

Penerapan Metode *Slope One* dan *K-Nearest Neighbor* Untuk Menentukan Rekomendasi Tempat Makan di Bali Berbasis Android

Lucy Rosalind¹, Leo Willyanto Santoso², Lily Puspa Dewi³
Program Studi Informatika, Fakultas Teknologi Industri, Universitas Kristen Petra
Jl. Siwalankerto 121 – 131 Surabaya 60236
Telp. (031) – 2983455. Fax. (031) - 8417658
lucyrosalind95@gmail.com¹, leow@petra.ac.id², lily@petra.ac.id³

ABSTRAK

Restoran menjadi salah satu alternatif jika dirumah tidak terdapat makanan atau sedang berpergian. Bali sendiri merupakan salah satu kota wisata yang populer karena banyaknya wisatawan yang berdatangan. Banyaknya restoran membuat user bingung untuk memilih restoran yang sesuai dengan keinginannya. Aplikasi rekomendasi ini dibuat untuk membantu agar user tidak bingung dalam memilih restoran yang sesuai keinginannya.

Langkah awal yang dilakukan adalah proses *scraping*, setelah proses *scraping* akan dicari restoran yang tidak memiliki harga yang nantinya akan dihitung prediksi harganya dengan algoritma *Slope One* setelah ini data akan dimasukkan kedalam database. Untuk perhitungan rekomendasi dilakukan di *backend*. Pada saat di Android hanya perlu melakukan pemanggilan API saja.

Berdasarkan hasil pengujian yang sudah dilakukan, program berhasil mengumpulkan data, mengelolah, dan menampilkan data restoran. Untuk proses prediksi harga dengan algoritma *Slope One* memiliki hasil yang sangat baik. Untuk Proses *scraping* data yang ingin diambil sudah sesuai dengan data aslinya. Untuk proses perhitungan *K-Nearest Neighbor* berhasil dilakukan dan memiliki akurasi yang cukup dengan *K* sebesar 35, memiliki akurasi sebesar 59%, dan memiliki data sebanyak 7.890 restoran.

Kata Kunci: rekomendasi, prediksi harga, *Slope One*, *K-Nearest Neighbor*, *web scraping*

ABSTRACT

Restaurants are an alternative if there is no food at home or when you are traveling somewhere. Bali is one of the popular tourism cities because of the many tourists. The number of restaurants makes users confused to choose a restaurant that suits their needs. This recommendation application is made to help users in choosing the restaurant that suits their needs easily.

The initial step taken is the scraping process, after the scraping process the next step is looking for restaurants that do not put prices which will later be calculated with the price prediction with the Slope One algorithm after the data entered into the database. The recommendation calculation is done on the backend. When on Android, it only needs to make API calls.

Based on the results of the tests that have been carried out, the program has succeeded in collecting data, managing, and displaying restaurant data. The price prediction process with the Slope One algorithm has very good results. For the process of scraping the data to be retrieved as the original data. The K-Nearest Neighbor calculation process was successfully carried out and has sufficient accuracy with K of 35, has an accuracy of 59%, and contains data of 7,890 restaurants.

Keywords: *recommendations, prediction price, Slope One, K-Nearest Neighbor, Web Scraping.*

1. PENDAHULUAN

Makanan merupakan hal yang sangat penting bagi manusia. Restoran menjadi pilihan banyak orang karena biasanya tidak sempat memasak makanan di rumah, untuk acara spesial tertentu, dan orang yang sedang berada jauh dari rumahnya misalnya orang yang sedang berpergian. Banyaknya restoran membuat konsumen bingung untuk memilih restoran yang sesuai dengan keinginannya. Dengan adanya aplikasi rekomendasi membantu agar konsumen tidak bingung lagi dalam memilih restoran yang sesuai.

Bali sendiri merupakan tempat yang sering dikunjungi oleh wisatawan sehingga membuat kota ini memiliki peringkat pertama sebagai kota wisata terpopuler di Indonesia [9]. Dengan Bali sebagai kota wisata yang populer menjadikan Bali memiliki banyak pilihan restoran yang ada. Banyaknya restoran, membuat konsumen bingung untuk memilih restoran yang sesuai dengan keinginannya, aplikasi rekomendasi dibuat agar tidak bingung lagi dalam memilih restoran yang sesuai dengan keinginan konsumen. Aplikasi ini juga memiliki fitur *offline* yang bisa membantu menemukan data restoran saat tidak berada ditempat yang memiliki koneksi internet yang bagus.

Akan digunakan metode *Slope One* dimana metode ini adalah salah satu algoritma dari metode Collaborative Filtering, *Slope One* memiliki kelebihan yaitu efisien saat melakukan *query* dan tidak memerlukan banyak *requirements* [8]. Selain menggunakan metode *Slope One* untuk menghitung prediksi harga juga akan digunakan *K-Nearest Neighbor* sebagai algoritma untuk mencari rekomendasi. Kelebihan dari metode *K-Nearest Neighbor* adalah memiliki konsistensi yang kuat sedangkan kelemahannya adalah tidak bisa memproses data yang memiliki *missing value* [6].

2. TINJAUAN PUSTAKA

2.1 Collaborative Filtering

Collaborative Filtering adalah algoritma yang bisa membantu untuk membuat sistem rekomendasi. Cara menentukan rekomendasi nya adalah mengolah data yang ada dan me disukai konsumen. Dari hasil perhitungan ini akan menunjukkan produk mana yang mungkin disukai oleh konsumen [10].

2.2 K-Nearest Neighbor

Algoritma *K-Nearest Neighbor* (*K-NN*) adalah algoritma yang melakukan klasifikasi dari suatu data yang diambil k tetangga terdekatnya (*nearest neighbor*), dimana k merupakan banyaknya tetangga terdekat dari sekumpulan data tersebut.

Perhitungan jarak didapatkan dari rumus perhitungan jarak Euclidean adalah sebagai berikut [6]:

$$d(x, y) = \sqrt{(\sum_{i=1}^n (x_i - y_i)^2)} \quad (1)$$

dimana:

x_i = sampel data

y_i = data uji atau data *testing*

i = variable data

$d(x,y)$ = jarak

n = dimensi data

2.3 Algoritma Slope One

Algoritma *Slope One* adalah algoritma yang bisa membantu untuk memprediksi sebuah kejadian dengan cara mencari selisih nilai dari *item-item* yang akan dibandingkan.

Algoritma *Slope One* memiliki perhitungan dari *weight item* yang nantinya akan dibandingkan. Dasar dari perhitungan algoritma *Slope One* ini adalah $y = mx + b$, dengan $m = 1$ sebagai *gradient* sehingga menjadi $b = y - x$. Rumus untuk algoritma *Slope One* adalah sebagai berikut:

$$dev_{j,i} = \frac{\sum(x) \cdot u_j - u_i}{card(S_{j,i}(x))} \quad (2)$$

Dimana:

$dev_{j,i}$ = rata-rata selisih *rating* item j dan i

u_j = *rating* item j

u_i = *rating* item i

$card(S_{j,i}(x))$ = banyaknya elemen yang dibandingkan

Apabila rata-rata selisih *rating item* j dan i sudah diperoleh, maka dilakukan untuk perhitungan terhadap j

$$PSI(u)_j = dev_{j,i} + u_j \quad (3)$$

Dimana:

$PSI(u)_j$ = nilai rekomendasi untuk item j

Algoritma *Slope One* melakukan perhitungan dari rata-rata setiap item yang kemudian akan dijumlahkan dengan *value* yang akan menghasilkan nilai yang memberikan hasil dari rekomendasinya [5].

2.4 Web scraping

Web scraping adalah proses ekstraksi data dari *website* yang hanya diambil bagian-bagian tertentu [11]. *web scraping* dilakukan dengan memberikan URL untuk diambil data-data yang diinginkan.

2.5 BeautifulSoup

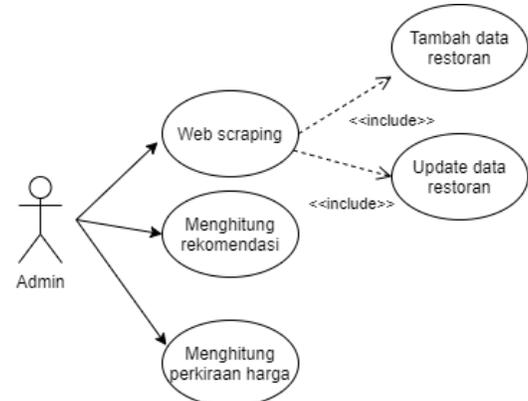
Beautiful Soup adalah *library* yang ada di Python yang biasanya digunakan untuk melakukan *web scraping* [13]. Cara untuk melakukan instalasi adalah dengan membuka *command prompt* dan mengetikkan `pip install bs4` pada *command prompt*. Untuk

penggunaanya hanya tinggal panggil saja *library* ini dengan cara `from bs4 import BeautifulSoup`.

3. ANALISIS DAN DESAIN SISTEM

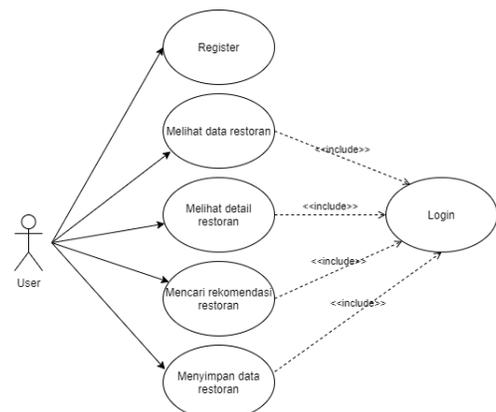
3.1 Use Case Diagram

Gambar 1 merupakan UCD atau yang biasa disebut *use case diagram* yang berguna untuk mengetahui apa saja yang bisa dilakukan oleh *actor* atau *user*. Pada aplikasi ini *user* bisa melakukan *register*, *login* aplikasi, melihat data restoran, melihat data detail dari restoran, mencari rekomendasi restoran yang sesuai, melakukan penyimpanan untuk data restoran yang ingin disimpan. Untuk data restoran sendiri bisa dilihat secara *offline*.



Gambar 1. Use case diagram user

Gambar 2 merupakan *use case diagram* untuk admin. Yang bisa dilakukan admin pada aplikasi ini adalah melakukan proses *web scraping* terlebih dahulu agar bisa melakukan penambahan data restoran, dan melakukan *update* data untuk restoran. Selain itu Admin juga melakukan perhitungan untuk perkiraan harga yang tidak dimiliki oleh restoran supaya bisa mengetahui kisaran harga untuk restoran tersebut. Admin juga melakukan perhitungan untuk mencari rekomendasi restoran yang diinginkan *user*.

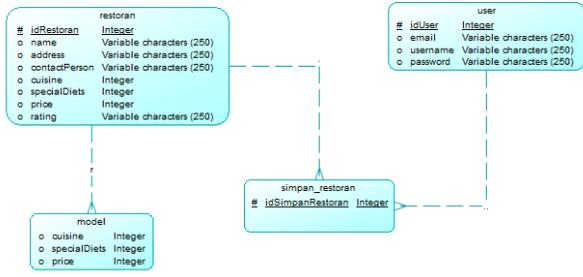


Gambar 2. Use case diagram admin

3.2 Entity Relationship Instagram

Gambar 3 merupakan ERD yang memiliki dua tabel yang terdiri dari tabel restoran dan tabel *user*. Untuk tabel restoran berisikan nama restoran, alamat restoran, nomor telepon restoran, jenis

makanan, dan tipe makanan. Untuk tabel *user* berisikan email, username, dan password. Sedangkan tabel *save* restoran berisikan id, fungsi tabel ini untuk menyimpan data restoran yang disimpan oleh *user*. Juga terdapat tabel model dimana berisikan perhitungan untuk rekomendasi tempat makan.

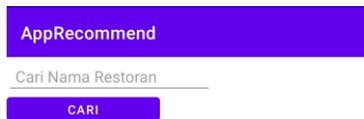


Gambar 3. Entity relationship diagram

4. PENGUJIAN SISTEM

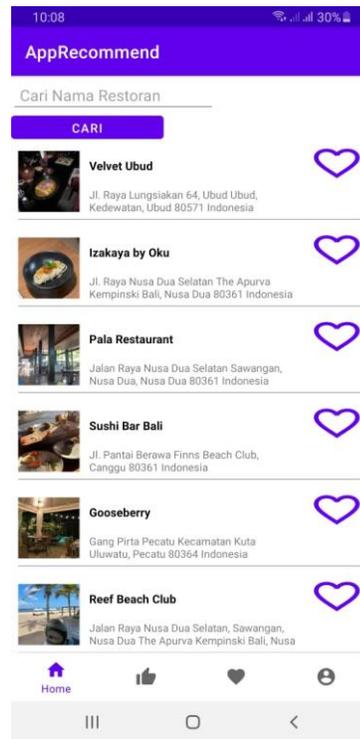
4.1 Pengujian keberhasilan *Scraping*

Pada Gambar 4 merupakan tampilan jika *scraping* data belum dijalankan. Untuk proses *scraping* sendiri menggunakan *cron jobs* yang datanya akan diperbarui setiap awal bulan. Hasil dari *scraping* akan dimasukkan kedalam *database*.



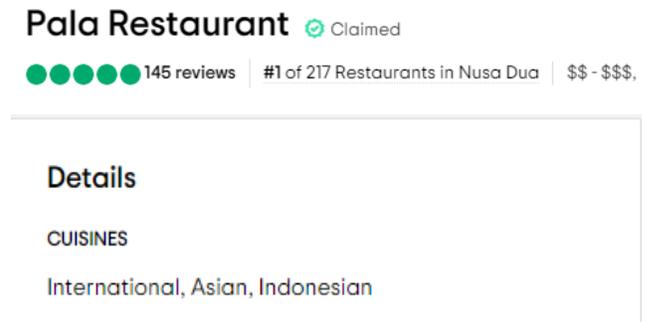
Gambar 4. Halaman sebelum *scraping* dijalankan

Pada Gambar 5 merupakan hasil jika *scraping* sudah dijalankan, yang nantinya datanya akan dipakai pada saat perhitungan rekomendasi.



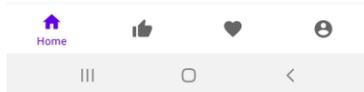
Gambar 5. Setelah *scraping* dilakukan

Pada Gambar 6 merupakan tampilan detail dari situs *Tripadvisor* yang terdiri dari nama restoran, alamat, harga, jenis masakan, tipe makanan, dan nomor telepon



Gambar 6. Informasi restoran *tripadvisor*

Pada Gambar 7 merupakan tampilan detail restoran yang terdapat di aplikasi. Membuktikan bahwa data yang ditampilkan sesuai dengan situs aslinya. Data yang digunakan adalah nama restoran, alamat restoran, foto restoran, nomor telepon restoran, harga, jenis masakan, dan tipe makanan yang terdapat di restoran. Semua data yang ada berasal dari proses *scraping* yang dilakukan. Jika alamat restoran ditekan maka akan otomatis membuka *maps* untuk menuju alamat restoran.

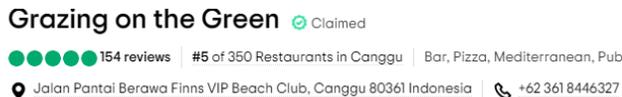




Gambar 7. Detail restoran di aplikasi

4.2 Pengujian Prediksi Harga

Pada Gambar 8 merupakan salah satu contoh restoran yang tidak memiliki harga. Nantinya akan digunakan algoritma *Slope One* untuk mengetahui prediksi harga yang akan dimiliki restoran. Untuk perhitungan prediksi yang dilakukan data yang diperlukan adalah data *rating*, dan data harga.



Gambar 8. Restoran yang belum memiliki harga

Pada Gambar 9 merupakan hasil setelah dilakukan perhitungan untuk prediksi harga. Restoran tersebut sudah memiliki data harga yang berasal dari perhitungan prediksi harga menggunakan algoritma *Slope One*. Untuk harga dibagi menjadi tiga bagian yaitu harga ekonomis, harga sedang, dan harga eksekutif.



Gambar 9. Setelah dilakukan perhitungan prediksi harga

Pada Gambar 10 merupakan hasil dari perhitungan algoritma *Slope One* pada salah satu restoran yang sudah memiliki harga. Bahwa hasil perhitungan untuk estimasi harga nya adalah 2.0 dimana sesuai dengan Gambar 11 yang merupakan harga asli dari situs *Tripadvisor*. Pada situs *Tripadvisor* data harga yang ditampilkan adalah \$\$ - \$\$\$ dimana memiliki arti 2.0. Dapat diartikan bahwa perhitungan prediksi menggunakan algoritma *Slope One* bisa memprediksi harga dengan baik. Untuk data *testing* yang digunakan sebesar 20% dan data *training* sebesar 80%.

id	name	address	price	rating	specialdiet	est
13974	Velvet Ubud	Jl. Raya Lungsiaan 64, Ubud Ubud, Kedewatan, ...	2	4	Vegetarian Friendly, Vegan Options, Gluten Fre...	2.00000

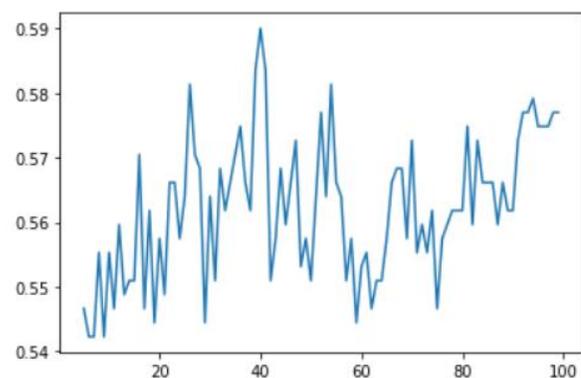
Gambar 10. Perhitungan Prediksi Harga Restoran yang Memiliki Harga



Gambar 11. Detail restoran di tripadvisor

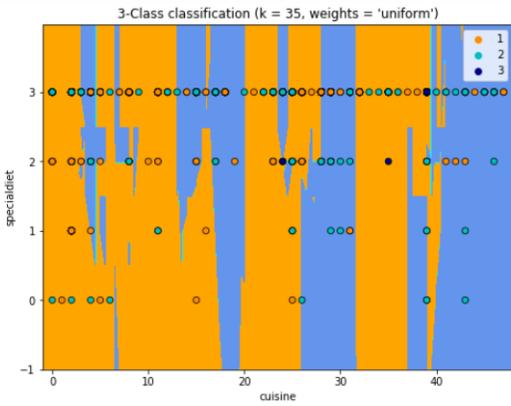
4.3 Perhitungan Rekomendasi

Dimulai dari membaca data, membagi data menjadi dua bagian yaitu data *testing* dan data *training*. Setelah itu dari data *training* dibuat model untuk menghitung hasil dari rekomendasinya. Untuk hasil perhitungan KNN yang telah dibuat menghasilkan akurasi sebesar 59% seperti Gambar 12 yang merupakan grafik hasil perhitungannya dimana sumbu Y adalah nilai akurasi dan sumbu X adalah K. Selanjutnya dari data yang akan di prediksi ada id yang unik, kemudian waktu *running* outputnya akan menampilkan id dan harga, kemudian telusuri dari id itu dapat nama restaurannya apa. setelah itu dimasukkan kedalam API agar bisa dipanggil di android. Untuk data *testing* yang digunakan sebesar 30% dan data *training* yang digunakan sebesar 70%.

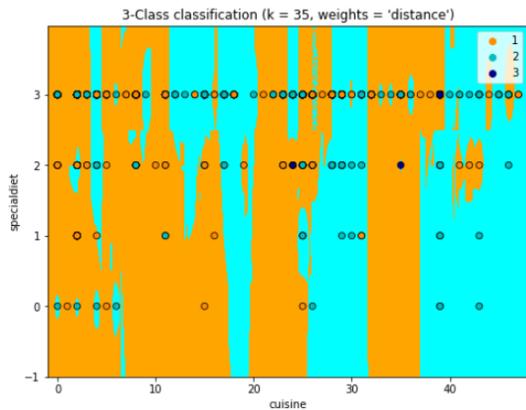


Gambar 12. Grafik hasil rekomendasi

Pada Gambar 13 dan Gambar 14 merupakan penggambaran data yang tersebar berdasarkan *uniform* dan berdasarkan *distance*. Misalnya dapat diasumsikan sebagai pemungutan “suara”, maka *uniform* mendapatkan semua *neighbor* nya seragam. Sedangkan *distance* akan memberi tahu untuk menimbang “suara” berdasarkan klasifikasi jarak pengamatan.

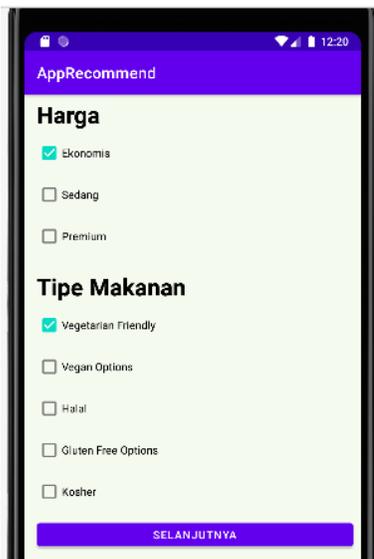


Gambar 13. Penggambaran data tersebar berdasarkan uniform



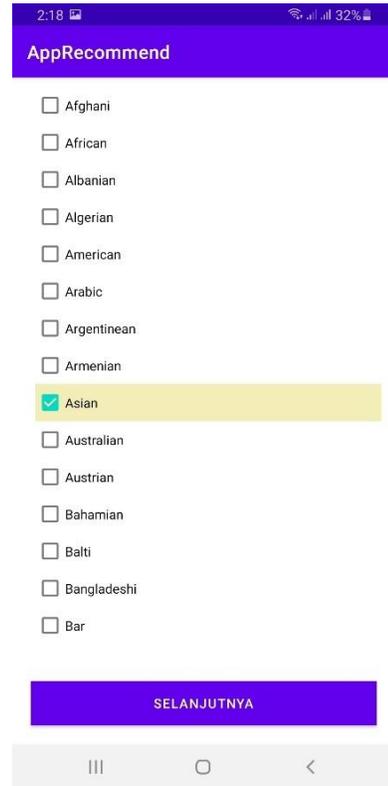
Gambar 14. Penggambaran data tersebar berdasarkan distance

Pada Gambar 15 merupakan halaman *input* untuk jenis masakan. Untuk jenis masakan *user* juga wajib melakukan *input*. Setelah melakukan *input user* menekan *button* selanjutnya agar bisa dilakukan perhitungan rekomendasinya.



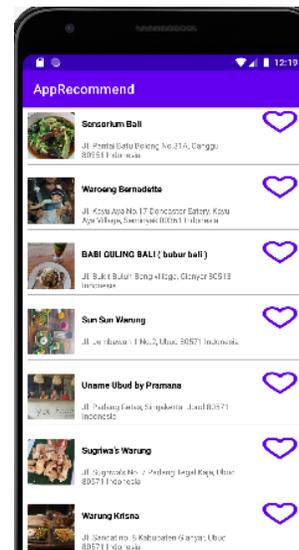
Gambar 15. Halaman input harga dan tipe makanan

Pada gambar 16 merupakan halaman untuk *user* melakukan *input* jenis masakan.



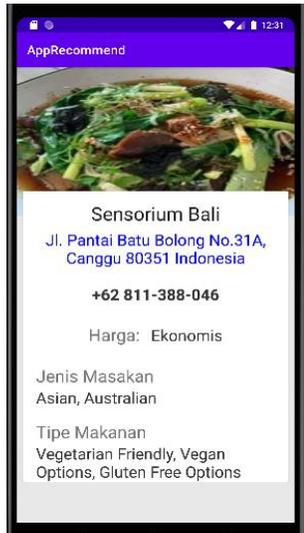
Gambar 16. Halaman input jenis masakan

Pada gambar 17 merupakan halaman dari hasil perhitungan rekomendasi. Hasil rekomendasi didapatkan dari perhitungan yang sebelumnya dibuat setelah itu dilakukan pemanggilan API untuk mengetahui hasil rekomendasinya.



Gambar 17. Hasil perhitungan rekomendasi

Pada Gambar 18 merupakan detail dari restoran yang telah dilakukan perhitungan rekomendasi. Restoran yang didapatkan sesuai dengan *input* dari *user* sebelumnya.



Gambar 18. Detail restoran perhitungan rekomendasi

4.4 Proses Melihat Detail Restoran

Untuk pengujian ini bertujuan untuk membandingkan bahwa data yang diambil dari proses *scrapping* yang dilakukan sesuai dengan data dari situs *Tripadvisor.com*. Gambar 19 merupakan detail dari restoran yang terdapat pada aplikasi. Halaman detail berisikan informasi seperti foto restoran, nama restoran, alamat restoran, nomor telepon restoran, harga makanan yang terdapat di restoran, jenis masakan, dan tipe makanan. Untuk hasil yang dihasilkan sesuai dengan situs *Tripadvisor.com*.



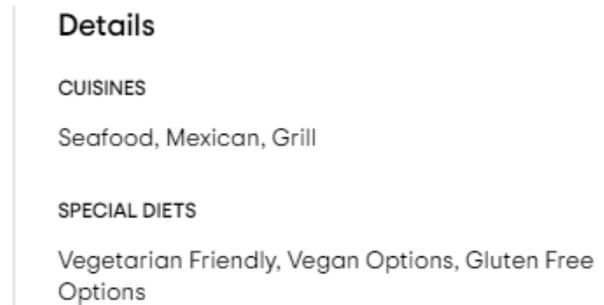
Gambar 19. Halaman detail restoran

Pada Gambar 20 merupakan detail restoran yang terdapat di situs *Tripadvisor.com*. Detail restoran yang digunakan pada situs ini terdiri dari harga, jenis masakan, nomor telepon, dan alamat restoran.



Gambar 20. Detail alamat, harga, dan jenis masakan di *Tripadvisor*

Pada Gambar 21 merupakan detail restoran yang terdapat pada situs *Tripadvisor.com*. Detail restoran yang ada terdiri dari jenis masakan, dan tipe makanan.



Gambar 21. Jenis masakan, dan tipe makanan di *Tripadvisor*

4.5 Proses Menambahkan Data Restoran Menggunakan Fitur Favorit

Untuk pengujian ini dilakukan dengan tujuan apakah fitur untuk menambahkan ke dalam favorit sudah berjalan dengan baik atau belum berjalan dengan baik. Gambar 22 adalah halaman favorit saat sudah terdapat data yang ditambahkan ke dalam favorit. Tampilan informasinya terdiri dari foto, nama restoran, dan alamat restoran.



Gambar 22. Halaman favorit saat terdapat data

5. KESIMPULAN DAN SARAN

Dari hasil pengujian yang dilakukan dapat disimpulkan beberapa hal diantaranya adalah sebagai berikut:

- Berdasarkan pengujian yang dilakukan proses *scraping* berhasil dijalankan.
- Terdapat restoran yang memiliki format struktur datanya berbeda meskipun satu situs, sehingga membutuhkan beberapa proses tambahan untuk melakukan *scraping*.
- Situs Tripadvisor terkadang merubah struktur data yang ada, sehingga program yang telah dibuat dengan baik dan tiba-tiba terdapat *update* dari Tripadvisor mengakibatkan tidak bisa menjalankan proses *scraping*.
- Terdapat beberapa perhitungan rekomendasi jika terdapat salah satu restoran yang sesuai dengan *input* dari *user* dan yang lainnya masih belum sesuai maka tetap dihitung bahwa restoran tersebut masuk dalam rekomendasi.
- Untuk *input* saat melakukan pencarian rekomendasi hanya bisa satu *input* dikarenakan data yang digunakan untuk menghitung rekomendasi berupa huruf yang harus dijadikan angka terlebih dahulu agar bisa dilakukan perhitungan menggunakan KNN.
- Untuk perhitungan yang sesuai dengan rekomendasi mendapatkan rata-rata 70, untuk fitur *offline* mendapatkan rata-rata 94, untuk fitur pencarian restoran mendapatkan rata-rata 68, untuk fitur melihat detail restoran mendapatkan rata-rata 92, dan untuk fitur menambahkan ke favorit mendapatkan rata-rata 92.

Sedangkan saran yang ada dalam penelitian adalah sebagai berikut:

- Memperbaiki beberapa tampilan yang masih kurang sempurna.
- Menambah *dataset* untuk di-*scraping*, agar tidak hanya bisa restoran di Bali untuk perhitungan rekomendasinya.
- Bisa menambahkan algoritma supaya perhitungan rekomendasinya semakin sesuai

6. DAFTAR PUSTAKA

- [1] Adexe Alifian, 2021, June 11. Belajar dasar web scraping bersama python. URI=<https://www.anbidev.com/python-web-scraping/>
- [2] Anggika, L. K. 2018. Pelayanan restoran dalam menciptakan kepuasan pelanggan: studi pengendalian mutu di restoran D'Ayam Crispy Yogyakarta. DOI=<https://doi.org/10.31227/osf.io/tg4ma>
- [3] Ardiansyah, E., Basuki, S., & Wicaksono, G. W. 2020. Aplikasi rekomendasi rumah makan menggunakan algoritma k-nearest neighbour berbasis android (Studi kasus: Kecamatan Banyuwangi). *Jurnal Repositor*, 2(2), 145-154. DOI=<https://doi.org/10.22219/repositor.v2i2.247.g37>
- [4] Artha, I. N. W., & Wikansari, R. 2014. Perilaku konsumen dalam memilih tempat makan dengan pendekatan stimulus pemasaran.
- [5] Dharma. Pratama & Seng. Hansun. 2017. Aplikasi rekomendasi tempat makan menggunakan algoritma slope one pada platform android, IJCCS, pp. 11-20, 2017.
- [6] Dsn Cahya. 2021, March 22. Algoritma k-nearest neighbors (K-NN). URI=<https://cahyadsn.phpindonesia.id/extra/knn.php>
- [7] Firmahsyah, F., & Gantini, T. 2016. Penerapan metode content-based filtering pada sistem rekomendasi kegiatan ekstrakurikuler (Studi Kasus di Sekolah ABC). *JuTISI (Jurnal Teknik Informatika dan Sistem Informasi)*, 2(3). DOI=<https://doi.org/10.28932/jutisi.v2i3.645>
- [8] Handoko, R. I. 2019. Rancang bangun aplikasi masacake untuk rekomendasi dan order online bahan kue di toko bagus memanfaatkan youtube api (Doctoral dissertation, Universitas Komputer Indonesia).
- [9] Idhom Addi. 2021, February 20. Daftar 10 kota wisata Indonesia terpopuler 2019 versi tripadvisor. URI=<https://tirto.id/daftar-10-kota-wisata-indonesia-terpopuler-2019-versi-tripadvisor-epEc>
- [10] Laksana, E. A. 2014. Collaborative filtering dan aplikasinya. *Jurnal Ilmiah Teknologi Infomasi Terapan*, 1(1). DOI=<https://doi.org/10.33197/jitter.vol1.iss1.2014.44>
- [11] Mitra, V., Sujaini, H., & Negara, A. B. P. 2017. Rancang bangun aplikasi web scraping untuk korpus paralel indonesia-inggris dengan metode html dom. *JUSTIN (Jurnal Sistem dan Teknologi Informasi)*, 5(1), 36-41.
- [12] Ratnasari, A. D. 2016. Pengaruh kualitas produk, harga, lokasi, dan kualitas layanan. *Jurnal Pendidikan Tata Niaga*.4(3).
- [13] Tandra, D. J., Noertjahyana, A., & Purbowo, A. N. 2020. Implementasi Web Scraping untuk Pengumpulan Informasi Promo Makanan Menggunakan Klasifikasi Naïve Bayes. *Jurnal Infra*, 8(1), 289-294.