LCD pada perangkat seperti *Laptop*, dan *Handphone* karena LCD lebih enteng dan tidak mengkonsumsi banyak energi. (TechOpedia, 2018)

h. *Casing*

Casing adalah tampak luar sebuah komputer, digunakan untuk melindungi komponen yang berada didalam, biasanya *Motherboard*, *Hard Disk*, *Power Supply* dan komponen yang lainnya dipasang didalam *Casing* untuk menjadi komputer yang utuh. (Bleeping Computer, 2018)

i. *CPU Cooler*

CPU Cooler adalah sebuah komponen yang bertugas untuk mendinginkan komponen *Processor* yang telah di pasang di *Motherboard* dan untuk mendinginkan area sekitar *Processor* seperti *Voltage Regulator Module*(VRM) (Burke, 2018)

2.4 Skala Likert

Skala Likert adalah skala yang digunakan untuk mengukur persepsi, sikap atau pendapat seseorang atau kelompok mengenai suatu peristiwa. Skala ini merupakan suatu skala psikometrik yang biasa diaplikasikan dalam angket dan paling sering digunakan untuk riset yang berupa survei. Dalam membuat skala likert ada beberapa prosedur yang harus dilakukan peneliti, antara lain:

- Peneliti mengumpulkan *item-item* yang cukup banyak, memiliki relevansi dengan masalah yang sedang diteliti, dan terdiri dari *item* yang cukup jelas disukai dan tidak disukai.
- Kemudian *item-item* itu dicoba kepada sekelompok responden yang cukup representatif dari populasi yang ingin diteliti.
- Responden di atas diminta untuk mengecek tiap *item*, apakah ia menyenangi (+) atau tidak menyukainya (-). Respons tersebut dikumpulkan dan jawaban yang memberikan indikasi menyenangi diberi skor tertinggi. Tidak ada masalah untuk memberikan angka 5 untuk yang tertinggi dan skor 1 untuk yang terendah atau sebaliknya. Yang penting adalah konsistensi dari arah sikap yang diperlihatkan. Demikian juga apakah jawaban “setuju” atau “tidak setuju” disebut yang disenangi, tergantung dari isi pertanyaan dan isi dari *item-item* yang disusun. Ketika menanggapi pernyataan dalam skala likert, responden menentukan tingkat persetujuan mereka terhadap suatu pernyataan dengan memilih salah satu dari pilihan yang tersedia (Nazir M, 2005).
- Total skor dari masing-masing individu adalah penjumlahan dari skor masing-masing *item* dari individu tersebut.

Respon dianalisis untuk mengetahui *item-item* mana yang sangat nyata batasan antara skor tinggi dan skor rendah dalam skala total. Misalnya, responden pada upper 25% dan lower 25% dianalisis untuk melihat sampai berapa jauh tiap *item* dalam kelompok ini berbeda. *Item-item* yang tidak menunjukkan beda yang nyata, apakah masuk dalam skoringgi atau rendah.

Tabel 2.1 Bobot Nilai

Keterangan	Bobot Nilai
Sangat Setuju	6
Setuju	5
Netral atau Cukup	4
Ragu	3
Tidak Setuju	2
Sangat Tidak Setuju	1

Tabel 2.2 Persentase Nilai

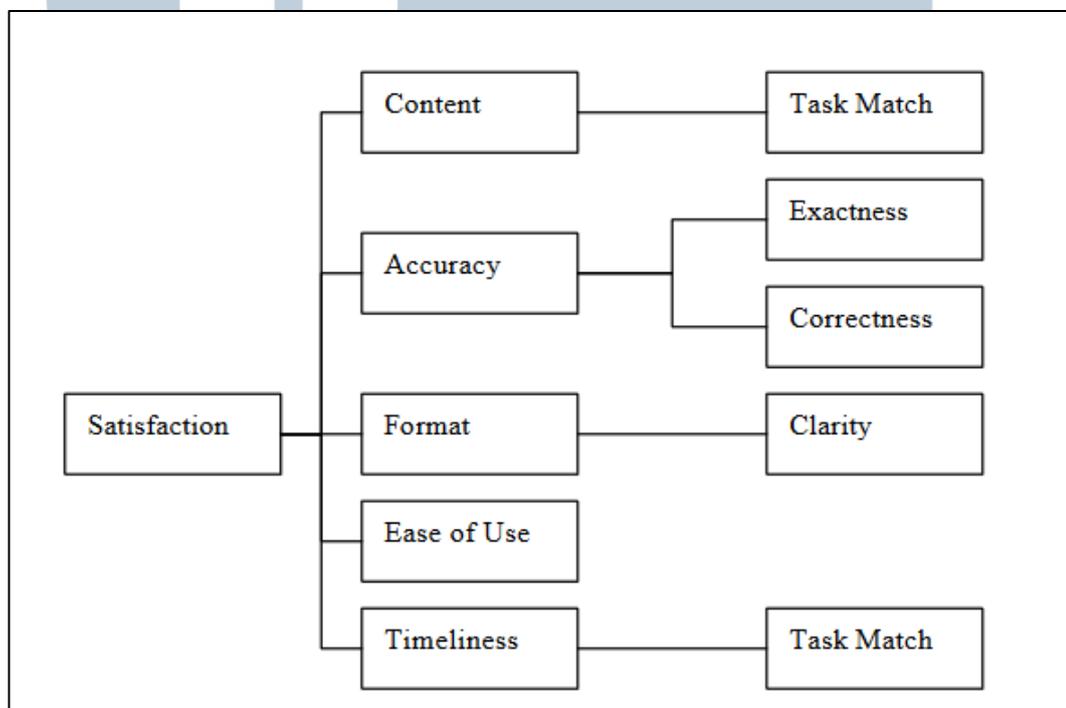
Jawaban	Keterangan
0% - 16%	Sangat (Tidak Setuju , Buruk , Kurang)
17% - 33%	Tidak Setuju atau Kurang Baik
34% - 50%	Ragu
51% - 66%	Netral atau Cukup
67% - 83%	Setuju, Baik atau Suka
84% - 100%	Sangat (Setuju , Baik , Suka)

Persentase Skor = (((Sangat Setuju * 6) + (Setuju * 5) + (Netral * 4) + (Ragu * 3) + (Tidak Setuju * 2) + (Sangat Tidak Setuju * 1)) / (6 * Jumlah Responden)) * 100%

Dengan demikian kita bisa menghitung presentase skor dengan rumus Sugiyono (2012), seperti contoh yang ada di atas.

2.7 End User Computing Satisfaction

End User Computing Satisfaction adalah metode untuk mengukur tingkat kepuasan dari pengguna suatu aplikasi dengan membandingkan antara harapan dan kenyataan dari sebuah sistem informasi (Doll et al, 1995). Model evaluasi EUCS ini dikembangkan oleh Doll & Torkzadeh. Evaluasi dengan menggunakan model ini lebih menekankan kepuasan pengguna akhir terhadap aspek teknologi, dengan menilai isi, keakuratan, format, waktu dan kemudahan penggunaan dari sistem.



Gambar 2.5 Model Evaluasi *End User Computing Satisfaction* (Doll et al, 1995)

Berikut adalah penjelasan dari tiap dimensi yang diukur dengan metode *End User Computing Satisfaction* menurut Doll & Torkzadeh (1995).

1. Dimensi *Content*, mengukur kepuasan pengguna ditinjau dari sisi isi dari suatu sistem. Isi dari sistem biasanya berupa fungsi dan modul yang dapat digunakan oleh pengguna sistem dan juga informasi yang dihasilkan oleh sistem. Dimensi

content juga mengukur apakah sistem menghasilkan informasi yang sesuai dengan kebutuhan pengguna. Semakin lengkap modul dan informatif sistem maka tingkat kepuasan dari pengguna akan semakin tinggi.

2. Dimensi *Accuracy*, mengukur kepuasan pengguna dari sisi keakuratan data ketika sistem menerima *input* kemudian mengolahnya menjadi informasi. Keakuratan sistem diukur dengan melihat seberapa sering sistem menghasilkan *output* yang salah ketika mengolah *input* dari pengguna, selain itu dapat dilihat pula seberapa sering terjadi *error* atau kesalahan dalam proses pengolahan data.
3. Dimensi *Format*, mengukur kepuasan pengguna dari sisi tampilan dan estetika dari antarmuka sistem, apakah antarmuka dari sistem itu menarik dan apakah tampilan dari sistem memudahkan pengguna ketika menggunakan sistem sehingga secara tidak langsung dapat berpengaruh terhadap tingkat efektifitas dari pengguna.
4. Dimensi *Ease of Use*, mengukur kepuasan pengguna dari sisi kemudahan pengguna atau *user friendly* dalam menggunakan sistem seperti proses memasukkan data, mengolah data dan mencari informasi yang dibutuhkan.
5. Dimensi *Timeliness*, mengukur kepuasan pengguna dari sisi ketepatan waktu sistem dalam menyajikan atau menyediakan data dan informasi yang dibutuhkan oleh pengguna.

U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A