



Rekomendasi Pemilihan Restoran Berdasarkan Rating Online Menggunakan Algoritma C4.5

Muhammad Farid, Devi Fitriana*

*Jurusan Teknik Informatika, Fakultas Ilmu Komputer, Universitas Mercu Buana,
Jl. Meruya Selatan, Jakarta 11650, Indonesia*

* Email Penulis Koresponden: devi.fitriana@mercubuana.ac.id

Abstrak:

Restoran adalah suatu usaha yang menyediakan tempat untuk menikmati hidangan kepada pelanggan serta menetapkan tarif tertentu. Tersedianya banyak pilihan restoran menjadi faktor penting yang dibutuhkan dalam memilih restoran. Masalah akan muncul secara langsung akibat banyaknya restoran yang tersedia sehingga membutuhkan waktu yang lama untuk menentukan pilihan. Hal ini disebabkan oleh penyebaran informasi yang tidak merata dan pengambilan keputusan yang tidak akurat sehingga pelanggan kesulitan untuk menentukan pilihan restoran. Dengan adanya rekomendasi pemilihan restoran untuk pelanggan akan mendukung pengambilan keputusan dalam menentukan restoran atau tempat makan. Latar belakang dipilihnya algoritma C4.5 sebagai pengambilan keputusan untuk menentukan rekomendasi pemilihan restoran berdasarkan *rating*. Data yang digunakan diambil dari Zomato API untuk dilakukan pengujian dengan menggunakan 1003 sampel data restoran. Hasil yang didapatkan dengan *ten-fold cross validation* yaitu menghasilkan akurasi 86,24% dengan *rating* yang paling dominan dan sesuai untuk direkomendasi adalah *Rating Good*. Hal ini diharapkan dapat memberikan rekomendasi untuk menentukan beberapa pilihan restoran yang sesuai untuk dikunjungi berdasarkan *rating* tersebut.

This is an open access article under the [CC BY-NC](https://creativecommons.org/licenses/by-nc/4.0/) license



Katakunci:

Algoritma C4.5;
Decision Tree;
Klasifikasi;
Rekomendasi;

Riwayat Artikel:

Diserahkan 1 Oktober 2020
Direvisi 4 Januari 2021
Diterima 11 Januari 2021
Dipublikasi 1 April 2021

DOI:

10.22441/incomtech.v11i1.9791

1. PENDAHULUAN

Restoran adalah suatu usaha yang menyajikan dan menyediakan tempat untuk menikmati hidangan kepada pelanggan serta menetapkan tarif tertentu. Berbagai cara penyajian restoran melayani seperti *dine in*, *take away*, dan *delivery*. Adapun restoran yang ada saat ini berusaha untuk memberikan yang terbaik dalam menarik minat pelanggan agar mengunjungi restorannya. Begitupun bisnis restoran hingga saat ini masih diyakini sebagai salah satu bisnis yang memiliki prospek yang menjanjikan dan mendapatkan keuntungan cukup menggugurkan [1].

Dengan target pasar yang begitu kompetitif membuat pelaku bisnis lebih selektif dalam menjalankan strategi bisnis untuk merebut dan mempertahankan pelanggan dengan menawarkan berbagai cara dalam menarik minat pelanggan [2].

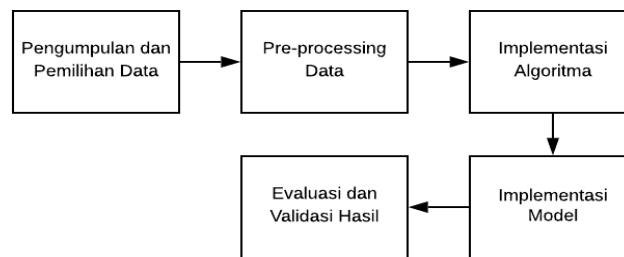
Biasanya restoran bergantung pada rasa makanan, variasi masakan yang disediakan, suasana restoran, ketersediaan pesan antar, pemesanan *online*, dan harga yang ditawarkan. Ketika salah satu faktor dimasukkan, ada kemungkinan untuk meningkatkan perhatian pelanggan [3]. Begitupun dengan tersedianya banyak pilihan restoran yang menjadi faktor penting yang dibutuhkan dalam pemilihan restoran seperti jenis makanan, lokasi, fasilitas yang ditawarkan, kisaran harga, serta *rating* sebagai tolak ukur dalam memilih restoran. Masalah akan muncul jika memilih restoran secara langsung akibat banyaknya restoran yang tersedia adalah kebingungan untuk memilih restoran sehingga membutuhkan waktu yang lama untuk menentukan pilihan. Hal ini disebabkan oleh penyebaran informasi yang tidak merata dan pengambilan keputusan yang tidak akurat sehingga pelanggan kesulitan untuk menentukan pilihan restoran saat mengunjungi secara langsung [4]. Dengan adanya rekomendasi pemilihan restoran untuk pelanggan akan mendukung pengambilan keputusan dalam menentukan restoran atau tempat makan.

Pada penelitian terkait mengenai rekomendasi restoran atau tempat makan diantaranya, sistem rekomendasi restoran dengan mengembangkan sistem menggunakan metode AHP dan VIKOR dengan LINE *Message* API dalam bentuk *carousel message* yang berguna untuk membantu pengguna dalam mencari restoran [5], dan sistem rekomendasi tempat makan dengan mengimplementasi algoritma *Squeezer* dan *Term Frequency Ranking* yang berguna untuk membantu memberikan rekomendasi tempat makan terhadap pengguna [6]. Pada penelitian lainnya mengenai algoritma C4.5 yaitu, melakukan penelitian untuk menerapkan teknik *data mining* metode *decision tree* algoritma C4.5 pada bisnis gerai makanan cepat saji XYZ dan diharapkan dapat memberikan informasi berupa klasifikasi penjualan menu makanan yang paling digemari pelanggan dan kurang digemari (laris dan tidak laris) [7].

Pada penelitian ini akan melakukan rekomendasi pemilihan restoran berdasarkan *rating online* [8][9] menggunakan algoritma C4.5 dan data restoran diambil dari Zomato API. Latar belakang dipilihnya algoritma C4.5 sudah banyak digunakan dalam implementasi pengambilan keputusan karena memiliki banyak kelebihan dan dengan data yang ada dapat digunakan untuk menentukan rekomendasi [10][11]. Oleh karena itu, penelitian ini bermaksud untuk membuat rekomendasi pemilihan restoran menggunakan teknik klasifikasi dengan algoritma C4.5. Tujuan dari penelitian ini adalah memberikan rekomendasi untuk menentukan beberapa pilihan restoran berdasarkan *rating* yang sesuai menggunakan teknik *data mining* dengan data Zomato API.

2. METODE

Terdapat beberapa tahapan metode dalam penelitian ini, sebagaimana ditunjukkan pada [Gambar 1](#).



Gambar 1. Block Diagram Penelitian

3.1. Pengumpulan dan Pemilihan Data

Data yang digunakan dalam penelitian ini adalah restoran yang terdapat di *website* Zomato. Data tersebut diambil menggunakan Zomato API yang disediakan oleh Zomato kepada pengembang dengan format JSON (*JavaScript Object Notation*). Total dataset yang digunakan berisi 1003 data. Data restoran meliputi *restaurant id*, *restaurant name*, *cuisines*, *average cost for two*, *price range*, *rating*, *votes range*, *locality*, serta tambahan faktor fasilitas pendukung.

3.2. Pre-processing Data

Pre-processing data adalah suatu proses yang bertujuan untuk meningkatkan kualitas data dengan menghapus data yang tidak diperlukan dari data asli. Dalam penelitian ini *pre-processing* yang dilakukan antara lain:

a. Data Cleaning

Data cleaning atau pembersihan data merupakan proses yang paling penting dalam pengolahan data agar data benar-benar relevan dengan kebutuhan, konsisten, dan tidak mengalami redundansi [12]. Pada tahap ini peneliti melakukan pembersihan data untuk membersihkan *noise* pada data, seperti adanya duplikat data dan data yang kosong. Lalu setelah itu semua atribut kemudian akan dipilih dan diseleksi untuk mendapatkan atribut yang mengandung nilai yang paling konsisten dan relevan. Dalam penelitian ini atribut yang digunakan yaitu, *restaurant name* sebagai id, *rating* sebagai label, *price range*, *votes range*, serta tambahan faktor fasilitas pendukung (*wifi*, *booking*, *delivery*, *takeaway*, *smoking area*, *indoor seat*, *outdoor seat*, *private dining*, *valet parking*, *pray room*).

b. Data Transformation

Pada tahap ini melakukan transformasi data untuk mengubah data menjadi nilai dengan format yang sesuai untuk diproses dalam *data mining*. Dalam penelitian ini atribut *price range* dan atribut *votes range* yang berupa numerik dikonversi menjadi kategorik. Hal ini dilakukan untuk membuat data lebih mudah diproses. Konversi nilai pada atribut tersebut dapat dilihat pada [Tabel 1](#) dan [Tabel 2](#).

3.3. Implementasi Algoritma

Implementasi algoritma dilakukan dengan menghitung nilai *entropy* dan *information gain* menggunakan rumus. Kemudian, nilai *entropy* dan *information gain* yang diperoleh akan digunakan untuk membuat simpul pohon keputusan [13][14].

Tabel 1. Transformasi Data Price Range

Price Range	
Value	Transformasi
1	Murah
2	Sedang
3	Mahal
4	Sangat Mahal

Tabel 2. Transformasi Data Votes Range

Votes Range	
Value	Transformasi
0	Not Voted
1 - 250	Rendah
251 - 500	Sedang
501 - 750	Tinggi
751 - seterusnya	Sangat Tinggi

Contoh salah satu atribut dalam perhitungan nilai *entropy* dan *gain* adalah [15, 16, 17]:

$$Entropy(Total) = 1,742162234 \quad (1)$$

Dengan menggunakan (1) dapat menghitung nilai *entropy*, menghitung *Entropy Total* dilakukan dengan cara menghitung jumlah keputusan seperti, “*Poor*”, “*Average*”, “*Good*”, “*Very Good*”, dan “*Excellent*” dari seluruh kasus yang ada. *Entropy Total* adalah menghitung nilai total keputusan dari *Poor* (14), *Average* (383), *Good* (409), *Very Good* (160) dan *Excellent* (37), sedangkan 1003 adalah jumlah keseluruhan kasus.

Kemudian hitung masing-masing *entropy* dari semua nilai variabel yang ada. Dan dengan menggunakan (2) dapat menghitung nilai *information gain* tiap-tiap atribut [18, 19, 20].

Contoh menghitung *gain* pada salah satu atribut sebagai berikut:

$$Gain(Total, Price Range) = 1,742162234 - \left[\left(\frac{229}{1003} * 1,202106563 \right) + \left(\frac{390}{1003} * 1,405072172 \right) + \left(\frac{346}{1003} * 1,80387618 \right) + \left(\frac{38}{1003} * 1,80387618 \right) \right] = 0,29908974 \quad (2)$$

Hasil perhitungan nilai *entropy* dan *information gain* secara keseluruhan dapat dilihat pada Tabel 3.

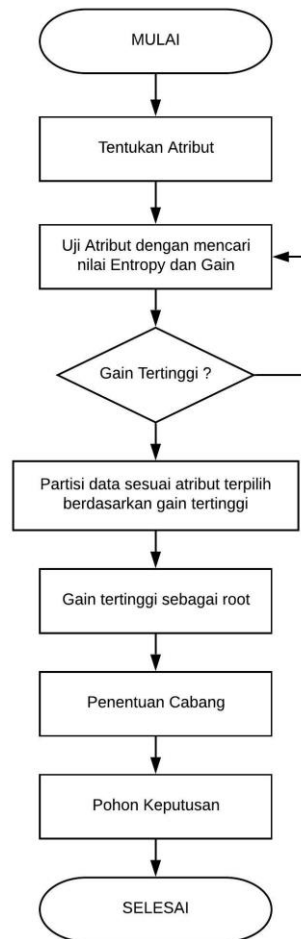
Berdasarkan perhitungan semua nilai *entropy* dan *information gain* pada Tabel 3, diperoleh nilai *information gain* terbesar adalah atribut “*Votes Range*” dengan nilai 0,559438185. Kemudian nilai *information gain* terbesar dijadikan simpul akar atau *node root* dalam pohon keputusan. Lalu hilangkan atribut yang dipilih sebelumnya dan ulangi perhitungan nilai *entropy*, *information gain*, dengan memilih *information gain* terbesar dan dijadikan simpul internal pohon. Ulangi perhitungan tersebut hingga semua atribut pohon memiliki kelas.

Secara keseluruhan algoritma klasifikasi C4.5 dapat dilihat pada diagram alur pada Gambar 2.

Tabel 3. Hasil Perhitungan Entropy dan Information Gain

Atribut	Keterangan	Total	Poor	Average	Good	Very Good	Excellent	Entropy	Information Gain
Total		1003	14	383	409	160	37	1,74216223	
Price Range									0,299089743
	Murah	229	6	144	75	3	1	1,20210656	
	Sedang	390	3	173	186	23	5	1,40507217	
	Mahal	346	5	66	144	115	16	1,80387618	
	Sangat Mahal	38	0	0	4	19	15	0	
Votes Range									0,559438185
	Not Voted	10	10	0	0	0	0	0	
	Rendah	848	3	382	403	59	1	1,33614338	
	Sedang	73	1	1	6	62	3	0,72907719	
	Tinggi	32	0	0	0	25	7	0	
	Sangat Tinggi	40	0	0	0	14	26	0	
Wifi									0,3516515
	Ada	495	3	33	296	130	33	1,51574752	
	Tidak Ada	508	11	350	113	30	4	1,26847883	
Booking									0,233566679
	Ada	272	1	25	97	114	35	1,78318961	
	Tidak Ada	731	13	358	312	46	2	1,40642102	
Delivery									0,425726186
	Ada	278	0	109	135	31	3	0	
	Tidak Ada	725	14	274	274	129	34	1,82122118	
Takeaway									0,018970907
	Ada	894	13	359	366	127	29	1,7052151	
	Tidak Ada	109	1	24	43	33	8	1,87062937	
Smoking Area									0,159801427
	Ada	278	2	16	159	86	15	1,5001982	
	Tidak Ada	725	12	367	250	74	12	1,61386592	
Indoor Seat									0,152903477
	Ada	913	9	326	384	157	37	1,74592172	
	Tidak Ada	90	15	57	25	2	0	0	
Outdoor Seat									0,027984587
	Ada	238	2	65	93	68	10	1,80758066	
	Tidak Ada	765	12	318	316	92	27	1,68511893	
Private Dining									0,183961576
	Ada	64	0	0	23	23	18	0	
	Tidak Ada	939	14	383	386	137	19	1,6644039	
Valet Parking									0,089581492
	Ada	26	0	0	2	14	10	0	
	Tidak Ada	977	14	383	407	146	27	1,69655935	
Pray Room									0,17838376
	Ada	90	0	6	50	28	6	0	
	Tidak Ada	913	14	377	359	132	31	1,71792969	

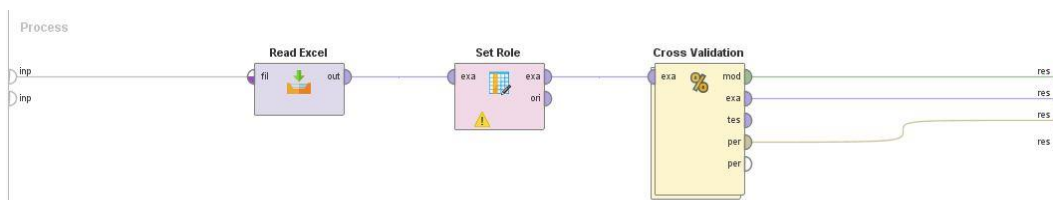
Algoritma ini dimulai dengan menentukan atribut yang akan dijadikan sebagai node. Setiap atribut yang telah ditentukan selanjutnya akan dilakukan perhitungan nilai entropy dan gain. Hal ini dilakukan untuk dapat menentukan posisi dari masing-masing atribut didalam pohon keputusan. Setelah mendapatkan atribut dengan nilai gain tertinggi maka atribut atau node tadi akan dijadikan sebagai root. Lakukan hal tersebut dengan melakukan partisi pohon sampai dengan tidak ada lagi atribut atau node yang tersisa, sehingga menjadi cabang-cabang. Setelah itu maka terbentuklah pohon keputusan.



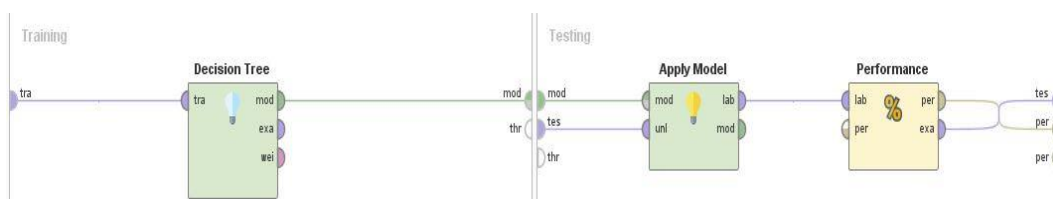
Gambar 2 Diagram Alir Algoritma C4.5

3.4. Implementasi Model

Pada tahap selanjutnya, model diuji menggunakan *tools* Rapidminer. Penjelasan masing-masing operator ditunjukkan pada [Gambar 3](#) dan [Gambar 4](#).



Gambar 3. Operator Pengujian



Gambar 4. Operator Cross Validation

Berdasarkan [Gambar 3](#), pada operator pengujian dapat dilihat yang berisi operator *read excel* dan operator *set role* diuji menggunakan operator *cross validation* yaitu operator yang berfungsi membagi data *training* dan data *testing* dalam beberapa bagian. Berikut penjelasan masing-masing operator:

- a. Operator *Read Excel*: Operator ini dapat digunakan untuk mengimpor atau memuat data dari *spreadsheet* Microsoft Excel. Pengguna dapat menentukan tabel data yang memiliki format sedemikian rupa sehingga setiap baris adalah contoh dan setiap kolom mewakili atribut.
- b. Operator *Set Role*: Operator ini digunakan untuk mengubah peran satu atribut atau lebih. Pada penelitian ini peran target atribut atau sebagai atribut spesial yang digunakan untuk klasifikasi adalah id sebagai atribut identifikasi dan label sebagai atribut target.
- c. Operator *Cross Validation*: Operator ini memiliki dua subproses yaitu *training* dan *testing*. Subproses *training* digunakan untuk melatih model. Model yang terlatih kemudian diterapkan dalam subproses *testing*. *Input ExampleSet* dipartisi ke dalam subset k dengan ukuran yang sama. Dari himpunan himpunan bagian, satu himpunan bagian dipertahankan sebagai set data uji (yaitu input dari subproses *testing*). Subset $k - 1$ yang tersisa digunakan sebagai kumpulan data pelatihan (yaitu input dari subproses *training*). Proses validasi silang kemudian diulang k kali, dengan masing-masing himpunan bagian k digunakan tepat sekali sebagai data uji. Hasil k dari iterasi k dirata-rata (atau dikombinasikan) untuk menghasilkan estimasi tunggal. Nilai k dapat disesuaikan menggunakan jumlah parameter lipatan.

Berdasarkan [Gambar 4](#), operator *cross validation* terdapat operator lain yaitu operator *decision tree* sebagai *data training*, sedangkan operator *apply model* dan operator *performance* sebagai *data testing*. Berikut penjelasan masing-masing operator:

- d. Operator *Decision Tree*: Operator ini sebagai *training* yang digunakan untuk membuat keputusan dengan algoritma C4.5. Pohon keputusan adalah pohon itu terbentuk dari kumpulan *node*. Setiap *node* mewakili aturan pemisahan untuk satu atribut tertentu dan untuk aturan klasifikasi memisahkan nilai-nilai kelas yang berbeda.
- e. Operator *Apply Model*: Operator ini sebagai *testing* yang digunakan setelah model pohon keputusan diterapkan lalu model ini dapat diterapkan pada *ExampleSet* lain. Tujuannya adalah untuk mendapatkan prediksi pada data yang tidak terlihat atau mengubah data dengan menerapkan model *pre-processing* dan membaca data yang akan diestimasi berdasarkan *data training* yang telah dipelajari.
- f. Operator *Performance*: Operator ini sebagai *testing* yang digunakan untuk evaluasi menentukan jenis tugas pembelajaran secara otomatis menentukan jenis tugas belajar dan menghitung nilai yang paling banyak kriteria kinerja umum untuk jenis itu. Pada penelitian ini kriteria dalam klasifikasi berupa polinomial yang menghasilkan kriteria *Accuracy* dan *Kappa statistic*.

3.5. Evaluasi dan Validasi Hasil

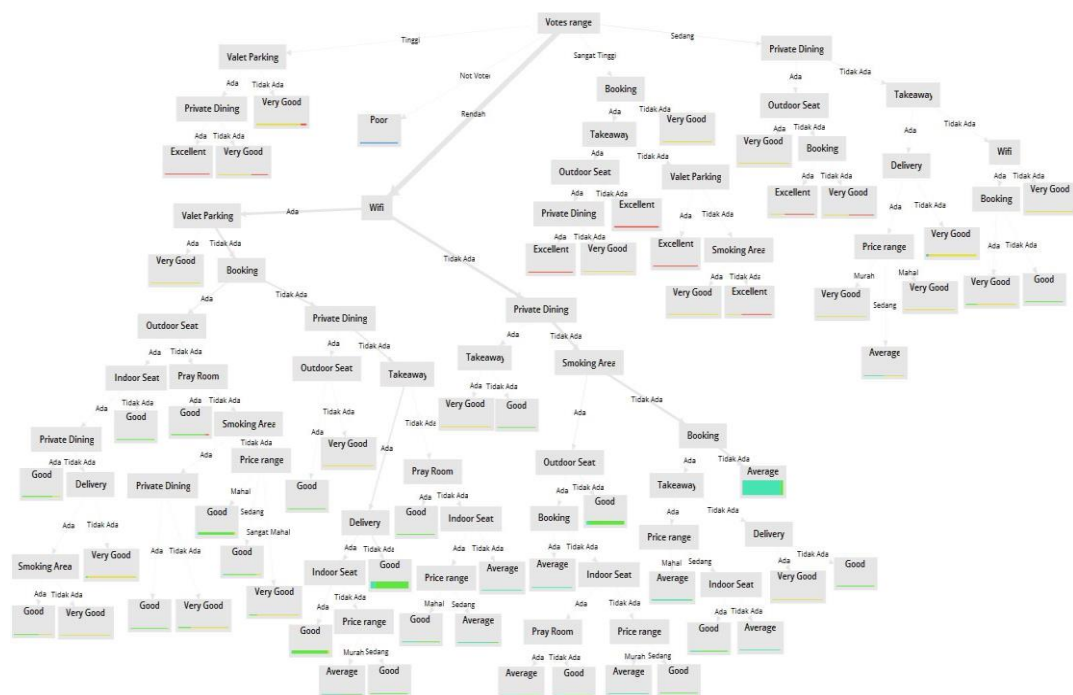
Berdasarkan evaluasi pengujian dilakukan dengan metode *Ten-fold Cross Validation*. *Ten-fold* adalah salah satu K -fold yang direkomendasikan untuk

pemilihan model terbaik karena dapat mengurangi waktu komputasi dengan tetap menjaga keakuratan estimasi. *Ten-fold Cross Validation* adalah validasi yang dilakukan dengan cara membagi suatu set data menjadi sepuluh segmen yang berukuran sama besar dengan cara melakukan pengacakan data. Dalam penelitian ini, teknik validasi silang k-fold digunakan dengan nilai $k = 10$. Yang dimana data dibagi menjadi 10-fold berukuran kira-kira sama, sehingga kita memiliki 10 subset data untuk mengevaluasi kinerja model atau algoritma. Untuk masing-masing dari 10 subset data tersebut, *cross validation* akan menggunakan 9-fold untuk pelatihan dan 1-fold untuk pengujian. Pemilihan nilai $k=10$ dalam penelitian ini lebih kepada untuk mendapatkan hasil yang tidak bias dimana dengan jumlah sample data yang digunakan pengujian terhadap model dapat terakomodasi [3]

Validasi dan pengujian dilakukan untuk mengetahui tingkat akurasi, presisi, dan *recall* dari hasil prediksi klasifikasi. Akurasi adalah persentase dari catatan yang diklasifikasikan dengan benar dalam pengujian dataset. Presisi adalah persentase data yang diklasifikasikan sebagai model baik yang sebenarnya juga baik. *Recall* adalah pengukuran tingkat pengenalan positif sebenarnya [21][22].

3. HASIL DAN PEMBAHASAN

Berdasarkan hasil perhitungan manual *entropy* dan *information gain* pada Tabel 3, atribut *Votes Range* dipilih sebagai *node root* atau simpul akar pohon keputusan karena mendapatkan nilai *information gain* tertinggi yaitu 0,559438185. Setelah itu dilanjutkan memilih *node* hingga semua atribut memiliki kelas. Jika semua atribut sudah memiliki kelas, maka pohon keputusan akan dibentuk seperti yang ditunjukkan pada Gambar 5.



Gambar 5. Model Pohon Keputusan yang Terbentuk

Berdasarkan dari deskripsi hasil pohon keputusan yang terbentuk pada [Gambar 5](#), ada 57 aturan atau *rule model* yang terbentuk dapat dilihat sebagai berikut;

1. *If Votes range = Not Voted Then Rating = Poor*
2. *If Votes range = Rendah, Wifi = Ada, Valet Parking = Ada Then Rating = Very Good*
3. *If Votes range = Rendah, Wifi = Ada, Valet Parking = Tidak Ada, Booking = Ada, Outdoor Seat = Ada, Indoor Seat = Ada, Private Dining = Ada Then Rating = Good*
4. *If Votes range = Rendah, Wifi = Ada, Valet Parking = Tidak Ada, Booking = Ada, Outdoor Seat = Ada, Indoor Seat = Ada, Private Dining = Tidak Ada, Delivery = Ada, Smoking Area = Ada Then Rating = Good*
5. *If Votes range = Rendah, Wifi = Ada, Valet Parking = Tidak Ada, Booking = Ada, Outdoor Seat = Ada, Indoor Seat = Ada, Private Dining = Tidak Ada, Delivery = Ada, Smoking Area = Tidak Ada Then Rating = Very Good*
6. *If Votes range = Rendah, Wifi = Ada, Valet Parking = Tidak Ada, Booking = Ada, Outdoor Seat = Ada, Indoor Seat = Ada, Private Dining = Tidak Ada, Delivery = Tidak Ada Then Rating = Very Good*
7. *If Votes range = Rendah, Wifi = Ada, Valet Parking = Tidak Ada, Booking = Ada, Outdoor Seat = Ada, Indoor Seat = Tidak Ada Then Rating = Good*
8. *If Votes range = Rendah, Wifi = Ada, Valet Parking = Tidak Ada, Booking = Ada, Outdoor Seat = Tidak Ada, Pray Room = Ada Then Rating = Good*
9. *If Votes range = Rendah, Wifi = Ada, Valet Parking = Tidak Ada, Booking = Ada, Outdoor Seat = Tidak Ada, Pray Room = Tidak Ada, Smoking Area = Ada, Private Dining = Ada Then Rating = Good*
10. *If Votes range = Rendah, Wifi = Ada, Valet Parking = Tidak Ada, Booking = Ada, Outdoor Seat = Tidak Ada, Pray Room = Tidak Ada, Smoking Area = Ada, Private Dining = Tidak Ada Then Rating = Very Good*
11. *If Votes range = Rendah, Wifi = Ada, Valet Parking = Tidak Ada, Booking = Ada, Outdoor Seat = Tidak Ada, Pray Room = Tidak Ada, Smoking Area = Tidak Ada, Price range = Mahal Then Rating = Good*
12. *If Votes range = Rendah, Wifi = Ada, Valet Parking = Tidak Ada, Booking = Ada, Outdoor Seat = Tidak Ada, Pray Room = Tidak Ada, Smoking Area = Tidak Ada, Price range = Sangat Mahal Then Rating = Very Good*
13. *If Votes range = Rendah, Wifi = Ada, Valet Parking = Tidak Ada, Booking = Ada, Outdoor Seat = Tidak Ada, Pray Room = Tidak Ada, Smoking Area = Tidak Ada, Price range = Sedang Then Rating = Good*
14. *If Votes range = Rendah, Wifi = Ada, Valet Parking = Tidak Ada, Booking = Tidak Ada, Private Dining = Ada, Outdoor Seat = Ada Then Rating = Good*
15. *If Votes range = Rendah, Wifi = Ada, Valet Parking = Tidak Ada, Booking = Tidak Ada, Private Dining = Ada, Outdoor Seat = Tidak Ada Then Rating = Very Good*
16. *If Votes range = Rendah, Wifi = Ada, Valet Parking = Tidak Ada, Booking = Tidak Ada, Private Dining = Tidak Ada, Takeaway = Ada, Delivery = Ada, Indoor Seat = Ada Then Rating = Good*
17. *If Votes range = Rendah, Wifi = Ada, Valet Parking = Tidak Ada, Booking = Tidak Ada, Private Dining = Tidak Ada, Takeaway = Ada, Delivery = Ada, Indoor Seat = Tidak Ada, Price range = Murah Then Rating = Average*

18. *If* Votes range = Rendah, Wifi = Ada, Valet Parking = Tidak Ada, Booking = Tidak Ada, Private Dining = Tidak Ada, Takeaway = Ada, Delivery = Ada, Indoor Seat = Tidak Ada, Price range = Sedang *Then* Rating = Good
19. *If* Votes range = Rendah, Wifi = Ada, Valet Parking = Tidak Ada, Booking = Tidak Ada, Private Dining = Tidak Ada, Takeaway = Ada, Delivery = Tidak Ada *Then* Rating = Good
20. *If* Votes range = Rendah, Wifi = Ada, Valet Parking = Tidak Ada, Booking = Tidak Ada, Private Dining = Tidak Ada, Takeaway = Tidak Ada, Pray Room = Ada *Then* Rating = Good
21. *If* Votes range = Rendah, Wifi = Ada, Valet Parking = Tidak Ada, Booking = Tidak Ada, Private Dining = Tidak Ada, Takeaway = Tidak Ada, Pray Room = Tidak Ada, Indoor Seat = Ada, Price range = Mahal *Then* Rating = Good
22. *If* Votes range = Rendah, Wifi = Ada, Valet Parking = Tidak Ada, Booking = Tidak Ada, Private Dining = Tidak Ada, Takeaway = Tidak Ada, Pray Room = Tidak Ada, Indoor Seat = Ada, Price range = Sedang *Then* Rating = Average
23. *If* Votes range = Rendah, Wifi = Ada, Valet Parking = Tidak Ada, Booking = Tidak Ada, Private Dining = Tidak Ada, Takeaway = Tidak Ada, Pray Room = Tidak Ada, Indoor Seat = Tidak Ada *Then* Rating = Average
24. *If* Votes range = Rendah, Wifi = Tidak Ada, Private Dining = Ada, Takeaway = Ada *Then* Rating = Very Good
25. *If* Votes range = Rendah, Wifi = Tidak Ada, Private Dining = Ada, Takeaway = Tidak Ada *Then* Rating = Good
26. *If* Votes range = Rendah, Wifi = Tidak Ada, Private Dining = Tidak Ada, Smoking Area = Ada, Outdoor Seat = Ada, Booking = Ada *Then* Rating = Average
27. *If* Votes range = Rendah, Wifi = Tidak Ada, Private Dining = Tidak Ada, Smoking Area = Ada, Outdoor Seat = Ada, Booking = Tidak Ada, Indoor Seat = Ada, Pray Room = Ada *Then* Rating = Average
28. *If* Votes range = Rendah, Wifi = Tidak Ada, Private Dining = Tidak Ada, Smoking Area = Ada, Outdoor Seat = Ada, Booking = Tidak Ada, Indoor Seat = Ada, Pray Room = Tidak Ada *Then* Rating = Good
29. *If* Votes range = Rendah, Wifi = Tidak Ada, Private Dining = Tidak Ada, Smoking Area = Ada, Outdoor Seat = Ada, Booking = Tidak Ada, Indoor Seat = Tidak Ada, Price range = Murah *Then* Rating = Average
30. *If* Votes range = Rendah, Wifi = Tidak Ada, Private Dining = Tidak Ada, Smoking Area = Ada, Outdoor Seat = Ada, Booking = Tidak Ada, Indoor Seat = Tidak Ada, Price range = Sedang *Then* Rating = Good
31. *If* Votes range = Rendah, Wifi = Tidak Ada, Private Dining = Tidak Ada, Smoking Area = Ada, Outdoor Seat = Tidak Ada *Then* Rating = Good
32. *If* Votes range = Rendah, Wifi = Tidak Ada, Private Dining = Tidak Ada, Smoking Area = Tidak Ada, Booking = Ada, Takeaway = Ada, Price range = Mahal *Then* Rating = Average
33. *If* Votes range = Rendah, Wifi = Tidak Ada, Private Dining = Tidak Ada, Smoking Area = Tidak Ada, Booking = Ada, Takeaway = Ada, Price range = Sedang, Indoor Seat = Ada *Then* Rating Good

34. *If* Votes range = Rendah, Wifi = Tidak Ada, Private Dining = Tidak Ada, Smoking Area = Tidak Ada, Booking = Ada, Takeaway = Ada, Price range = Sedang, Indoor Seat = Tidak Ada *Then* Average
35. *If* Votes range = Rendah, Wifi = Tidak Ada, Private Dining = Tidak Ada, Smoking Area = Tidak Ada, Booking = Ada, Takeaway = Tidak Ada, Delivery = Ada *Then* Rating = Very Good
36. *If* Votes range = Rendah, Wifi = Tidak Ada, Private Dining = Tidak Ada, Smoking Area = Tidak Ada, Booking = Ada, Takeaway = Tidak Ada, Delivery = Tidak Ada *Then* Rating = Good
37. *If* Votes range = Rendah, Wifi = Tidak Ada, Private Dining = Tidak Ada, Smoking Area = Tidak Ada, Booking = Tidak Ada *Then* Rating = Average
38. *If* Votes range = Sangat Tinggi, Booking = Ada, Takeaway = Ada, Outdoor Seat = Ada, Private Dining = Ada *Then* Rating = Excellent
39. *If* Votes range = Sangat Tinggi, Booking = Ada, Takeaway = Ada, Outdoor Seat = Ada, Private Dining = Tidak Ada *Then* Rating = Very Good
40. *If* Votes range = Sangat Tinggi, Booking = Ada, Takeaway = Ada, Outdoor Seat = Tidak Ada *Then* Rating = Excellent
41. *If* Votes range = Sangat Tinggi, Booking = Ada, Takeaway = Tidak Ada, Valet Parking = Ada *Then* Rating = Excellent
42. *If* Votes range = Sangat Tinggi, Booking = Ada, Takeaway = Tidak Ada, Valet Parking = Tidak Ada, Smoking Area = Ada *Then* Rating = Very Good
43. *If* Votes range = Sangat Tinggi, Booking = Ada, Takeaway = Tidak Ada, Valet Parking = Tidak Ada, Smoking Area = Tidak Ada *Then* Rating = Excellent
44. *If* Votes range = Sangat Tinggi, Booking = Tidak Ada *Then* Rating = Very Good
45. *If* Votes range = Sedang, Private Dining = Ada, Outdoor Seat = Ada *Then* Rating Very Good
46. *If* Votes range = Sedang, Private Dining = Ada, Outdoor Seat = Tidak Ada, Booking = Ada *Then* Rating = Excellent
47. *If* Votes range = Sedang, Private Dining = Ada, Outdoor Seat = Tidak Ada, Booking = Tidak Ada *Then* Rating = Very Good
48. *If* Votes range = Sedang, Private Dining = Tidak Ada, Takeaway = Ada, Delivery = Ada, Price range = Mahal *Then* Rating = Very Good
49. *If* Votes range = Sedang, Private Dining = Tidak Ada, Takeaway = Ada, Delivery = Ada, Price range = Murah *Then* Rating = Very Good
50. *If* Votes range = Sedang, Private Dining = Tidak Ada, Takeaway = Ada, Delivery = Ada, Price range = Sedang *Then* Rating = Average
51. *If* Votes range = Sedang, Private Dining = Tidak Ada, Takeaway = Ada, Delivery = Tidak Ada *Then* Rating = Very Good
52. *If* Votes range = Sedang, Private Dining = Tidak Ada, Takeaway = Tidak Ada, Wifi = Ada, Booking = Ada *Then* Rating = Very Good
53. *If* Votes range = Sedang, Private Dining = Tidak Ada, Takeaway = Tidak Ada, Wifi = Ada, Booking = Tidak Ada *Then* Rating = Good
54. *If* Votes range = Sedang, Private Dining = Tidak Ada, Takeaway = Tidak Ada, Wifi = Tidak Ada *Then* Rating = Very Good
55. *If* Votes range = Tinggi, Valet Parking = Ada, Private Dining = Ada *Then* Rating = Excellent

56. *If Votes range = Tinggi, Valet Parking = Ada, Private Dining = Tidak Ada Then Rating = Very Good*
57. *If Votes range = Tinggi, Valet Parking = Tidak Ada Then Rating = Very Good*

Pengujian dilakukan dengan validasi silang. Salah satu jenis validasi silang adalah *ten-fold cross validation*. Hasil dari *ten-fold cross validation* dapat dilihat pada [Gambar 6](#).

Berdasarkan pengujian dengan menggunakan metode *Ten-Fold Cross Validation* menghasilkan nilai akurasi sebesar 86,24 % dan *Kappa* sebesar 0,790. Dan analisis hasil pengujian dilakukan dengan melakukan perhitungan secara manual dengan *confusion matrix*. Perhitungan menggunakan model *confusion matrix*. Hasil dari perhitungan *confusion matrix* dapat dilihat pada [Tabel 4](#).

PerformanceVector

```

PerformanceVector:
accuracy: 86.24% +/- 2.76% (micro average: 86.24%)
ConfusionMatrix:
True:   Poor   Average Good   Very Good   Excellent
Poor:   10     0     0     0     0
Average: 3     338   31     2     0
Good:   0     44    364   22     1
Very Good: 1     1     14    128    11
Excellent: 0     0     0     8     25
kappa: 0.790 +/- 0.044 (micro average: 0.790)
ConfusionMatrix:
True:   Poor   Average Good   Very Good   Excellent
Poor:   10     0     0     0     0
Average: 3     338   31     2     0
Good:   0     44    364   22     1
Very Good: 1     1     14    128    11
Excellent: 0     0     0     8     25

```

Gambar 6. Hasil Validasi Ten-fold Cross Validation

Tabel 4. Confusion Matrix

	True Poor	True Average	True Good	True Very Good	True Excellent
Pred. Poor	10	0	0	0	0
Pred. Average	3	338	31	2	0
Pred. Good	0	44	364	22	1
Pred. Very Good	1	1	14	128	11
Pred. Excellent	0	0	0	8	25

Nilai akurasi dapat di peroleh dari,

$$Accuracy = \left(\left(\frac{10 + 338 + 364 + 128 + 25}{1003} \right) * 100\% \right) = 86,24\%$$

Sedangkan nilai *Kappa* adalah, *Kappa* = 0,790

Dari perhitungan tersebut dapat disimpulkan bahwa hasil dari perhitungan yang dilakukan secara manual bahwa nilai akurasi dan *Kappa* tersebut sesuai dengan hasil perhitungan yang ditunjukkan pada [Gambar 6](#). Berdasarkan pengujian dan

analisa hasil pengujian yang dilakukan, dengan tingkat akurasi sebesar 86,24 % menunjukkan nilai yang baik dan hampir akurat, sedangkan *Kappa* sebesar 0,790 dapat diinterpretasikan dalam kategori baik atau *substantial agreement*.

4. KESIMPULAN

Berdasarkan penelitian yang telah dilakukan, tujuan dari penelitian ini untuk memberikan rekomendasi pemilihan restoran berdasarkan rating yang sesuai dalam aplikasi Zomato. Dalam pengujiannya menggunakan 1003 sampel data dan menggunakan algoritma C4.5 untuk menentukan restoran yang sesuai berdasarkan *rating online*. Data yang telah dikumpulkan lalu dilakukan *pre-processing* dengan melakukan *cleaning* data dan transformasi data. Dan dari penelitian ini akan dihasilkan 5 klasifikasi berdasarkan dari *rating* yaitu *Rating Poor*, *Rating Average*, *Rating Good*, *Rating Very Good*, dan *Rating Excellent*. Hasil perhitungan *confusion matrix* dan *Ten-Fold Cross Validation* dihasilkan akurasi yang baik sebesar 86,24%, *Kappa* 0,790 dan menghasilkan 57 aturan atau *rule model* yang dapat membantu pengguna dalam pengambilan keputusan.

Dari hasil penerapan algoritma C4.5 dalam rekomendasi pemilihan restoran berdasarkan rating dapat disimpulkan bahwa rating yang paling dominan dan sesuai untuk direkomendasi adalah *Rating Good*. Dan dapat diimplementasikan sebagai rekomendasi dalam pemilihan restoran. Diharapkan dengan adanya penelitian ini dapat menjadi studi lanjutan untuk pembuatan aplikasi atau dengan melakukan pengujian menggunakan algoritma lain.

REFERENSI

- [1] S. S. Pandean and S. Hansun, "Aplikasi WEB untuk Rekomendasi Restoran Menggunakan Weighted Product," *Jurnal Teknologi Informasi dan Ilmu Komputer*, vol. 5, no. 1, pp. 87–95, 2018, doi: 10.25126/jtiik.201851626
- [2] B. A. Tama, "Data Mining for Predicting Customer Satisfaction in Fast-Food Restaurant," *Journal of Theoretical and Applied Information Technology*, vol. 75, no. 1, pp. 3–7, 2015
- [3] T. Gayathri, "Data mining of restaurant review using WEKA," *International Journal of Engineering and Techniques*, vol. 4, no. 3, pp. 642–645, June 2018
- [4] R. A. Nugroho, "Prototipe Sistem Rekomendasi Menu Makanan dengan Pendekatan Contextual Model dan Multi-Criteria Decision Making," *MediaTeknika Jurnal Teknologi*, vol. 10, no. 2, pp. 111–121, 2015
- [5] A. U. Martliong and N. M. S. Iswari, "Rancang Bangun Sistem Rekomendasi Restoran Menggunakan Metode AHP dan VIKOR pada Platform LINE," *Ultima Computing: Jurnal Sistem Komputer*, vol. 10, no. 1, pp. 27–33, 2018, doi: 10.31937/sk.v10i1.847
- [6] V. Wirawan, S. Hansun, and H. T. Nugroho, "Implementasi Algoritma Squeezer dan Term Frequency Ranking dalam Pembangunan Sistem Rekomendasi Tempat Makan," *Ultima Computing: Jurnal Sistem Komputer*, vol. 6, no. 1, pp. 25–32, 2014
- [7] E. P. Cynthia and E. Ismanto, "Metode Decision Tree Algoritma C4.5 Dalam Mengklasifikasi Data Penjualan Bisnis Gerai Makanan Cepat Saji," *Jurasik: Jurnal Riset Sistem Informasi dan Teknik Informatika*, vol. 3, no. July, pp. 1–13, 2018, doi: 10.30645/jurasik.v3i0.60
- [8] X. Ramirez-Garcia and M. Garcia-Valdez, "Post-Filtering for a Restaurant Context-Aware Recommender System," *Studies in Computational Intelligence*, vol. 551, pp. 695–706, 2014, doi: 10.1007/978-3-319-05170-3_49
- [9] H. Abdool, A. Pooransingh and Y. Li, "Recommend My Dish: A multi-sensory food recommender," *2015 IEEE Pacific Rim Conference on Communications, Computers and Signal Processing (PACRIM)*, 2015, pp. 240–245, doi: 10.1109/PACRIM.2015.7334841

- [10] A. A. P. Devi and D. B. Tonara, "Rancang Bangun Recommender System dengan Menggunakan Metode Collaborative Filtering untuk Studi Kasus Tempat Kuliner di Surabaya," *JUISI: Jurnal Informatika dan Sistem Informasi*, vol. 01, no. 02, pp. 102–112, Agustus 2015
- [11] R. A. Nugroho and R. Ferdiana, "Teknik Pemberian Rekomendasi Menu Makanan dengan Pendekatan Contextual Model dan Multi - Criteria Decision Making," *The 6th Conference on Information Technology and Electrical Engineering (CITEE 2014)*, Yogyakarta, Indonesia, 2014, pp. 88–94
- [12] S. Al-janabi and R. Janicki, "A density-based data cleaning approach for deduplication with data consistency and accuracy," *2016 SAI Computing Conference (SAI)*, London, UK, 2016, pp. 492-501, doi: 10.1109/SAI.2016.7556026
- [13] D. Fitriyah, N. H. Praptono, A. N. Hidayanto, and A. M. Arymurthy, "Feature Exploration for Prediction of Potential Tuna Fishing Zones," *International Journal of Information and Electronics Engineering*, vol. 5, no. 4, pp. 270–274, 2015.
- [14] M. Sadikin and F. Alfiandi, "Comparative Study of Classification Method on Customer Candidate Data to Predict its Potential Risk," *International Journal of Electrical and Computer Engineering*, vol. 8, no. 6, pp. 4763–4771, 2018, doi: 10.11591/ijece.v8i6.pp4763-4771
- [15] N. Azwanti, "Analisa Algoritma C4.5 Untuk Memprediksi Penjualan Motor Pada PT. Capella Dinamik Nusantara Cabang Muka Kuning," *Informatika Mulawarman: Jurnal Ilmiah Ilmu Komputer*, vol. 13, no. 1, pp. 33–38, 2018, doi: 10.30872/jim.v13i1.629
- [16] Y. I. Kurniawan, "Perbandingan Algoritma Naive Bayes dan C.45 dalam Klasifikasi Data Mining," *Jurnal Teknologi Informatika dan Ilmu Komputer*, vol. 5, no. 4, pp. 455–464, 2018, doi: 10.25126/jtiik.201854803
- [17] R. P. S. Putri and I. Waspada, "Penerapan Algoritma C4.5 pada Aplikasi Prediksi Kelulusan Mahasiswa Prodi Informatika," *Khazanah Informatika: Jurnal Ilmu Komputer dan Informatika*, vol. 4, no. 1, pp. 1–7, 2018, doi: 10.23917/khif.v4i1.5975
- [18] A. Jayasimhan, P. Rai, Y. Parekh, and O. Patwardhan, "Recommendation System for Restaurants," *International Journal of Computer Application*, vol. 167, no. 6, pp. 23–25, 2017, doi: 10.5120/ijca2017914312
- [19] A. M. Alfatah, R. Arifudin, and M. A. Muslim, "Implementation of Decision Tree and Dempster Shafer on Expert System for Lung Disease Diagnosis," *Scientific Journal of Informatics*, vol. 5, no. 1, p. 57, 2019, doi: 10.15294/sji.v5i1.13440
- [20] I. El Guabassi, M. A. Achhab, I. Jellouli and B. E. El Mohajir, "Recommender system for ubiquitous learning based on decision tree," *2016 4th IEEE International Colloquium on Information Science and Technology (CiSt)*, 2016, pp. 535-540, doi: 10.1109/CIST.2016.7805107
- [21] M. Chair, Y. N. Nasution, and N. A. Rizki, "Aplikasi Klasifikasi Algoritma C4.5 (Studi Kasus Masa Studi Mahasiswa Fakultas Matematika Dan Ilmu Pengetahuan Alam Universitas Mulawarman Angkatan 2008)," *Informatika Mulawarman: Jurnal Ilmiah Ilmu Komputer*, vol. 12, no. 1, pp. 50–55, 2017, doi: 10.30872/jim.v12i1.223
- [22] M. F. Arifin and D. Fitriyah, "Penerapan Algoritma Klasifikasi C4.5 Dalam Rekomendasi Penerimaan Mitra Penjualan Studi Kasus: PT Atria Artha Persada," *InComTech: Jurnal Telekomunikasi dan Komputer*, vol. 8, no. 2, pp. 87–102, 2018, doi: 10.22441/incomtech.v8i2.2198