

# Aplikasi Hybrid Filtering Dan Naïve Bayes Untuk Sistem Rekomendasi Pembelian Laptop

Hybrid Filtering and Naïve Bayes Application for Laptop Purchase Recommendation Systems

**Eri Eli Lavindi<sup>1</sup>, Wijanarto<sup>2</sup>, Asih Rohmani<sup>3</sup>**

<sup>1,2,3</sup>Fakultas Ilmu Komputer, Universitas Dian Nuswantoro,

Jl Imam Bonjol 207 telp/fax 0274 3517261/3569684

e-mail : <sup>1</sup>erielilavindi@gmail.com, <sup>2</sup>wijanarto@dsn.dinus.ac.id, <sup>3</sup>aseharsoyo@dsn.dinus.ac.id

## **Abstrak**

*Perkembangan E-commerce telah menjadi terobosan dalam strategi bisnis modern yang dapat menghasilkan kenaikan pendapatan secara virtual. Belanja merupakan kegiatan sosial dan biasanya pembeli mendapatkan rekomendasi dari orang lain untuk membeli suatu produk. Ketika mendapatkan rekomendasi, tingkat kepercayaan untuk membeli produk tersebut akan meningkat. Seiring perkembangan zaman, laptop telah menjadi kebutuhan pokok bagi masyarakat. Toko – toko yang menawarkan penjualan laptop saat ini melakukan rekomendasi produk laptop dengan mengacu pada pembelian konsumen lain. Faktor berikutnya adalah informasi mengenai harga, spesifikasi laptop, kemudahan, serta rekomendasi dalam proses pembelian yang akurat bagi calon pembeli. Penelitian membuat Sistem Rekomendasi Penjualan Laptop menggunakan metode Hybrid Filtering dan klasifikasi Naïve Bayes, yang bertujuan untuk membantu masyarakat maupun pegawai toko untuk menentukan apakah suatu laptop di rekomendasikan untuk dibeli atau tidak. Penelitian ini menggunakan 95 data spesifikasi, beserta rating dari laptop yang di rekomendasikan dan tidak di rekomendasikan. Dengan menggunakan 20 data testing, berdasarkan pengujian menggunakan confusion matrix, Algoritma Naïve Bayes mampu mengklasifikasikan suatu laptop di rekomendasikan atau tidak dengan tingkat akurasi 80%.*

**Kata kunci :** Sistem Rekomendasi, Hybrid Filtering, Naïve Bayes

## **Abstract**

*The development of E-commerce has become a breakthrough in modern business strategies that can result in increased virtual income. Shopping is the best social activity and expenditure from other people to buy products. When getting a recommendation, the level of trust to buy the product will increase. Along with the development of the times, laptops became the main need for the community. Stores that offer laptop sales are currently recommending laptop products with approval for other consumer purchases. The next factor is information about prices, specifications of laptops, convenience, and accurate assessment of the purchasing process for prospective buyers. The study made the Laptop Sales Recommendation System using the Hybrid Filtering method and the Naïve Bayes classification, which offered to help the public or store employees to determine whether the laptop was recommended for purchase or not. This study uses 95 specification data, along with an assessment of laptops that are recommended and not recommended. By using 20 test data, using testing using a confusion matrix, the Naïve Bayes algorithm is able to classify laptops as recommended or not with an 80% accuracy rate.*

**Keywords :** Recommendation System, Hybrid Filtering, Naïve Bayes

## 1. PENDAHULUAN

Menurut Biro Sensus A.S. pada Survei Perdagangan, penjualan ritel di Internet untuk tahun 2000 adalah \$ 25,8 miliar, atau 49% lebih tinggi dari penjualan tahun 1999 sebesar \$ 17,3 miliar [1]. Seperti halnya yang di alami oleh Amazon, peningkatan pendapatan dari \$ 600 juta pada tahun 1998 naik menjadi \$10.7 miliar di tahun 2006. Terhitung peningkatan penghasilan mencapai 65% di tahun 2005 hingga 2007 [2]. Perkembangan *e – commerce* telah menjadi terobosan dalam strategi bisnis modern yang dapat menghasilkan kenaikan pendapatan secara virtual yang tidak bisa didapatkan dari layanan toko cabang [3]. Belanja merupakan kebutuhan sosial dan biasanya pembeli mendapatkan rekomendasi dari orang lain untuk membeli sebuah produk. Ketika sudah mendapatkan rekomendasi dari orang lain, tingkat kepercayaan untuk membeli produk tersebut akan meningkat [4]. Seiring perkembangan zaman , laptop telah menjadi kebutuhan pokok bagi masyarakat. Toko – toko yang menawarkan penjualan laptop saat ini melakukan rekomendasi produk laptop dengan mengacu pada pembelian konsumen lain. Selain itu , informasi mengenai harga, spesifikasi laptop, kemudahan, serta rekomendasi dalam proses pembelian yang akurat sangat diperlukan bagi calon pembeli [5].

Sistem Rekomendasi merupakan suatu sistem yang berjalan pada *e-commerce* dimana digunakan untuk memberikan saran kepada calon pembeli mengenai informasi yang membantu pemilihan keputusan dalam pembelian produk [3]. Rekomendasi tradisional fokus pada menemukan produk yang diyakini merupakan yang terbaik. Sistem Rekomendasi modern yang ada pada *e-commerce* menggunakan metode–metode seperti *content-based filtering*, *collaborative filtering* dan *hybrid filtering* [6]. *Content-based filtering* merupakan metode pemilihan rekomendasi berdasarkan kemiripan dengan produk yang disukai user pada waktu sebelumnya. *Collaborative filtering* merupakan metode pemilihan rekomendasi yang didasarkan pada kesukaan banyak user terhadap suatu produk. Sedangkan *Hybrid filtering* merupakan kombinasi dari *content-based* dan *collaborative filtering* yang memanfaatkan keuntungan dari kedua metode [7].

Dalam penelitian, kerangka utamanya akan menggunakan metode *hybrid recommendation method* untuk menyelesaikan masalah yang dihadapi calon pembeli laptop. Metode ini dipilih karena menggabungkan kekuatan metode *collaborative* yaitu rekomendasi multi user pada satu item dengan metode *content based* dimana rekomendasi atas dasar preferensi user pada suatu item pada masa lalu. Sistem akan menggunakan algoritma Naïve Bayes untuk perhitungan probabilistik rekomendasi yang dihasilkan, karena dapat melakukan pemutusan data nominal dengan baik dalam proses klasifikasi serta dapat menghasilkan akurasi lebih dari 80% dalam system rekomendasi [13]. Algoritma Naïve Bayes merupakan algoritma dalam bidang data mining yang digunakan untuk melakukan klasifikasi dimana algoritma ini bersifat statistik. Algoritma ini menjanjikan akurasi tinggi dan kecepatan operasi untuk diimplementasikan pada data [8]. Pada penelitian sebelumnya, algoritma Naïve Bayes dapat di implementasikan pada sistem rekomendasi bacaan tugas akhir menggunakan metode *collaborative filtering* [9]. Maka dari itu penulis akan mengimplementasikan metode *hybrid filtering* dan klasifikasi Naïve Bayes untuk merekomendasikan laptop pada proses pembelian.

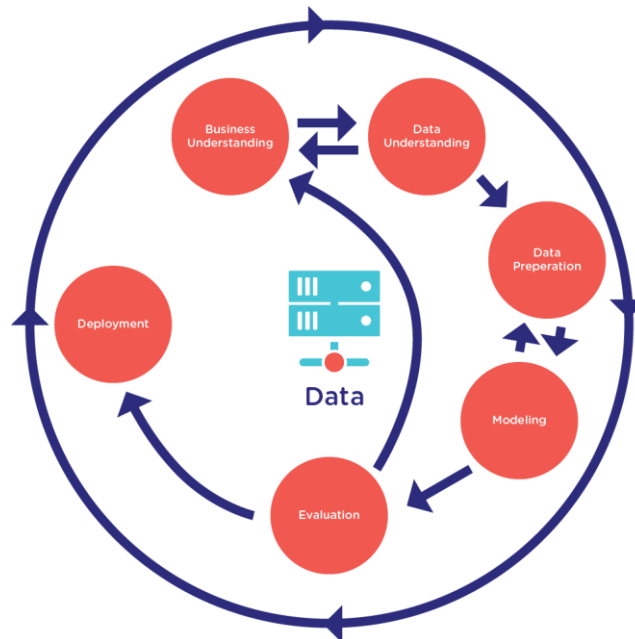
## 2. METODE PENELITIAN

Terdapat dua bagian dalam penelitian ini yaitu metode analisa data serta metode perancangan system. Dalam proses analisa data, metode yang dipakai adalah *Cross Industry Standard Process for Data Mining* (CRISP-DM). CRISP-DM dirilis oleh Daimler Chrysler, SPSS, dan NCR pada tahun 1996. Tahapan CRISP-DM pada penelitian ini adalah sebagai berikut:

### 1. *Bussines Understanding*

Observasi secara langsung dilakukan pada Situs Amazon. Dimana pada tahapan ini peneliti memahami proses jual beli serta proses input rating yang terdapat di Amazon. Data diambil dengan teknik *crawling*, yaitu laptop yang dijual beserta rating yang ada pada laptop tersebut pada

situs. Dan dari data yang diperoleh, dipilih secara acak 449 user untuk memberikan rating (nilai dari 1 sampai dengan 5) terhadap laptop yang dibelinya untuk direkomendasikan atau tidak.



Gambar 1. Alur *Cross Industry Standard Process for Data Mining*

## 2. *Data Understanding*

Pemahaman mengenai data oleh peneliti dilakukan untuk memahami keseluruhan bagian-bagian dari data, bagaimana data akan dimanfaatkan serta format data yang akan dipakai. Dalam tahapan ini, peneliti melakukan proses pemahaman terhadap list data item laptop meliputi spesifikasi dan rating yang telah didapat.

## 3. *Data Preparation*

Setelah peneliti melakukan pemahaman dan persiapan data, maka selanjutnya dilakukan filter terhadap entitas data yang ada. Entitas pada tabel item laptop yang dipakai yaitu **RAM**, **VGA**, **Hardisk**, **Procesor**, dan **Rating**. Record-record yang ada pada entitas tersebut akan dijadikan satu menjadi sebuah tabel set data. Setelah dilakukan preprosesing didapatkan 190 data yang valid untuk dilakukan proses modeling.

## 4. *Modelling*

Pemodelan dilakukan menggunakan *Algoritma Naïve Bayes* untuk mendapatkan hasil berupa rule yang akan digunakan sebagai acuan bahwa dari sampel data yang ada memiliki pengetahuan yang dapat dipakai sebagai bahan rekomendasi. Selanjutnya dilakukan implementasi algoritma *Naive Bayes* dengan metode *hybrid* pada sistem rekomendasi pembelian laptop sebagai proses modeling secara langsung untuk dapat menangani kasus pemberian rekomendasi. Metode *hybrid* yang di gunakan adalah **Content Based Filtering** dan **Collaborative Filtering**. *Content Based Filtering* di gunakan untuk memberikan rekomendasi kepada masing-masing user yang teregistrasi berdasarkan produk yang di sukai user itu sendiri. Sedangkan *Collaborative Filtering* digunakan untuk memberikan rekomendasi kepada user yang tidak teregistrasi dengan menggunakan data produk yang di sukai oleh seluruh user yang sudah teregistrasi.

## 5. *Evaluation*

Evaluasi dilakukan terhadap model yang dihasilkan yang berupa hasil rekomendasi kepada user. Teknik evaluasi dilakukan dengan perbandingan manual terkait algoritma *Naïve Bayes*. *Record* awal seorang user beserta personalisasinya dibandingkan dengan hasil rekomendasi menggunakan sistem rekomendasi. Selain itu dilakukan pengujian dengan metode **Confusion Matrix** untuk mengetahui tingkat akurasi dan tingkat **error** dari hasil klasifikasi menggunakan *Naive Bayes*. Metode evaluasi ini akan memberikan informasi aktual dan prediksi klasifikasi yang dilakukan sistem klasifikasi yang umumnya dievaluasi menggunakan data dalam matrik. *True*

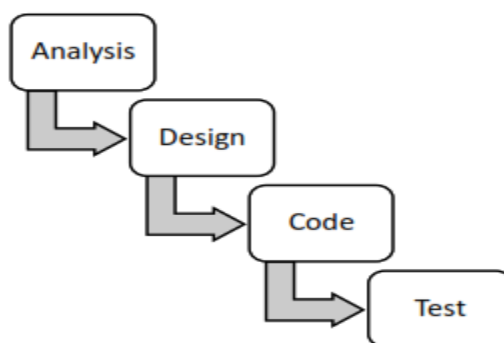
*Negative*, merupakan banyaknya prediksi yang benar bahwa *record* uji adalah negatif (dapat disebut *True Negative*), *False Positive* merupakan banyaknya prediksi yang salah bahwa *record* uji merupakan positif (dapat disebut *False Positive*), *False Negative* merupakan banyaknya yang salah prediksi bahwa *record* uji adalah negatif (dapat disebut *False Negative*), dan *True Positive* merupakan banyaknya prediksi yang benar bahwa *record* uji merupakan positif (dapat disebut *True Positive*). Sehingga dapat dihitung nilai akurasi dan *error rate* berdasarkan formulasi diatas.

$$Akurasi = \frac{TP + TN}{TP + TN + FP + FN} \quad \text{dan} \quad Error\ Rate = \frac{FP + FN}{TP + TN + FP + FN}$$

6. *Deployment*

Setelah tahap evaluasi dilakukan dan mendapatkan hasil sesuai yang diharapkan maka *prototype* sistem rekomendasi pembelian laptop dapat dikembangkan lebih lanjut untuk implementasi metode *hybrid* menggunakan algoritma Naive Bayes serta dilakukan *update* secara berkala jika dibutuhkan.

Sementara dalam pengembangan sistem, metode yang dipakai adalah *waterfall*, seperti pada gambar 2 di bawah ini.



Gambar 2. Metode *Waterfall*

Pendekatan *waterfall* dimulai dari *Analysis*, yaitu menentukan kebutuhan sistem yang akan dibangun (tool, input data rating, perhitungan probabilitas, klasifikasi data dan membuat rekomendasi). *Design*, dalam tahap ini akan menggunakan UML dalam pemodelannya dari *use case* hingga *class diagram*. *Code*, model yang dipilih akan dikodekan dengan editor dan pemroses bahasa PHP. Sementara *test system* yang dipilih adalah *Whitebox*. Untuk pengujian analisa data, menggunakan *confusion matrix* dan *error rate*.

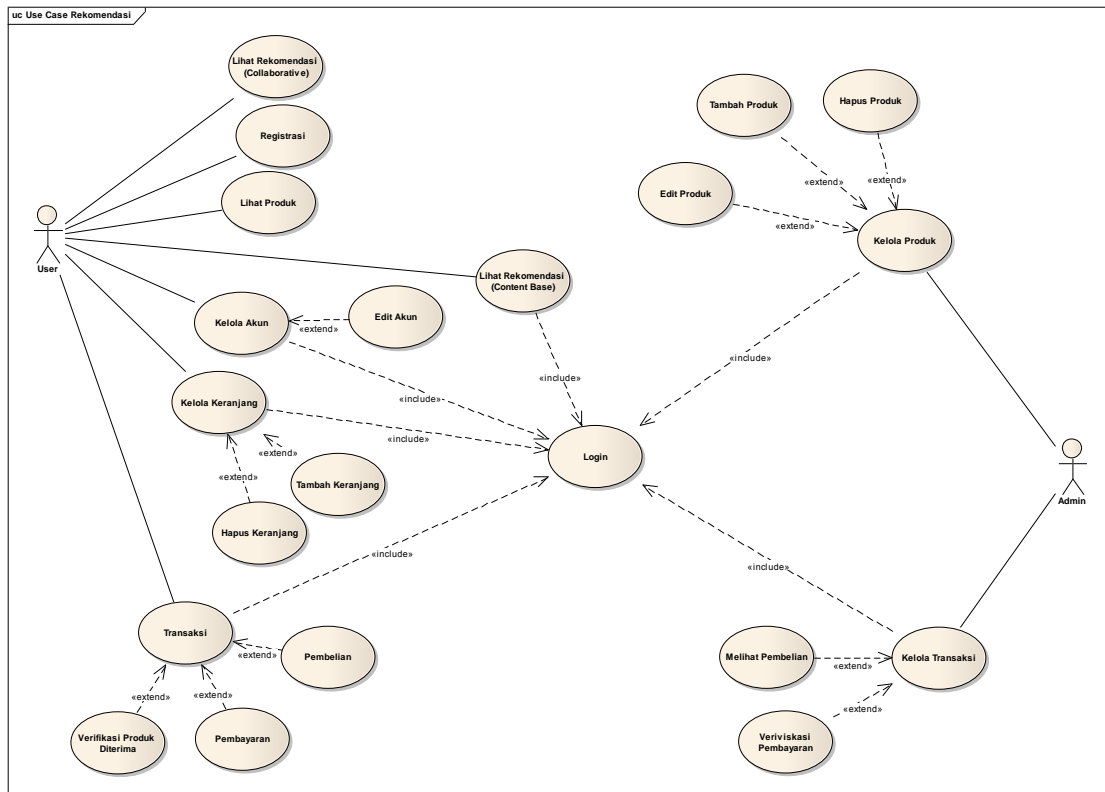
### 3. HASIL DAN PEMBAHASAN

Pada bagian ini akan dipaparkan hasil rancangan serta implementasi sistem rekomendasi yang dibuat dalam aplikasi berbasis web. Rancangan inti berupa *use case*, *sequence* dan *class diagram*.

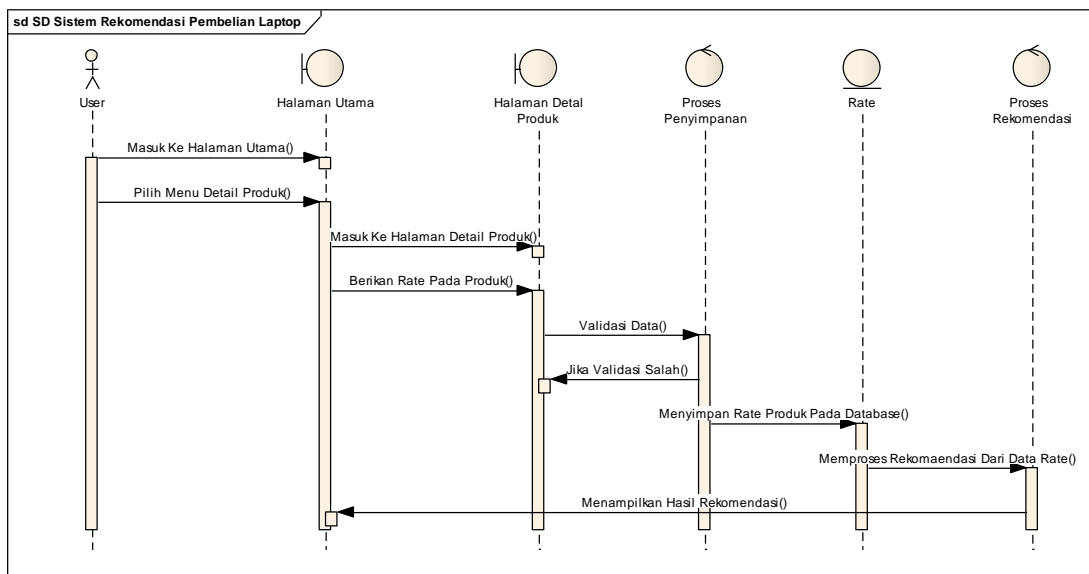
*Use case* diagram yang di gambarkan pada gambar 3 merupakan gambaran dari aksi-aksi yang dapat di lakukan oleh *user* (pemakai sistem rekomendasi) dan *admin*. Aksi-aksi tersebut meliputi menu-menu pembelian dan aksi rekomendasi yang di berikan kepada *user*.

*Sequence* diagram yang diacu pada gambar 4 menjelaskan bagaimana alur proses rekomendasi pada sistem pembelian laptop. Dari data *rate* produk yang di berikan oleh user kepada produk laptop yang disimpan ke dalam database kemudian hasil rekomendasi yang dikomputasi dengan metode *hybrid* tersebut di proses perhitungan *probabilitas* menggunakan Algoritma *Naive Bayes*. Kemudian hasil perhitungan probabilitas menghasilkan rekomendasi produk laptop yang akan di tampilkan pada halaman utama user. Gambar 5 menampilkan hasil akhir dari disain berupa *class diagram* yang menampilkan relasi antar *class*, dimana terdapat

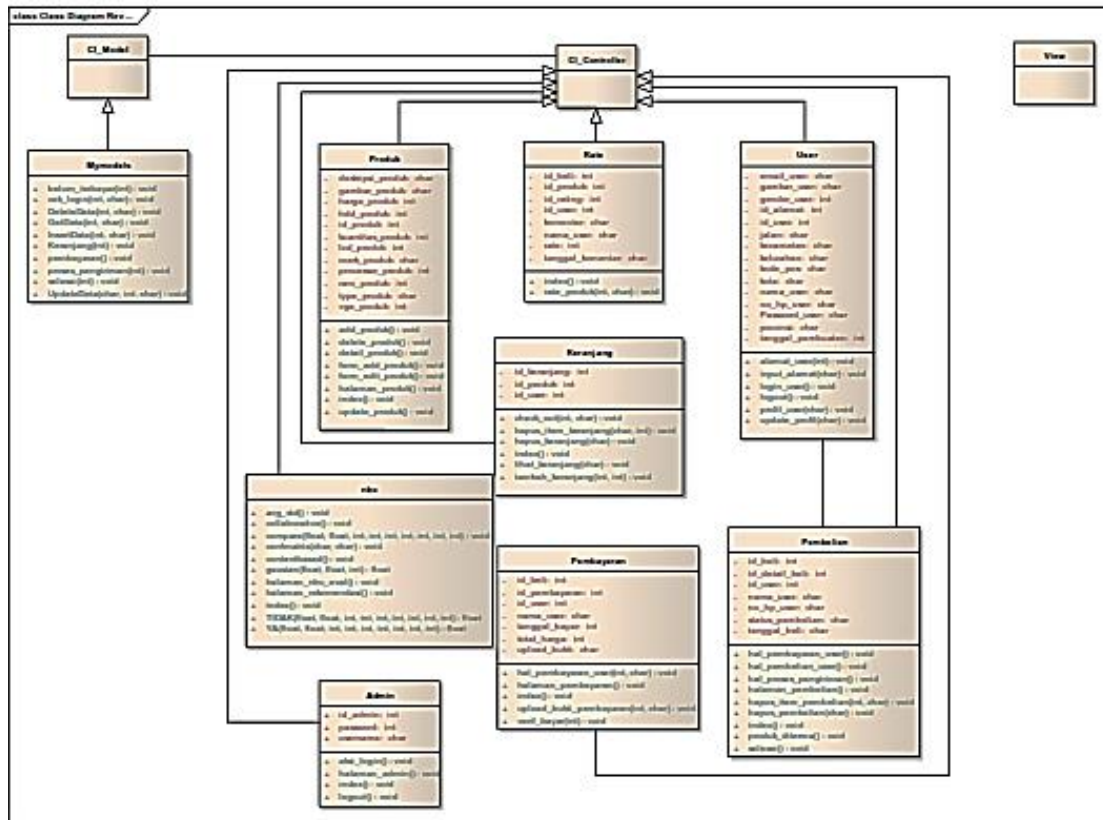
beberapa method utama yang dipakai dalam melakukan komputasi untuk sistem rekomendasi dan probabilitasnya.



Gambar 3. Use Case Diagram Sistem Rekomendasi



Gambar 4. Sequence Diagram Sistem Rekomendasi Pembelian Laptop



Gambar 5. Class Diagram Sistem Rekomendasi Pembelian Laptop.

Terdapat 3 metode utama dalam aplikasi yang dipakai dalam komputasi sistem rekomendasi, yaitu rerata standar deviasi (*avg\_std*), Gaussian dan *Compare*, sehingga pada tiga metode utama ini akan dilakukan uji *whitebox* dan unit (*phpunit testing*), berikut hasilnya di bawah ini :

```

avg_std(data) /*untuk menghitung rerata dan standar deviasi*/

train←getData(data)
set 0 to n_rekomen,n_tidakrekomen,n_data
set 0 to S_merk_r, S_ram_r, S_vga_r, S_hdd_r, S_lcd_r, S_proc_r, S_rate_r
set 0 to S_merk_tr, S_ram_tr, S_vga_tr, S_hdd_tr, S_lcd_tr, S_proc_tr, S_rate_tr
set 0 to U_merk_r, U_ram_r, U_vga_r, U_hdd_r, U_lcd_r, U_proc_r, U_rate_r
set 0 to U_merk_tr, U_ram_tr, U_vga_tr, U_hdd_tr, U_lcd_tr, U_proc_tr, U_rate_tr

for atribut in len(data.X)
  if atribut[1] then
    n_rekomen← n_rekomen+1, S_merk_r← atribut [merk]
    S_ram_r ← atribut [ram], S_vga_r ← atribut [vga]
    S_hdd_r ← atribut [hdd], S_lcd_r ← atribut [lcd]
    S_proc_r ← atribut [processor], S_rate_r ← atribut [rate]

  Else
    n_tidakrekomen ← n_tidakrekomen +1, S_merk_tr ← atribut [merk]
    S_ram_tr ← atribut [ram], S_vga_tr ← atribut [vga]
    S_hdd_tr ← atribut [hdd], S_lcd_tr ← atribut [lcd]
    S_proc_tr ← atribut [processor], S_rate_tr ← atribut [rate]
  n_data←n_data+1
  
```

```

U_merk_r ← S_merk_r/n_rekomen, U_ram_r ← S_ram_r/n_rekomen
U_vga_r ← S_vga_r/n_rekomen, U_hdd_r ← S_hdd_r/n_rekomen
U_lcd_r ← S_lcd_r/n_rekomen, U_proc_r ← S_proc_r/n_rekomen,
U_rate_r ← S_rate_r/n_rekomen
U_merk_tr ← S_merk_tr/n_tidakrekomen, U_ram_tr ← S_ram_r/n_tidakrekomen
U_vga_tr ← S_vga_r/n_tidakrekomen, U_hdd_tr ← S_hdd_r/n_tidakrekomen
U_lcd_tr ← S_lcd_r/n_tidakrekomen, U_proc_tr ← S_proc_r/n_tidakrekomen,
U_rate_tr ← S_rate_r/n_tidakrekomen

set 0 to sum2_merk_rekomen, sum2_ram_rekomen
,sum2_vga_rekomen,sum2_hdd_rekomen,sum2_lcd_rekomen
,sum2_processor_rekomen,sum2_rate_rekomen
set 0 to sum2_merk_tidakrekomen, sum2_ram_tidakrekomen, sum2_vga_tidakrekomen,
sum2_hdd_tidakrekomen, sum2_lcd_tidakrekomen, sum2_processor_tidakrekomen,
sum2_rate_tidakrekomen

for atribut in len(data.X)
  if atribut[1] then
    sum2_merk_rekomen ← pow(atribut [merk]- U_merk_r,2)
    sum2_ram_rekomen ← pow(atribut [ram]- U_ram_r,2)
    sum2_vga_rekomen ← pow(atribut [vga]- U_vga_r,2)
    sum2_hdd_rekomen ← pow(atribut [hdd]- U_hdd_r,2)
    sum2_lcd_rekomen ← pow(atribut [lcd]- U_lcd_r,2)
    sum2_processor_rekomen ← pow(atribut [proc]- U_proc_r,2)
    sum2_rate_rekomen ← pow(atribut [rate]- U_rate_r,2)

  Else
    sum2_merk_tidakrekomen ← pow(atribut [merk]- U_merk_tr,2)
    sum2_ram_tidakrekomen ← pow(atribut [ram]- U_ram_tr,2)
    sum2_vga_tidakrekomen ← pow(atribut [vga]- U_vga_tr,2)
    sum2_hdd_tidakrekomen ← pow(atribut [hdd]- U_hdd_tr,2)
    sum2_lcd_tidakrekomen ← pow(atribut [lcd]- U_lcd_tr,2)
    sum2_processor_tidakrekomen ← pow(atribut [proc]- U_proc_tr,2)
    sum2_rate_tidakrekomen ← pow(atribut [rate]- U_rate_tr,2)

std_merk_rekomen ← sqrt(sum2_merk_rekomen/(n_rekomen-1))
std_ram_rekomen ← sqrt(sum2_ram_rekomen/(n_rekomen-1))
std_vga_rekomen ← sqrt(sum2_vga_rekomen/(n_rekomen-1))
std_hdd_rekomen ← sqrt(sum2_hdd_rekomen/(n_rekomen-1))
std_lcd_rekomen ← sqrt(sum2_lcd_rekomen/(n_rekomen-1))
std_processor_rekomen ← sqrt(sum2_processor_rekomen/(n_rekomen-1))
$std_rate_rekomen ← sqrt(sum2_rate_rekomen/(n_rekomen-1))
std_merk_tidakrekomen ← sqrt(sum2_merk_tidakrekomen/(n_tidakrekomen -1))
std_ram_tidakrekomen ← sqrt(sum2_ram_tidakrekomen/(n_tidakrekomen -1))
std_vga_tidakrekomen ← sqrt(sum2_vga_tidakrekomen/(n_tidakrekomen -1))
std_hdd_tidakrekomen ← sqrt(sum2_hdd_tidakrekomen/(n_tidakrekomen -1))
std_lcd_tidakrekomen ← sqrt(sum2_lcd_tidakrekomen/(n_tidakrekomen -1))
std_processor_tidakrekomen ← sqrt(sum2_processor_tidakrekomen/(n_tidakrekomen -1))
std_rate_tidakrekomen ← sqrt(sum2_rate_tidakrekomen/(n_tidakrekomen -1))
std[0] ← std_merk_rekomen, std[1] ← std_ram_rekomen, std[2] ← std_vga_rekomen,
std[3] ← std_hdd_rekomen, std[4] ← std_lcd_rekomen, std[5] ← std_processor_rekomen,

```

```

std[6] ← std_rate_rekomen, std[7]← std_merk_tidakrekomen, std[8]←
std_ram_tidakrekomen, std[9]← std_vga_tidakrekomen, std[10] ←
std_hdd_tidakrekomen, std[11] ← std_lcd_tidakrekomen, std[12] ←
std_processor_tidakrekomen, std[13] ← std_rate_tidakrekomen
avg[0] ← u_merk_rekomen, avg[1] ← u_ram_rekomen ,avg[2] ← u_vga_rekomen,
avg[3] ← u_hdd_rekomen, avg[4] ← u_lcd_rekomen, avg[5]← u_processor_rekomen
avg[6]← u_rate_rekomen, avg[7] ← u_merk_tidakrekomen, avg[8] ←
u_ram_tidakrekomen, avg[9] ← u_vga_tidakrekomen, avg[10] ←
u_hdd_tidakrekomen, avg[11] ← u_lcd_tidakrekomen, avg[12]←
u_processor_tidakrekomen, avg[13] ← u_rate_tidakrekomen

Return avg [avg], std[std]

```

Dari algoritma diatas didapatkan *Cyclomatic Complexity* V(G) sebesar 6 (6 path). Algoritma Gaussian dapat dilihat pada notasi berikut :

```

Gaussian(x , u, std)
G ← (1.0/(std* sqrt(2*3.1415926535))) * exp(-((x-u)*(x-u)) / (2.0 * std * std))
Return G

```

Dari algoritma diatas didapatkan *Cyclomatic Complexity* V(G) sebesar 1 (1 path), dan terakhir fungsi *Compare* sebagai berikut :

```

compare(merk,ram,vga,hdd,lcd,processor,rate,avg,std)
YA ← gaussian(merk,avg[0],std[0]) * gaussian(ram,avg[1],std[1]) *
gaussian(vga,avg[2],std[2]) * gaussian(hdd,avg[3],std[3]) * gaussian(lcd,avg[4],std[4])*
gaussian(processor,avg[5],std[5])* gaussian(rate,avg[6],std[6])
TIDAK←gaussian(merk,avg[7],std[7]) * gaussian(ram,avg[8],std[8]) *
gaussian(vga,avg[9],std[9]) * gaussian(hdd,avg[10],std[10]) *
gaussian(lcd,avg[11],std[11])*gaussian(processor,avg[12],std[12])*
gaussian(rate,avg[13],std[13])
Return Y>=Tidak

```

Dari algoritma diatas didapatkan *Cyclomatic Complexity* V(G) sebesar 2 (2 path). Implementasi dari rancangan diatas dapat dilihat pada gambar. 6 dalam bentuk *web based*.

Dengan perhitungan probabilitas menggunakan Algoritma Naïve Bayes pada *content based* dan *collaborative filtering* di hasilkan output berupa hasil rekomendasi produk laptop yang terlihat pada gambar 6. Dimana rekomendasi **Content Based Filtering** merupakan hasil rekomendasi yang di berikan kepada user yang telah teregistrasi dimana hasil rekomendasi berdasarkan apa yang telah user di beli dan di beri rating oleh user itu sendiri. Sementara hasil **Collaborative Filtering** merupakan rekomendasi yang di berikan user yang berdasarkan produk laptop yang pernah di beli dan di beri rating oleh user lain yang telah teregistrasi. Untuk mengetahui tingkat akurasi dan *error rate* dari Algoritma Naïve Bayes yang di lakukan pengujian menggunakan *Confusion Matrix*. Dari pengujian terhadap 20 data uji dari 190 data training yang ada, menggunakan Algoritma Naïve Bayes, diperoleh hasil seperti yang di gambarkan pada tabel 1.





Gambar 6. Tampilan *Collaborative Filtering* dan *Content Based Filtering*.

Tabel 1. Hasil Prediksi Algoritma Naïve Bayes

No	RAM	Merk	VGA	Harddisk	LCD	Prosesor	Rate	Prediksi
1	1	2	2	500	11	1	1	TIDAK
2	2	2	1	500	14	2	2	YA
3	6	4	1	320	11	2	1	TIDAK
4	1	4	1	512	17	2	2	YA
5	1	4	1	500	14	2	1	TIDAK
6	3	2	1	500	14	3	2	YA
7	3	6	2	1000	15	3	2	TIDAK
8	4	2	2	320	14	2	1	TIDAK
9	5	4	2	1080	14	2	2	YA
10	6	2	2	300	13	2	1	TIDAK
11	1	8	1	1000	15	4	5	YA
12	2	4	1	750	14	3	3	YA
13	2	4	1	1000	14	3	4	YA
14	2	4	1	1000	15	3	4	YA
15	2	8	1	1000	14	3	4	YA
16	3	4	1	320	14	3	3	YA
17	4	4	1	500	15	2	4	YA
18	5	4	1	500	14	1	3	YA
19	5	8	1	1000	14	3	4	YA
20	5	2	2	320	11	3	5	YA

Kemudian dari tabel 1 di translasikan ke dalam tabel *confusion matrix* untuk dievaluasi akurasi seperti terlihat pada tabel 2.

Tabel 2. *Confussion Matrix*

		Predicted	
		Positif	Negatif
Kelas	Positif	TP : 10	FN : 0
	Negatif	FP : 4	TN : 6

Dari tabel *Confussion Matrix* yang di acu pada tabel 2 di peroleh hasil akurasi dan *error rate* sebagai berikut :

$$Akurasi = \frac{10 + 6}{10 + 6 + 4 + 0} \times 100\% = 80\%$$

$$Error Rate = \frac{4 + 0}{10 + 6 + 4 + 0} \times 100\% = 20\%$$

#### 4. KESIMPULAN

Berdasarkan analisa dan hasil pembahasan, maka dapat disimpulkan :

- Metode *Hybrid Collaborative* dan *Content Based Filltering* dapat di implementasikan pada Sistem Rekomendasi Pembelian Laptop.
- Dari 20 data uji dan 190 data training di peroleh hasil akurasi 80% dan *error rate* 20% pada Algoritma Naïve Bayes dengan menggunakan pengujian *confussion matrix*.
- Motode *Hybrid Filtering* dan Klasifikasi Algoritma Niave Bayes berhasil lolos uji *Whitebox* yang di implementasikan pada Sistem Rekomendasi Pembelian Laptop.

#### 5. SARAN

Ke depan perlu dibuatkan sistem rekomendasi dengan melakukan perbandingan metode filtering dan klasifikasi yang lain, serta melakukan *tuning*, *boosting* atau *bagging* untuk meningkatkan model. Selain itu komentar user perlu dimasukan kedalam fitur untuk mendapatkan data yang tidak terlihat sehingga hasil klasifikasi semakin presisi.

#### DAFTAR PUSTAKA

- [1] S. S. Srinivasan, R. Anderson, and K. Ponnnavolu, "Customer loyalty in e-commerce: An exploration of its antecedents and consequences," *J. Retail.*, vol. 78, no. 1, pp. 41–50, 2002.
- [2] K. C. Laudon and C. Guercio Traver, "E -commerce: business, technology, society," *Business, Technol. Soc.*, 2007.
- [3] J. B. E. N. Schafer, "E-Commerce Recommendation Applications," pp. 115–153, 2001.
- [4] Y. Li, C. Wu, and C. Lai, "A social recommender mechanism for e-commerce : Combining similarity, trust , and relationship," *Decis. Support Syst.*, vol. 55, no. 3, pp. 740–752, 2013.
- [5] J. Ilmiah, I. Komputa, H. Setiawan, and S. Hansun, "Rancang Bangun Aplikasi Rekomendasi Pembelian Laptop dengan Database Fuzzy Model Tahani Berbasis Web Program Studi Teknik Informatika , Universitas Multimedia Nusantara , Tangerang ,

- Indonesia Jurnal Ilmiah Komputer dan Informatika ( KOMPUTA ),” vol. 3, no. 2, 2014.
- [6] J. Wang and Y. Zhang, “Opportunity Models for E-commerce Recommendation : Right Product , Right Time,” pp. 303–312, 2013.
- [7] Y. Shoham, “Fab : Content-Based, Collaborative Recommendation,” *Commun. ACM*, vol. 40, no. 3, 1997.
- [8] F. Gorunescu, *Data Mining: Concepts and Techniques*, vol. 12. 2011.
- [9] V. No, “Sistem Rekomendasi Bacaan Tugas Akhir Jurusan Teknik Informatika Universitas Sriwijaya menggunakan Metode Collaborative Filtering dan Naive Bayes,” vol. 2, no. 1, pp. 343–347, 2016.