

TESIS

**ALGORITMA PENCARIAN STRING BOYER MOORE DALAM
PENDEKATAN CONTENT BASED FILTERING PADA SISTEM
REKOMENDASI RESEP MASAKAN**



Disusun oleh :

Nama : Didik Irawan
NIM : 19.52.1270
Konsentrasi : Bussness Intelligence

**PROGRAM STUDI S2 TEKNIK INFORMATIKA
PROGRAM PASCASARJANA UNIVERSITAS AMIKOM YOGYAKARTA
YOGYAKARTA
2021**

HALAMAN JUDUL

**ALGORITMA PENCARIAN STRING BOYER MOORE DALAM
PENDEKATAN CONTENT BASED FILTERING PADA SISTEM
REKOMENDASI RESEP MASAKAN**

***BOYER MOORE STRING SEARCH ALGORITHM IN CONTENT BASED
FILTERING APPROACH IN CUISINE RECIPES RECOMMENDATION
SYSTEM***

Diajukan untuk memenuhi salah satu syarat memperoleh derajat Magister



Disusun oleh :

Nama : Didik Irawan
NIM : 19.52.1270
Konsentrasi : Bussness Intelligence

**PROGRAM STUDI S2 TEKNIK INFORMATIKA
PROGRAM PASCASARJANA UNIVERSITAS AMIKOM YOGYAKARTA
YOGYAKARTA
2021**

HALAMAN PENGESAHAN

**ALGORITMA PENCARIAN STRING BOYER MOORE DALAM
PENDEKATAN CONTENT BASED FILTERING PADA SISTEM
REKOMENDASI RESEP MASAKAN**

***BOYER MOORE STRING SEARCH ALGORITHM IN CONTENT BASED
FILTERING APPROACH IN CUISINE RECIPES RECOMMENDATION
SYSTEM***

Dipersiapkan dan Disusun oleh

Didik Irawan

19.52.1270

Telah Diujikan dan Dipertahankan dalam Sidang Ujian Tesis
Program Studi S2 Teknik Informatika
Program Pascasarjana Universitas AMIKOM Yogyakarta
Pada hari Senin, 3 Januari 2022

Tesis ini telah diterima sebagai salah satu persyaratan
Untuk memperoleh gelar Magister Komputer

Yogyakarta, 3 Januari 2022

Rektor

Prof. Dr. M Suyanto, M.M.
NIK.190302001

HALAMAN PERSETUJUAN

**ALGORITMA PENCAIRAN STRING BOYER MOORE DALAM
PENDEKATAN CONTENT BASED FILTERING PADA SISTEM
REKOMENDASI RESEP MASAKAN**

***BOYER MOORE STRING SEARCH ALGORITHM IN CONTENT BASED
FILTERING APPROACH IN CUISINE RECIPES RECOMMENDATION
SYSTEM***

Dipersiapkan dan Disusun oleh

Didik Irawan
19.52.1270

Telah Diujikan dan Dipertahankan dalam Sidang Ujian Tesis
Program Studi S2 Teknik Informatika
Program Pascasarjana Universitas AMIKOM Yogyakarta
Pada hari Senin, 3 Januari 2022

Pembimbing Utama

Prof. Dr. Kusriani, M.Kom.
NIK. 190302106

Anggota Tim Penguji

Prof. Dr. Ema Utami, S.Si., M.Kom.
NIK. 190302037

Pembimbing Pendamping

Muhammad Rudyanto Arief, M.T.
NIK. 190302098

Dr. Andi Sunyoto, M.Kom.
NIK. 190302052

Prof. Dr. Kusriani, M.Kom.
NIK. 190302106

Tesis ini telah diterima sebagai salah satu persyaratan
Untuk memperoleh gelar Magister Komputer

Yogyakarta, 3 Januari 2022

Direktur Program Pascasarjana

Prof. Dr. Kusriani, M.Kom.
NIK. 190302106

HALAMAN PERNYATAAN KEASLIAN TESIS

Yang bertandatangan di bawah ini,

Nama mahasiswa : Didik Irawan

NIM : 19.52.1270

Konsentrasi : Business Intelligence

Menyatakan bahwa Tesis dengan judul :

Algoritma Pencarian String Boyer Moore Dalam Pendekatan Content Based Filtering Pada Sistem Rekomendasi Resep Masakan

Dosen Pembimbing Utama : Prof. Dr. Kusriani, M.Kom

Dosen Pembimbing Pendamping : Muhammad Rudyanto Arief, M.T.

1. Karya tulis ini adalah benar-benar ASLI dan BELUM PERNAH diajukan untuk mendapatkan gelar akademik, baik di Universitas AMIKOM Yogyakarta maupun di Perguruan Tinggi lainnya.
2. Karya tulis ini merupakan gagasan, rumusan dan penelitian SAYA sendiri tanpa bantuan pihak lain kecuali arahan dari Tim Dosen Pembimbing.
3. Dalam karya tulis ini tidak terdapat karya atau pendapat orang lain, kecuali secara tertulis dengan jelas dicantumkan sebagai acuan dalam naskah dengan disebutkan nama pengarang dan disebutkan dalam Daftar Pustaka pada karya tulis ini.
4. Perangkat lunak yang digunakan dalam penelitian ini sepenuhnya menjadi tanggung jawab SAYA, bukan tanggung jawab Universitas AMIKOM Yogyakarta.
5. Pernyataan ini SAYA buat dengan sesungguhnya, apabila di kemudian hari terdapat penyimpangan dan ketidakbenaran dalam pernyataan ini, maka SAYA bersedia menerima SANKSI AKADEMIK dengan pencabutan gelar yang sudah diperoleh, serta sanksi lainnya yang sesuai dengan norma yang berlaku di Perguruan Tinggi.

Yogyakarta, 3 Januari 2022

Yang Menyatakan,



Didik Irawan

HALAMAN PERSEMBAHAN

Puji Syukur penulis panjatkan kepada Allah SWT, yang telah memberikan kesehatan, rahmat dan hidayah, sehingga penulis masih diberikan kesempatan untuk menyelesaikan tesis ini, sebagai salah satu syarat untuk mendapatkan gelar magister. Walaupun jauh dari kata sempurna, namun penulis bangga telah mencapai pada titik ini, yang akhirnya tesis ini bisa selesai diwaktu yang tepat.

Tesis ini penulis persembahkan untuk Almarhumah Ibu, Bapak, Mama Mertua, Bapak Mertua, Istri tercinta, Adik dan Saudara-saudara yang tidak dapat penulis sebutkan satu persatu. Berkat merekalah penulis dapat menyelesaikan karya ilmiah ini dan mendapatkan pencapaian sejauh ini. Terimakasih yang sangat besar khususnya kepada istri tercinta yang sangat sabar dan selalu mendorong untuk terus maju. Doa terbaik untuk mereka dan semoga Allah Subhanahu Wata'ala membalas segala kebaikan mereka. Aamiin

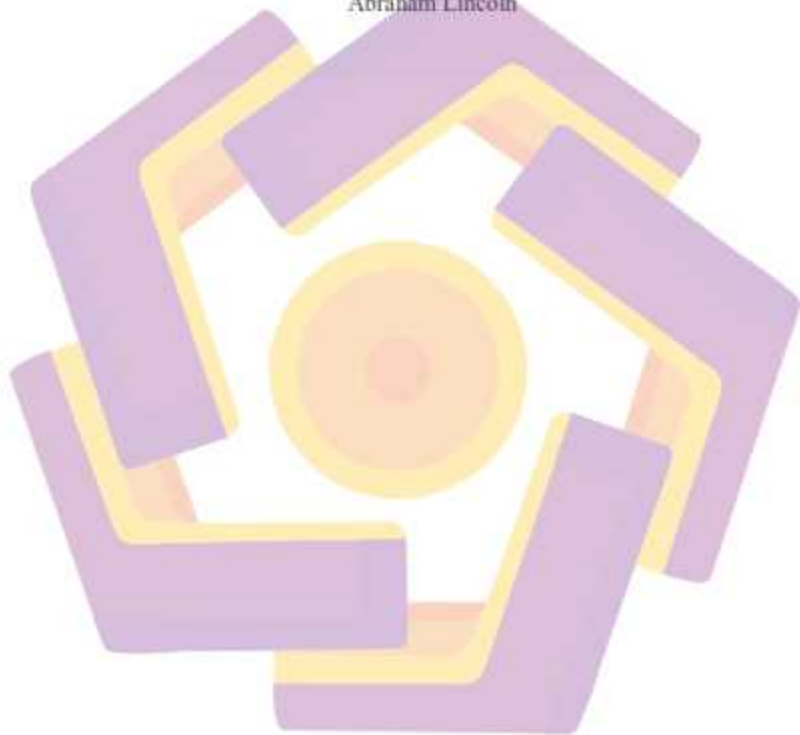
HALAMAN MOTTO

“Jangan menilai saya dari kesuksesan, tetapi nilai saya dari seberapa sering saya jatuh dan berhasil bangkit kembali.”

Nelson Mandela

“Sukses berjalan dari satu kegagalan ke kegagalan yang lain, tanpa kita kehilangan semangat.”

Abraham Lincoln



KATA PENGANTAR

Segala puji syukur kehadiran Allah Subhanahu Wa Ta'ala yang telah melimpahkan anugerah dan karunia-Nya sehingga penulis dapat menyelesaikan Tesis yang berjudul "Algoritma Pencarian String *Boyer-Moore* Dalam Pendekatan *Content Based Filtering* Pada Sistem Rekomendasi Resep Masakan". Tesis ini disusun sebagai salah satu syarat menyelesaikan studi magister di Program Studi Magister Teknik Informatika Universitas AMIKOM Yogyakarta. Dengan ini, penulis menyampaikan penghormatan dan terima kasih kepada pihak-pihak yang telah memberikan bantuan dan dukungan baik berupa moral maupun material secara langsung maupun tidak langsung antara lain kepada:

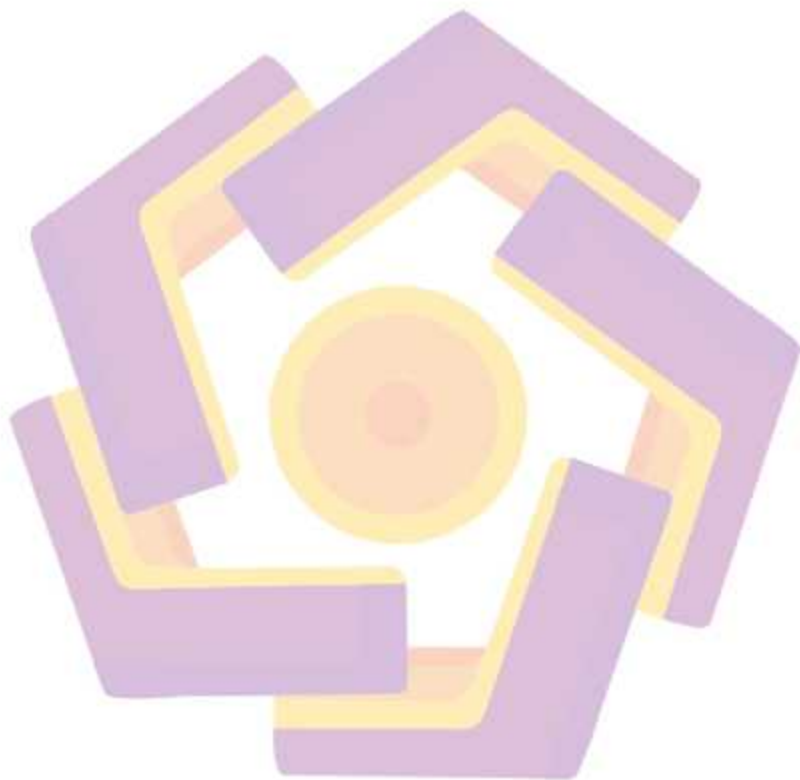
1. Prof. Dr. M. Suyanto, M.M. selaku Rektor Universitas Amikom Yogyakarta.
2. Ibu Prof. Dr. Kusri, M.Kom selaku Ketua Program Studi Magister Teknik Informatika dan selaku Dosen Pembimbing Utama Tesis yang telah meluangkan waktu, tenaga dan pikiran dalam memberikan bimbingan, pengarahan, dan ilmu pengetahuan.
3. Bapak Muhammad Rudyanto Arief, M.T. selaku Dosen Pembimbing Pendamping Tesis yang telah meluangkan waktu, tenaga dan pikiran dalam memberikan bimbingan, pengarahan, dan ilmu pengetahuan.
4. Orang tua dan istri tercinta yang selalu memberikan dukungan baik moral dan materiil bagi penulis.
5. Saudara-saudara tercinta yang juga selalu memberikan dukungan moral bagi penulis.
6. Teman-teman MTI angkatan 2019 yang selalu memotivasi, mengingatkan, memberi masukan, dan selalu memberi suntikan semangat kepada penulis.
7. Semua pihak yang tidak dapat disebutkan satu persatu, yang telah banyak memberikan berbagai macam bantuan dalam penyusunan Tesis ini.

Akhir kata, penulis berharap Tesis ini dapat memberikan manfaat kepada pembaca mengenai sistem rekomendasi berbasis konten. Penulis menyadari bahwa tesis ini masih jauh dari kesempurnaan dan memiliki banyak kekurangan. Oleh

karena itu, dengan kerendahan hati penulis mengharapkan masukan dan saran yang membangun untuk perbaikan ke depan.

Yogyakarta, 3 Januari 2022

Penulis



DAFTAR ISI

HALAMAN JUDUL	ii
HALAMAN PENGESAHAN	iii
HALAMAN PERSETUJUAN	iv
HALAMAN PERNYATAAN KEASLIAN TESIS	v
HALAMAN PERSEMBAHAN	vi
HALAMAN MOTTO	vii
KATA PENGANTAR	viii
DAFTAR ISI	x
DAFTAR TABEL	xii
DAFTAR GAMBAR	xiii
INTISARI	xv
ABSTRACT	xvi
BAB I PENDAHULUAN	1
1.1. Latar Belakang Masalah	1
1.2. Rumusan Masalah	3
1.3. Batasan Masalah	4
1.4. Tujuan Penelitian	4
1.5. Manfaat Penelitian	5
1.6. Hipotesis	5
BAB II TINJAUAN PUSTAKA	7
2.1. Tinjauan Pustaka	7
2.2. Keaslian Penelitian	10
2.3. Landasan Teori	16
2.3.1. Resep Masakan	16
2.3.2. Sistem Rekomendasi	16
2.3.3. <i>Text Preprocessing</i>	18
2.3.4. Algoritma Pencarian String <i>Boyer-Moore</i>	21
2.3.5. <i>Content Based Filtering</i>	23
2.3.6. Algoritma <i>Term Frequency-Invers Document Frequency</i>	24
2.3.7. Algoritma <i>Cosine Similarity</i>	25

2.3.8. <i>Recall, Precision dan F1-Score</i>	26
2.3.9. <i>Mean Average Precision</i>	28
BAB III METODE PENELITIAN	29
3.1. Jenis, Sifat dan Pendekatan Penelitian.....	29
3.2. Metode Pengumpulan Data.....	29
3.3. Analisa Data.....	30
3.4. Pengujian.....	31
3.5. Alur Penelitian.....	32
BAB IV HASIL PENELITIAN DAN PEMBAHASAN.....	34
4.1. Pengumpulan Data.....	34
4.2. Skenario Model Rekomendasi.....	35
4.3. Preprocessing.....	41
4.4. Model Rekomendasi.....	44
4.5. Hasil Model Rekomendasi.....	51
4.5.1. Skenario 1.....	51
4.5.2. Skenario 2.....	52
4.5.3. Skenario 3.....	54
4.5.4. Skenario 4.....	55
4.5.5. Skenario 5.....	56
4.6. Evaluasi Model Rekomendasi.....	58
4.6.1. <i>Recall, Precision dan F1-score</i>	59
4.6.2. <i>Mean Average Precision</i>	60
4.6.3. <i>Average Weight Ingredient Contained</i>	64
BAB V PENUTUP	67
5.1. Kesimpulan.....	67
5.2. Saran	68
DAFTAR PUSTAKA	69
LAMPIRAN.....	71

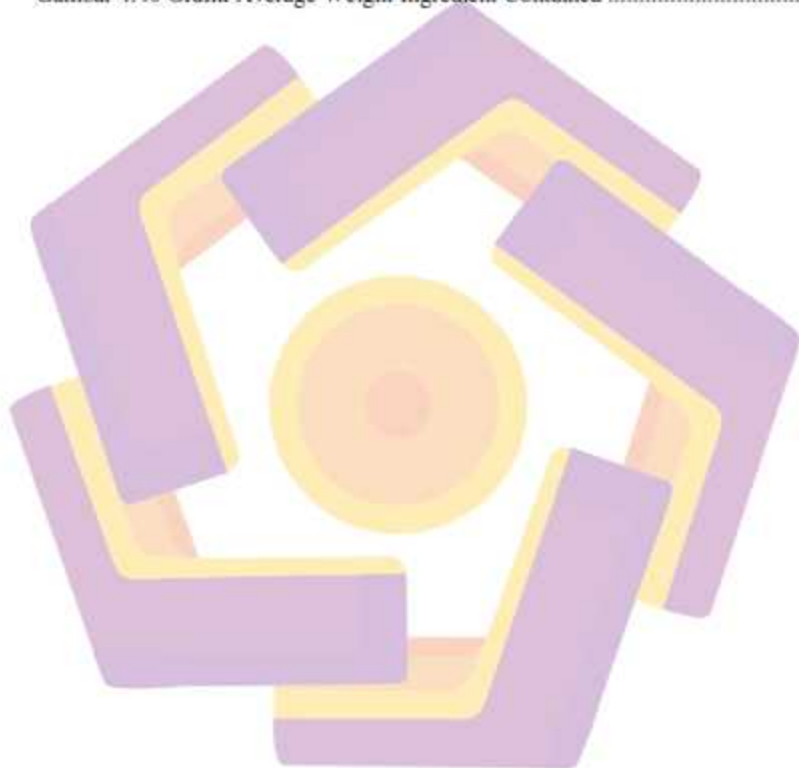
DAFTAR TABEL

Tabel 1.1 Hipotesis	6
Tabel 2.1 Matriks Literatur Review dan Posisi Penelitian Pencarian String <i>Boyer-Moore</i> Dalam Pendekatan <i>Content Based Filtering</i> Pada Sistem Rekomendasi Resep Masakan	10
Tabel 2.2 Contoh Case Folding	18
Tabel 2.3 Contoh Tokenizing Dari Hasil Case Folding	19
Tabel 2.4 Contoh Filtering Dari Hasil Tokenizing	20
Tabel 2.5 Contoh Stemming Dari Hasil Filtering	21
Tabel 2.6 Recall dan Precision pada Information Retrieval Evaluation	27
Tabel 2.7 Penilaian efektivitas Precision pada Information Retrieval System	27
Tabel 4.1 Perbedaan Skenario Model Rekomendasi	39
Tabel 4.2 Query Pengujian	58
Tabel 4.3 Recall and Precision Skenario 1	59
Tabel 4.4 Recall and Precision Skenario 2	59
Tabel 4.5 Recall and Precision Skenario 3	59
Tabel 4.6 Recall and Precision Skenario 4	59
Tabel 4.7 Recall and Precision Skenario 5	59
Tabel 4.8 Nilai rata-rata recall, precision dan f1-score	60
Tabel 4.9 Relevansi Hasil Rekomendasi Model Skenario 1	60
Tabel 4.10 Relevansi Hasil Rekomendasi Model Skenario 2	61
Tabel 4.11 Relevansi Hasil Rekomendasi Model Skenario 3	61
Tabel 4.12 Relevansi Hasil Rekomendasi Model Skenario 4	61
Tabel 4.13 Relevansi Hasil Rekomendasi Model Skenario 5	61
Tabel 4.14 Hasil Pengujian AP	62
Tabel 4.15 Mean Average Precision Semua Model Rekomendasi	63
Tabel 4.16 Bobot Kandungan Bahan Makanan Hasil Rekomendasi	64
Tabel 4.17 Bobot Rata-rata Bahan Makanan Terkandung	65
Tabel 4.18 Evaluasi Skenario Model Rekomendasi	66

DAFTAR GAMBAR

Gambar 2.1 Vektor Cosine Similarity	26
Gambar 3.1 Alur Penelitian.....	32
Gambar 4.1 Tabel Resep Masakan	34
Gambar 4.2 Tabel Bahan Makanan	35
Gambar 4.3 Skenario Model Pertama.....	36
Gambar 4.4 Skenario Model Kedua dan Ketiga.....	37
Gambar 4.5 Skenario Model Keempat dan Kelima.....	38
Gambar 4.6 Header Program, Koneksi Database dan Select Data.....	40
Gambar 4.7 Data dan Info Data Resep Masakan.....	41
Gambar 4.8 Data dan Info Data Bahan Makanan.....	41
Gambar 4.9 Fungsi lowercase dan remove special.....	42
Gambar 4.10 Fungsi remove number, punctuation, whitespace, whitespace multiple, single char, tokenizer	42
Gambar 4.11 Output Data Preprocessing Dan Data Token Preprocessing	43
Gambar 4.12 Stopword Removal	43
Gambar 4.13 Detokenizer	44
Gambar 4.14 Term Frequency Pada Program.....	44
Gambar 4.15 Matrix Cosine Similarity Pada Program.....	45
Gambar 4.16 Data Series Untuk Rekomendasi Model Skenario 1.....	45
Gambar 4.17 Fungsi Model Rekomendasi Skenario 1	45
Gambar 4.18 Fungsi Pencarian String Boyer Moore.....	46
Gambar 4.19 Fungsi Untuk Menuliskan Data Resep Hasil Preprocessing	46
Gambar 4.20 Fungsi Untuk Mengekstrak Bahan Dasar Resep Masakan.....	47
Gambar 4.21 Fungsi Pembobotan Boyer Moore.....	48
Gambar 4.22 Fungsi Model Rekomendasi Skenario 2	48
Gambar 4.23 Fungsi Model Rekomendasi Skenario 3	49
Gambar 4.24 Fungsi Model Rekomendasi Skenario 4	50
Gambar 4.25 Fungsi Model Rekomendasi Skenario 5	50
Gambar 4.26 Hasil Model Rekomendasi Skenario 1.....	51
Gambar 4.27 Grafik Kesamaan Bahan Makanan Model Rekomendasi Skenario 1	51
Gambar 4.28 Hasil Model Rekomendasi Skenario 2.....	52
Gambar 4.29 Grafik Kesamaan Bahan Makanan Model Rekomendasi Skenario 2	53
Gambar 4.30 Hasil Model Rekomendasi Skenario 3.....	54
Gambar 4.31 Grafik Kesamaan Bahan Makanan Model Rekomendasi Skenario 3	54
Gambar 4.32 Hasil Model Rekomendasi Skenario 4.....	55
Gambar 4.33 Grafik Kesamaan Bahan Makanan Model Rekomendasi Skenario 4	56

Gambar 4.34 Hasil Model Rekomendasi Skenario 5.....	57
Gambar 4.35 Grafik Kesamaan Bahan Makanan Model Rekomendasi Skenario 5	57
Gambar 4.36 Perbandingan Nilai Recall, Precision Dan F1-Score.....	60
Gambar 4.37 Grafik Average Precision Hasil Rekomendasi.....	62
Gambar 4.38 Grafik Pengujian Mean Average Precision.....	63
Gambar 4.39 Grafik Bobot Kandungan Makanan Hasil Rekomendasi.....	64
Gambar 4.40 Grafik Average Weight Ingredient Contained	65



INTISARI

Informasi resep masakan sangat membantu masyarakat dalam membuat menu masakan yang bervariasi setiap harinya. Berbagai aplikasi resep masakan dapat diunduh di internet dan masyarakat dapat berbagi informasi mengenai inovasi dan variasi lain dari sebuah resep masakan. Sering kali ide masakan lainnya sangat dibutuhkan yang sesuai dengan bahan makanan yang ada. Pendekatan *Content Based Filtering* (CBF) merekomendasikan dokumen dengan menghitung kedekatan jarak vektor, namun komposisi bahan makanan yang berbeda jauh, menyebabkan hasil rekomendasi tidak mampu merepresentasikan nilai relevansi dokumen terhadap *query*. Pada penelitian ini, penulis mengusulkan model rekomendasi CBF dengan algoritma *Boyer-Moore* (BM) untuk meningkatkan bobot rata-rata kandungan bahan makanan dan relevansi pada rekomendasi resep masakan. Dasar usulan adalah menggabungkan CBF dan BM dengan mempertimbangkan nilai bobot bahan makanan yang ditemukan pada dokumen. Hasil dari 5 skenario yang dilakukan pada penelitian ini, Model Rekomendasi Skenario ke-4 menggunakan TF-IDF *bigram* dan ekstraksi pada *query* dan data koleksi mampu menghasilkan nilai relevansi (MAP) sebesar 0,9240 dan AWIC sebesar 64,69783 dengan nilai F1-score 0,695443. Sedangkan pada model CBF standar hanya mampu menghasilkan nilai relevansi (MAP) sebesar 0,6350 dan nilai AWIC sebesar 46,18103 dengan nilai F1-score 0,658084.

Kata kunci: *content based-filtering*, TF-IDF, *cosine simiarity*, *boyer-moore*, resep masakan

ABSTRACT

Recipe information is very helpful for the community in making a varied menu of dishes every day. Various recipe applications can be downloaded on the internet and the public can share information about innovations and other variations of a recipe. Often other cooking ideas are needed that match the existing food ingredients. The Content Based Filtering (CBF) approach recommends documents by calculating the proximity of the vector distance, but the composition of food ingredients differs greatly, causing the recommendation results to not be able to represent the value of the relevance of the document to the query. In this study, the authors propose a CBF recommendation model with the Boyer-Moore (BM) algorithm to increase the average weight of food ingredients and their relevance to recipe recommendations. The basis of the proposal is to combine CBF and BM by considering the weight value of the food ingredients found in the document. The results of the 5 scenarios carried out in this study, the 4th Scenario Recommendation Model using the TF-IDF bigram and extraction of queries and data collections were able to produce a relevance (MAP) value of 0.9240 and an AWIC of 64.69783 with an F1-score value of 0.695443. While the standard CBF model is only able to produce a relevance (MAP) value of 0.6350 and an AWIC value of 46.18103 with an F1-score of 0.658084.

Keywords: content based-filtering, TF-IDF, cosine similarity, boyer-moore, recipes

BAB I PENDAHULUAN

1.1. Latar Belakang Masalah

Memasak pada prinsipnya adalah proses pemberian panas pada bahan makanan, untuk mematangkan dan menjadikan bahan makanan menjadi hidangan yang dapat dimakan, enak dan lezat (Karina dan Endang, 2017). Memasak sering dianggap sebuah seni dalam memilih bahan makanan, mempersiapkan, mengolah dan menyajikan makanan menjadi hidangan yang menarik. Ketrampilan sangat dibutuhkan dalam memasak, teknik pengolahan yang tepat, akan membuat hidangan menjadi sangat lezat untuk dinikmati walau hanya dengan menggunakan bahan makanan yang sederhana.

Bagi sebagian orang informasi resep masakan sangat dibutuhkan untuk menggali ketrampilan dan sebagai inovasi menu masakan. Bagi ibu rumah tangga, informasi resep masakan dibutuhkan untuk menyajikan menu yang bervariasi sehingga hidangan rumahan tidak membosankan. Informasi resep masakan baik masakan tradisional, modern, dari dalam dan luar negeri dapat dengan mudah ditemukan di internet pada situs-situs media sosial dan situs yang khusus menyajikan informasi resep masakan. Ide masakan yang kreatif dan inovatif sangat cepat berkembang seiring banyaknya pengguna situs yang membagikan pengalamannya dalam memasak. Setiap situs akan menampilkan resep masakan favorit, ulasan terbanyak dan rating terbanyak namun belum tentu sesuai dengan apa yang dicari oleh pengguna.

Sistem rekomendasi telah menjadi bagian penting dalam revolusi teknologi. Berbagai metode diterapkan untuk dapat memberikan rekomendasi yang relevan kepada setiap pengguna aplikasi. Sistem rekomendasi didefinisikan sebagai aplikasi pada website e-commerce untuk mengusulkan informasi dan menyediakan fasilitas yang diinginkan pengguna dalam membuat suatu keputusan (Ricci, F., 2002). Sistem tersebut diasumsikan sebagai representasi kebutuhan dan gambaran keinginan pengguna melalui pendekatan rekomendasi atau informasi lain yang memiliki kemiripan dari karakteristik informasi. Untuk menghasilkan informasi,

sistem rekomendasi memiliki beberapa metode yang digunakan untuk mendukung cara kerja diantaranya seperti *demographic recommendation*, *collaborative recommendation*, *content-based recommendation* dan *knowledge based recommendation*. Metode *collaborative* dan *content-based* adalah metode yang sering digunakan pada sistem rekomendasi karena teknik ini menyaring informasi berdasarkan keinginan pengguna dan berdasarkan content yang disediakan (Sebastia, L et al., 2009).

Dalam penelitian yang dilakukan dengan judul "Pencarian String Menggunakan Algoritma Boyer Moore Pada Dokumen" yang dilakukan oleh Eza Rahmanita, S.T., M.T pada tahun 2014. Dari hasil uji coba, ditemukan bahwa efektifitas Algoritma *Boyer-Moore* tergantung pada panjang kata yang dicari. Parameter seperti panjang karakter per kata dan perulangan karakter dalam kata dapat mempengaruhi efektivitas algoritma pencarian String. Pada tahun 2016, penelitian tersebut dijadikan pustaka oleh Halim Agung dan Yogyakarta pada jurnal "Implementasi *Boyer-Moore* Pada Aplikasi Pencarian Rumus Matematika Dan Fisika" dimana algoritma *Boyer-Moore* berfungsi dengan baik untuk menemukan rumus yang valid untuk menyelesaikan soal cerita matematika dan fisika tingkat SMA.

Penelitian lainnya yaitu oleh Putri Nastiti pada tahun 2019 dengan judul "Penerapan Metode *Content Based Filtering* Dalam Implementasi Sistem Rekomendasi Tanaman Pangan". Tujuan dari penelitian ini adalah Memberi rekomendasi bagaimana cara memangkas rantai distribusi pemasaran sehingga dapat meningkatkan kesejahteraan petani. Pengujian presisi dilakukan dengan membandingkan kelompok tani yang relevan bagi pedagang. Berdasarkan pengujian terhadap 10 profil pedagang dengan 15 rekomendasi kelompok tani teratas, didapatkan hasil *Average Precision* sebesar 78.40%.

Penelitian R. Janani dan S. Vijayarani dengan judul "An Efficient Text Pattern Matching Algorithm for Retrieving Information from Desktop" pada tahun 2016 mengambil informasi setelah menganalisa isi dokumen yang disimpan di desktop dengan menerapkan algoritma pencocokan string. Penelitian tersebut

menggunakan algoritma pencocokan pola untuk menemukan semua kemunculan serangkaian pola terbatas dalam teks input atau dokumen masukan. Beberapa algoritma yang digunakan dalam penelitian tersebut adalah Algoritma *Brute Force*, algoritma *Knuth-Morris-Pratt* (KMP), Algoritma *Boyer-Moore* dan Algoritma *Rabin Karp*. Dari hasil penelitian diketahui bahwa Algoritma *Boyer-Moore* pada file .txt memiliki nilai relevansi 100% dengan inputan Single Word, 100% dengan inputan Multiple Word dan 92% pada inputan File sedangkan pada file .docx memiliki nilai relevansi 99% dengan inputan Single Word, 97% dengan inputan Multiple Word dan 90% pada inputan File. Penelitian tersebut menunjukkan adanya peluang penambahan Algoritma *Boyer-Moore* kedalam *Recommender System*.

Pada setiap penelitian yang berkaitan dengan *Recommender System*, sistem rekomendasi ditujukan dapat mendorong pengguna membuat sebuah keputusan atau keyakinan dari informasi yang disediakan (Ke Ma, 2016). Setiap metode memiliki kelebihan dan kekurangan masing-masing, tidak terkecuali metode *Content-Based Filtering*. Dokumen yang sangat panjang namun hanya memiliki sedikit kandungan bahan makanan menyebabkan bobot makanan yang terkandung pada hasil rekomendasi tidak mampu merepresentasikan nilai relevansi dokumen terhadap *query*. Penggunaan *Boyer-Moore* dalam pencarian string mempertimbangkan kemunculan urutan kata yang sama antara *query* dan teks di dalam dokumen. Relevansi hasil rekomendasi dilihat dari presentase bobot bahan makanan yang ditemukan pada dokumen dibandingkan dengan panjang dokumen tersebut. Oleh karena itu pada penelitian ini, penulis melakukan beberapa skenario penelitian untuk mendapatkan model yang paling efektif pada penerapan algoritma *Boyer-Moore* dalam pendekatan *Content Based Filtering* pada sistem rekomendasi resep masakan. Efektifitas model didapatkan dengan menguji tingkat relevansi serta bobot rata-rata kandungan bahan makanan pada hasil rekomendasi.

1.2. Rumusan Masalah

Berdasarkan latar belakang yang telah disampaikan penulis, rumusan masalah yang diangkat adalah sebagai berikut :

- a. Seperti apa model yang efektif untuk penerapan algoritma *Boyer-Moore*

dalam pendekatan *Content Based Filtering* pada sistem rekomendasi resep masakan?

- b. Seberapa besar nilai *recall*, *precision*, *F1-score*, bobot rata-rata kandungan bahan makanan dan tingkat relevansi yang dihasilkan dari model penerapan algoritma *Boyer-Moore* dalam pendekatan *Content Based Filtering* pada sistem rekomendasi resep masakan?

1.3. Batasan Masalah

Penelitian ini akan berfokus pada permasalahan yang telah dirumuskan sebelumnya dan memiliki batasan masalah sebagai berikut :

- a. Hasil akhir dari penelitian ini berupa model rekomendasi.
- b. Model rekomendasi yang diteliti menggunakan 5 skenario dan diuji untuk diperoleh skenario paling efektif berdasarkan tingkat *Precision*, hasil pengujian MAP dan AWIC tertinggi.
- c. Pengujian model berupa perbandingan nilai rata-rata bobot komposisi serta nilai relevansi sebelum dan setelah menggunakan algoritma *Boyer-Moore* dengan *Recall*, *Precision*, *F1-score* dan *Mean Average Precision (MAP) Information Retrieval*.
- d. Penelitian menggunakan dataset resep masakan indonesia, bahan makanan dan stopword bahasa indonesia dari kaggle.

1.4. Tujuan Penelitian

Penelitian ini memiliki tujuan sebagai berikut :

- a. Mendapatkan model yang paling efektif dari penerapan algoritma *Boyer-Moore* dalam pendekatan *Content Based Filtering* pada sistem rekomendasi resep masakan.
- b. Mendapatkan perbandingan *recall*, *precision*, *F1-score*, bobot komposisi bahan makanan dan nilai relevansi pada setiap skenario penelitian penerapan Algoritma *Boyer-Moore* dalam pendekatan *Content Based Filtering*.

1.5. Manfaat Penelitian

Manfaat dari penelitian ini adalah :

- a. Mengetahui model yang paling efektif untuk penerapan algoritma *Boyer-Moore* dalam pendekatan *Content Based Filtering* pada sistem rekomendasi resep masakan.
- b. Mengetahui efektifitas dari pengujian *recall*, *precision*, *F1-score*, tingkat rata-rata bobot komposisi dan nilai relevansi pada setiap skenario model rekomendasi.
- c. Memberikan kontribusi pada penelitian yang berkaitan dengan *Recommender System* berbasis *Content Based Filtering* yang mempertimbangkan urutan kata atau gabungan kata.
- d. Dapat dijadikan referensi ilmu pengetahuan dan pengembangan teknologi.
- e. Dapat digunakan untuk mengembangkan perangkat lunak yang berkaitan dengan sistem rekomendasi berbasis konten yang mengutamakan komposisi lebih dari satu suku kata sebagai *keyword*.

1.6. Hipotesis

Resep masakan merupakan informasi yang memuat nama masakan, bahan masakan, cara memasak dan kandungan gizi dari masakan tersebut. Ada banyak metode untuk merekomendasikan resep masakan kepada pengguna, namun untuk penelitian ini metode yang cocok digunakan adalah *content based filtering*. Penulis memilih pendekatan *content based filtering* berdasarkan permasalahan menemukan saran resep masakan sesuai dengan bahan makanan yang terkandung didalamnya.

Bahan makanan memiliki lebih dari satu suku kata, sedangkan algoritma TF-IDF dan *cosine similarity* pada *content based filtering* hanya mengolah kedekatan kalimat per satu suku kata dan mengukur berdasarkan kemiripan vektornya. Vektorisasi pada algoritma TF-IDF dihitung berdasarkan kemunculan setiap kata pada dokumen setelah melalui tahap *preprocessing*. Algoritma *boyer moore* memiliki kemampuan untuk menemukan kombinasi suku kata di dalam dokumen. Pembobotan suku kata berdasarkan total kata pada *corpus* menjadi menunjukkan seberapa tinggi tingkat kesamaan komposisi bahan makanannya.

Dengan metode ini, rekomendasi resep masakan akan lebih relevan dengan komposisi bahan makanan yang ada pada resep masakan.

Penulis membandingkan dua buah kalimat sederhana untuk membentuk hipotesis rekomendasi resep masakan antara *content based filtering* ($tf-idf + cosine\ similarity$), algoritma *boyer moore* dan penerapan algoritma *boyer moore* pada *content based filtering* pada tabel 1.1 berikut :

Tabel 1.1 Hipotesis

Kata Kunci (<i>Query</i>)	Telur Ayam, Bawang merah	Telur Ayam, Bawang merah
Resep	Kaldu Ayam, Telur puyuh, Bawang putih, garam	Telur Ayam, Bawang Putih, Bawang merah
<i>Content Based Filtering</i>	Direkomendasikan. Karena mengandung ayam, telur, bawang	Direkomendasikan. Karena mengandung telur, ayam, bawang dan merah
<i>Boyer moore</i>	Tidak direkomendasikan. Karena tidak sesuai dengan <i>query</i>	Direkomendasikan. karena terdapat "Telur Ayam" dan "Bawang merah"
<i>Content Based Filtering + Boyer moore</i>	Tidak relevan terhadap bahan makanan pada <i>query</i>	Relevan karena mengandung bahan makanan dengan bobot rata-rata bahan makanan diatas 50% yang ada pada <i>query</i> .

Berdasarkan rumusan masalah yang telah dijabarkan, maka kesimpulan sementara atau hipotesa yang dapat ditarik adalah model penerapan algoritma *Boyer-Moore* dalam pendekatan *Content Based Filtering* dapat meningkatkan nilai rata-rata bobot komposisi pada hasil rekomendasi. Dengan meningkatnya nilai rata-rata bobot komposisi bahan makanan tersebut maka nilai relevansi dari hasil rekomendasi juga akan meningkat.

BAB II TINJAUAN PUSTAKA

2.1. Tinjauan Pustaka

Penelitian ini dilakukan tidak terlepas dari penelitian yang pernah dilakukan sebelumnya yaitu "Pencarian String Menggunakan Algoritma Boyer Moore Pada Dokumen" yang dilakukan oleh Eza Rahmanita, S.T., M.T pada tahun 2014. Dari hasil uji coba, ditemukan bahwa efektifitas Algoritma Boyer Moore tergantung pada panjang kata yang dicari. Semakin panjang kata yang dicari, maka semakin hemat waktu yang ditempuh pada file berekstensi .txt. namun pada file berekstensi .doc dan .pdf hal ini merupakan kebalikannya, artinya semakin sedikit m yang ditemukan maka semakin cepat waktu prosesnya, semakin banyak m yang ditemukan maka akan semakin lambat. Sehingga panjang karakter per kata dan perulangan karakter dalam kata dapat mempengaruhi efektivitas algoritma pencarian String.

Halim Agung dan Yogyawan (2016) dengan judul "Implementasi *Boyer-Moore* Pada Aplikasi Pencarian Rumus Matematika Dan Fisika", menyimpulkan bahwa metode string matching dengan algoritma *Boyer-Moore* dapat diterapkan pada aplikasi pencarian rumus matematika dan fisika tingkat SMA yang disertai dengan pengujian algoritma sebanyak 30 kali dimana aplikasi dapat mencocokkan pola kata kunci pada soal serta menampilkan list atau kumpulan rumus matematika dan fisika tingkat SMA yang cocok dan sesuai dengan soal yang dikerjakan

Putri Nastiti (2019) dengan judul "Penerapan *Metode Content Based Filtering* Dalam Implementasi Sistem Rekomendasi Tanaman Pangan". Tujuan dari penelitian ini adalah Memberi rekomendasi bagaimana cara memangkas rantai distribusi pemasaran sehingga dapat meningkatkan kesejahteraan petani. Pengujian presisi dilakukan dengan membandingkan kelompok tani yang relevan bagi pedagang. Berdasarkan pengujian terhadap 10 profil pedagang dengan 15 rekomendasi kelompok tani teratas, didapatkan hasil *Average Precision* sebesar 78.40%. Namun dalam pengujian *confusion matrix* hanya perhitungan *precision* saja yang dilakukan sehingga tidak diketahui besaran nilai *accuracy* dan *recall*-nya.

Setiawan et al., (2019) dengan judul “Aplikasi Rekomendasi Pemilihan Menu Makanan Sehat Untuk Anak Penderita Obesitas Menggunakan Metode Fuzzy Tahani”. Tujuan penelitian ini memberikan rekomendasi pemilihan menu makanan sehat untuk anak penderita obesitas. Hasil dari penelitian yang dilakukan tersebut adalah dari 100 data menu makanan yang diambil pada Daftar Komposisi Bahan Makanan (DKBM) didapatkan hasil uji dari 10 data uji dengan hasil perbandingan rasio tingkat akurasi tinggi sebesar 70% dan tingkat akurasi rendah sebesar 30%.

Ke Ma (2016) dengan judul “*Content-based Recommender System for Movie Website*” dimana tesis tersebut bertujuan untuk menemukan cara yang baru untuk meningkatkan klasifikasi film dan meningkatkan rekomendasi berbasis konten film kepada penggunanya. Penelitian tersebut menggunakan TF-IDF-DC dan mengekstraksi fitur untuk mengevaluasi nilai akurasi dan selanjutnya menggunakan KNN dan berbagai metrik untuk mengevaluasi peningkatan pendekatan baru. Diilustrasikan bahwa pendekatan baru berkontribusi positif sesuai dengan evaluasi.

Wayan Gede Suka Parwita (2019), melakukan penelitian dengan judul “Penguji Akurasi Sistem Rekomendasi Berbasis *Content-Based Filtering*”. Penelitian tersebut meninjau pengaruh *stopword* dalam tingkat akurasi penentuan rekomendasi dokumen dengan klasifikasi dokumen yang digunakan. Penelitian tersebut menyimpulkan nilai sistem rekomendasi dengan proses *stopword removal* masih lebih unggul dibandingkan sistem rekomendasi tanpa proses *stopword removal*. Hal ini tampak pada nilai *precision*, *recall*, dan *f-measure* yang semua dicapai pada sistem rekomendasi dengan proses *stopword removal* yaitu *precision* dengan nilai 0,3903 pada *minimum similarity* 30%, *recall* 0,9870 pada *minimum similarity* 5%, dan *f-measure* 0,3202 pada *minimum similarity* 20%.

Penelitian R. Janani dan S. Vijayarani dengan judul “An Efficient Text Pattern Matching Algorithm for Retrieving Information from Desktop” pada tahun 2016 mengambil informasi setelah menganalisa isi dokumen yang disimpan di desktop dengan menerapkan algoritma pencocokan string. Penelitian tersebut menggunakan algoritma pencocokan pola untuk menemukan semua kemunculan

serangkaian pola terbatas dalam teks input atau dokumen masukan. Beberapa algoritma yang digunakan dalam penelitian tersebut adalah Algoritma *Brute Force*, algoritma *Knuth-Morris-Pratt* (KMP), Algoritma *Boyer-Moore* dan Algoritma *Rabin Karp*. Dari hasil penelitian diketahui bahwa Algoritma *Boyer-Moore* pada file .txt memiliki nilai relevansi 100% dengan inputan Single Word, 100% dengan inputan Multiple Word dan 92% pada inputan File sedangkan pada file .docx memiliki nilai relevansi 99% dengan inputan Single Word, 97% dengan inputan Multiple Word dan 90% pada inputan File. Penelitian tersebut menunjukkan adanya peluang penambahan Algoritma *Boyer-Moore* kedalam *Recommender System*.



2.2. Keaslian Penelitian

Tabel 2.1 Matriks Literatur Review dan Posisi Penelitian Pencarian String *Boyer-Moore* Dalam Pendekatan *Content Based Filtering* Pada Sistem Rekomendasi Resep Masakan

No	Judul	Peneliti, Publikasi, Tahun	Tujuan Penelitian	Kesimpulan	Saran dan Kelemahan	Perbandingan
1	Pencarian String Menggunakan Algoritma Boyer Moore Pada Dokumen	Eza Rahmanita, S.T., M.T. Jurnal Ilmiah NERO Vol. 1 No. 1 2014	Mengetahui tingkat efektifitas dengan menghitung kecepatan <i>boyer-moore</i> dalam menemukan kata kunci.	Efektifitas Algoritma Boyer Moore tergantung pada panjang kata yang dicari	Dapat digunakan secara maksimal pada pencarian string dokumen yang berindeks dokumen medis ICD-10	Penelitian pada naskah ini untuk mengetahui tingkat relevansi dan bobot rata-rata kandungan bahan makanan pada rekomendasi resep masakan.
2	Implementasi Boyer-Moore Pada Aplikasi	Halim Agung, Yogyakarta JITTER,	Mendeteksi kata kunci untuk menemukan rumus yang tepat	Metode string matching dengan algoritma <i>Boyer-</i>	Bisa dikembangkan dengan membuat penelitian yang dapat	Penelitian pada naskah ini menemukan rekomendasi yang

Tabel 2.1 Matriks Literatur Review dan Posisi Penelitian Pencarian String *Boyer-Moore* Dalam Pendekatan *Content Based Filtering* Pada Sistem Rekomendasi Resep Masakan (lanjutan)

No	Judul	Peneliti, Publikasi, Tahun	Tujuan Penelitian	Kesimpulan	Saran dan Kelemahan	Perbandingan
	Pencarian Rumus Matematika Dan Fisika	Volume III, No I, 15 Desember 2016	dalam menyelesaikan soal cerita matematika dan fisika tingkat SMA.	<i>Moore</i> dapat diterapkan pada aplikasi pencarian rumus matematika dan fisika tingkat SMA.	menjawab soal matematika dan fisika dengan disertai tahapan penyelesaiannya	sesuai dengan komposisi bahan makanan dari konten yang sedang diakses.
3	Penerapan Metode Content Based Filtering Dalam Implementasi Sistem	Putri Nastiti, Teknik, 2019, 8.1: 1-10	Memberi rekomendasi bagaimana cara memangkas rantai distribusi rantai pemasaran sehingga dapat meningkatkan kesejahteraan petani.	Berdasarkan pengujian terhadap 10 profil pedagang dengan 15 rekomendasi kelompok tani teratas, didapatkan	Sistem akan lebih baik apabila pengujian sistem dilakukan tidak hanya menggunakan precision tetapi juga menggunakan recall.	Pengujian AP menggunakan <i>confusion matrix</i> sedangkan naskah ini pengujian AP dari <i>Information Retrieval</i>

Tabel 2.1 Matriks Literatur Review dan Posisi Penelitian Pencarian String *Boyer-Moore* Dalam Pendekatan *Content Based Filtering* Pada Sistem Rekomendasi Resep Masakan (lanjutan)

No	Judul	Peneliti, Publikasi, Tahun	Tujuan Penelitian	Kestmpulan	Saran dan Kelemahan	Perbandingan
	Rekomendasi Tanaman Pangan			hasil <i>Average precision</i> sebesar 78.40%		
4	Aplikasi Rekomendasi Pemilihan Menu Makanan Sehat Untuk Anak Penderita Obesitas Menggunakan Metode Fuzzy Tahani	Erdi Setiawan, Soegiarto Progresif: Jurnal Ilmiah Komputer, 2017, 13.1.	Memberikan rekomendasi pemilihan menu makanan sehat untuk anak penderita obesitas	Hasil yang didapat dari 100 data menu makanan yang diambil pada daftar komposisi bahan makanan (DKBM) didapatkan hasil uji implementasi dari 10 data uji dengan hasil perbandingan	Dari 100 data menu makanan yang diambil pada daftar komposisi bahan makanan (DKBM) didapatkan hasil uji implementasi dengan hasil perbandingan rasio tingkat akurasi tinggi sebesar 70%	Perbedaan dengan penelitian yang akan dilakukan terletak pada metode yang digunakan.

Tabel 2.1 Matriks Literatur Review dan Posisi Penelitian Pencarian String *Boyer-Moore* Dalam Pendekatan *Content Based Filtering* Pada Sistem Rekomendasi Resep Masakan (lanjutan)

No	Judul	Peneliti, Publikasi, Tahun	Tujuan Penelitian	Kesimpulan	Saran dan Kelemahan	Perbandingan
				rasio dengan tingkat akurasi tinggi sebesar 70% dan tingkat akurasi rendah sebesar 30%	dan tingkat akurasi rendah sebesar 30%.	
5	Content-based Recommender System for Movie Website	Ke Ma, 2016, KTH Royal Institute of Technology, Swedia	Menemukan cara yang baru untuk meningkatkan klasifikasi film dan meningkatkan rekomendasi berbasis konten film.	Algoritma KNN memberikan ilustrasi positif terhadap nilai evaluasi yang dilakukan dengan TF-IDF-DC.	Masih banyak aspek yang perlu dipertimbangkan untuk penelitian selanjutnya.	Penelitian yang akan dilakukan mengusulkan Algoritma Boyer-Moore untuk mengevaluasi sistem rekomendasi berbasis konten pada resep masakan.

Tabel 2.1 Matriks Literatur Review dan Posisi Penelitian Pencarian String *Boyer-Moore* Dalam Pendekatan *Content Based Filtering* Pada Sistem Rekomendasi Resep Masakan (lanjutan)

No	Judul	Peneliti, Publikasi, Tahun	Tujuan Penelitian	Kesimpulan	Saran dan Kelemahan	Perbandingan
6	Pengujian Akurasi Sistem Rekomendasi Berbasis Content-Based Filtering	Wayan Gede Suka Parwita, JIJK 2019, 14.1	Meninjau pengaruh stopword dalam tingkat akurasi penentuan rekomendasi dokumen dengan klasifikasi dokumen yang digunakan.	Nilai sistem rekomendasi dengan proses <i>stopword removal</i> masih lebih unggul dibandingkan sistem rekomendasi tanpa proses <i>stopword removal</i> .		Pada penelitian yang akan dilaksanakan, proses <i>stopword removal</i> yang akan digunakan menggunakan dataset stopwords bahasa Indonesia.

Tabel 2.1 Matriks Literatur Review dan Posisi Penelitian Pencarian String *Boyer-Moore* Dalam Pendekatan *Content Based Filtering* Pada Sistem Rekomendasi Resep Masakan (lanjutan)

No	Judul	Peneliti, Publikasi, Tahun	Tujuan Penelitian	Kestmpulan	Saran dan Kelemahan	Perbandingan
7	An Efficient Text Pattern Matching Algorithm for Retrieving Information from Desktop	R. Janani dan S. Vijayarani. Indian Journal of Science and Technology, Vol 9(43). 2016	Mengukur Kinerja Algoritma <i>Brute Force</i> , algoritma <i>Knuth-Morris-Pratt</i> (KMP), Algoritma <i>Boyer-Moore</i> dan Algoritma <i>Rabin Karp</i> dalam menemukan pola terbatas pada kumpulan dokumen di pencarian desktop.	Algoritma KMP memberikan hasil akurasi yang lebih baik dari algoritma pencarian string lainnya.		Penulis mengacu pada perhitungan Boyer-moore yang diteliti sebagai perbandingan hasil pengujian penelitian yang penulis lakukan.

2.3. Landasan Teori

2.3.1. Resep Masakan

Resep merupakan alat bantu seorang juru masak untuk membuat sebuah hidangan, sehingga resep merupakan sekumpulan instruksi atau langkah-langkah kerja untuk membuat suatu masakan, memberikan petunjuk secara jelas dan urut mengenai prosedur bekerja seperti; mencampur, mengolah hingga memasaknya (karina dan endang, 2017). Resep yang baik adalah resep yang sangat mudah untuk di baca, difahami dan dipraktikkan oleh si pembaca dan pembuat masakan.

Resep masakan secara harfiah adalah takaran dari bahan-bahan makanan yang digunakan untuk membuat masakan baik makanan maupun minuman yang sebelumnya telah teruji. Untuk dapat membuat masakan tentunya si pemasak (juru masak) harus menyiapkan bahan-bahan terlebih dahulu untuk diolah menjadi masakan siap saji. Selain menyiapkan bahan, dalam resep masakan juga tersedia keterangan dan panduan seputar cara mengolah bahan-bahan yang akan dimasak serta keterangan tentang cara menyajikan hasil masakan tersebut.

Menurut karina dan endang (2017), terdapat beberapa faktor yang menjadi fundamental dalam meracik, mengolah, mengkreasikan dan menghadirkan masakan, yaitu :

1. Berapa lama waktu yang dibutuhkan sebelum masakan yang dibuat mencapai titik basi / rusak.
2. Jumlah kalori yang akan dihasilkan dalam setiap masakan.
3. Durasi yang dibutuhkan untuk memasak makanan.
4. Media untuk menyajikan hasil masakan.

2.3.2. Sistem Rekomendasi

Menurut Yuan et al., (2014) seperti dikutip oleh Chandra Saha Dewa Prasetya (2017), sistem rekomendasi yang menggunakan metode *content based* menggunakan kesamaan produk untuk ditawarkan kepada pengguna. Namun, metode *content based* memiliki kelemahan yaitu ketika fitur konten yang tersedia

terbatas, maka akurasi rekomendasi yang dihasilkan cukup rendah. Sistem rekomendasi sangat dibutuhkan untuk membandingkan atau memberikan suatu nilai saran. Sebuah sistem rekomendasi membantu dalam membuat suatu keputusan untuk mendukung aktivitas bisnis dan organisasi. Rekomendasi yang dibuat akan memberikan saran bagi penggunanya mengenai item/produk yang mungkin akan diminati atau menarik untuk penggunanya.

Rekomendasi adalah fitur-fitur dan teknik-teknik pada perangkat lunak yang menyediakan sesuatu hal yang berguna untuk user (Ricci, et al.2011). Sistem rekomendasi juga menyediakan saran-saran dari beberapa item yang berpotensi menarik untuk pengguna. Saran yang diberikan erat kaitannya dengan pengambilan keputusan, seperti item apa saja yang harus dibeli, musik seperti apa yang harus didengarkan, dan berita apa yang harus dibaca. Dalam hal ini, item adalah sebuah objek yang direkomendasikan.

Terdapat tiga teknik rekomendasi utama yaitu: *collaborative filtering*, *content-based filtering*, dan *knowledge-based recommendation*. *Collaborative filtering* merupakan metode yang merekomendasikan sebuah item yang berdasarkan pada kemiripan ketertarikan antar pengguna. Sistem rekomendasi *content-based* merekomendasikan item yang mirip dengan yang disukai user sebelumnya. Nilai kesamaan antar item dihitung berdasarkan fitur yang ada pada setiap konten. Sistem rekomendasi *knowledge-based* merekomendasikan item berdasarkan domain pengetahuan yang spesifik tentang bagaimana fitur-fitur yang ada pada suatu item dapat memenuhi kebutuhan pengguna dan berguna bagi pengguna. Nilai kesamaan dihitung berdasarkan seberapa besar nilai kesamaan antara kebutuhan pengguna dengan rekomendasi yang ada. Terdapat dua pendekatan dalam metode *knowledge-based recommendation*, yaitu *case-based* dan *constraint-based recommendation*. Kesamaan dari kedua pendekatan ini adalah pengguna harus memberikan permintaan terlebih dulu. Kemudian sistem akan mengidentifikasi solusi yang sesuai dengan permintaan pengguna.

2.3.3. Text Preprocessing

Text preprocessing adalah proses untuk menyeleksi data teks agar menjadi terstruktur dengan serangkaian tahapan yang meliputi tahapan *case folding*, *tokenizing*, *filtering* dan *stemming*. Tidak ada aturan pasti pada setiap tahapan dalam *text preprocessing*, tergantung dengan jenis serta kondisi data yang dimiliki. *Text preprocessing* merupakan salah satu implementasi dari *text mining*. *Text mining* sendiri adalah suatu kegiatan menambang data, dimana data yang biasanya diambil berupa teks yang bersumber dari dokumen-dokumen yang memiliki tujuan akhir untuk mencari kata kunci yang mewakili dari sekumpulan dokumen tersebut yang selanjutnya digunakan untuk analisa hubungan antar dokumen-dokumen tersebut.

A. Case Folding

Tahapan pertama yang banyak dilakukan adalah *case folding*. Tahapan ini hampir selalu digunakan ketika melakukan *text preprocessing*. Data yang tidak terstruktur dan tidak konsisten dalam penggunaan huruf kapital menjadi alasan utama digunakannya tahapan *case folding*. Tahapan ini lebih sering digunakan untuk mengubah huruf kapital pada dokumen menjadi huruf kecil (*lowercase*) sebagai bentuk standar. Sebagai contoh text "Satu Siung Bawang puTIH" akan dirubah menjadi "satu siung bawang putih".

Tabel 2.2 Contoh Case Folding

Dokumen asli	Case Folding
Satu Ekor Ayam, 5 Siung Bawang merah diHaluskan, 4 buah daun SERAI dan Kecap Asin secukupnya.	satu ekor ayam, 5 siung bawang merah dihaluskan, 4 buah daun serai dan kecap asin secukupnya.

B. Tokenizing

Tahap *Tokenizing* memudahkan proses analisis data dengan memecah kalimat menjadi kata atau disebut dengan token. Setiap kalimat akan dipotong-potong berdasarkan tiap kata penyusunnya. Tokenisasi

secara garis besar memecah sekumpulan karakter dalam suatu teks ke dalam satuan kata, bagaimana membedakan karakter-karakter tertentu yang dapat diperlakukan sebagai pemisah kata atau bukan.

Sebagai contoh karakter *whitespace* seperti enter, tabulasi, spasi dianggap sebagai pemisah kata. Namun untuk karakter petik tunggal ('), titik (.), semikolon (;), titik dua (:) atau lainnya, dapat memiliki peran yang cukup banyak sebagai pemisah kata. Karakter-karakter dalam teks sangat tergantung pada konteks aplikasi yang dikembangkan.

Tabel 2.3 Contoh Tokenizing Dari Hasil *Case Folding*

Hasil <i>Case Folding</i>	<i>Tokenizing</i>
satu ekor ayam, 5 siung bawang merah dihaluskan, 4 buah daun serai dan kecap asin secukupnya.	satu ekor ayam 5 siung bawang merah dihaluskan 4 buah daun serai dan kecap asin secukupnya

C. *Filtering*

Tahap *Filtering* adalah tahap mengambil kata-kata penting dari hasil token menggunakan algoritma *stoplist* (membuang kata kurang penting) atau *wordlist* (menyimpan kata penting). *Stoplist/stopword* adalah kata-kata yang tidak deskriptif yang dapat dibuang dalam pendekatan *Bag-of-Words*. Contoh *stopwords* adalah "yang", "dan", "di", "dari" dan seterusnya.

Kata-kata seperti "dari", "yang", "di", dan "ke" adalah beberapa contoh kata-kata yang berfrekuensi tinggi dan dapat ditemukan hampir dalam setiap dokumen (disebut sebagai *stopword*). Tahap *filtering* dapat

juga digunakan untuk membersihkan tanda baca, angka dan karakter simbol. *Stopword* ini dapat mengurangi ukuran index dan waktu pemrosesan dan dapat mengurangi level noise.

Tabel 2.4 Contoh *Filtering* Dari Hasil *Tokenizing*

Hasil <i>Tokenizing</i>	<i>Filtering</i>
Satu ekor ayam 5 siung bawang merah dihaluskan 4 buah daun serai dan kecap asin secukupnya	satu ekor ayam siung bawang merah dihaluskan buah daun serai kecap asin secukupnya

D. *Stemming*

Teknik *Stemming* diperlukan selain untuk memperkecil jumlah indeks yang berbeda dari suatu dokumen, juga untuk melakukan pengelompokan kata-kata lain yang memiliki kata dasar dan arti yang serupa namun memiliki bentuk atau form yang berbeda karena mendapatkan imbuhan yang berbeda. Sebagai contoh kata bersama, kebersamaan, menyamai, akan *di-stemming* ke kata dasarnya yaitu "sama". Namun, seperti halnya *stopping*, kinerja *stemming* juga bervariasi dan sering tergantung pada domain bahasa yang digunakan.

Proses *stemming* pada teks berbahasa Indonesia berbeda dengan *stemming* pada teks berbahasa Inggris. Pada teks berbahasa Inggris, proses yang diperlukan hanya proses menghilangkan *sufiks*. Sedangkan pada teks berbahasa Indonesia semua kata imbuhan baik itu *sufiks* dan *prefiks* juga dihilangkan.

Tabel 2.5 Contoh *Stemming* Dari Hasil *Filtering*

Hasil <i>Filtering</i>	<i>Stemming</i>
Satu	satu
ekor	ekor
ayam	ayam
siung	siung
bawang	bawang
merah	merah
dihaluskan	halus
buah	buah
daun	daun
serai	serai
kecap	kecap
asin	asin
secukupnya	cukup

2.3.4. Algoritma Pencarian String *Boyer-Moore*

Cara kerja dari algoritma *Boyer-Moore* adalah dengan melakukan pencocokan karakter *pattern* dari paling kanan ke kiri dari string yang ingin dicari. Dengan menggunakan algoritma *Boyer-Moore*, rata-rata proses pencarian akan lebih cepat dibanding dengan proses pencarian lainnya. Ide dibalik algoritma *Boyer-Moore* adalah bahwa dengan memulai pencocokkan karakter dari kanan, dan bukan dari kiri, maka akan lebih banyak informasi yang didapat (Efendi, 2012).

Algoritma *Boyer-Moore* menggunakan dua fungsi *shift* yaitu *good-suffix shift* dan *bad-character shift* untuk mengambil langkah berikutnya setelah terjadi ketidakcocokan antara karakter *pattern* dan karakter teks yang dicocokkan (Sagita dan Prasetyowati, 2013). Sebagai contoh ada sebuah usaha pencocokan yang terjadi pada teks[$i..i + n - 1$], dan anggap ketidakcocokan pertama terjadi di antara teks[$i + j$] dan $pattern[j]$, dengan $0 < j < n$. Berarti, teks[$i + j + 1..i + n - 1$] - $pattern[j + 1..n - 1]$ dan $a = \text{teks}[i + j]$ tidak sama dengan $b = pattern[j]$. Jika u adalah akhiran dari *pattern* sebelum b dan v adalah sebuah awalan dari *pattern*, maka penggeseran yang mungkin adalah :

1. Penggeseran *good-suffix* yang terdiri dari mensejajarkan potongan teks[$i + j + 1..i + n - 1$] - $pattern[j + 1..n - 1]$ dengan kemunculannya paling kanan

di *pattern* yang didahului oleh karakter yang berbeda dengan *pattern[j]*. Jika tidak ada potongan seperti itu, maka algoritma akan mensejajarkan akhiran v dari teks[$i + j + 1..i + n - 1$] dengan awalan dari *pattern* yang sama.

2. Penggeseran *bad-character* yang terdiri dari mensejajarkan teks[$i + j$] dengan kemunculan paling kanan karakter tersebut di *pattern*. Bila karakter tersebut tidak ada di *pattern*, maka *pattern* akan disejajarkan dengan teks[$i + n + 1$].

Secara sistematis, langkah-langkah yang dilakukan algoritma *boyer moore* pada saat mencocokkan string menurut Sagita dan Prasetyowati (2013) adalah :

1. Algoritma *Boyer-Moore* mulai mencocokkan *pattern* pada awal teks.
2. Dari kanan ke kiri, algoritma ini akan mencocokkan karakter per karakter *pattern* dengan karakter di teks yang bersesuaian, sampai salah satu kondisi berikut dipenuhi :
 - a. Karakter di *pattern* dan di teks yang dibandingkan tidak cocok (*mismatch*).
 - b. Semua karakter di *pattern* cocok. Kemudian algoritma akan memberitahukan penemuan di posisi ini.
3. Algoritma kemudian menggeser *pattern* dengan memaksimalkan nilai penggeseran *good-suffix* dan penggeseran *bad-character*, lalu mengulangi langkah 2 sampai *pattern* berada di ujung teks.

Dengan menggunakan algoritma *boyer moore*, sistem akan mencari kecocokan dengan *pattern* pada *query* dan menghitung jumlah *pattern* yang ditemukan di dalam konten. Hasil perhitungan tersebut menentukan nilai bobot kandungan *query* di setiap konten masakan yang akan di rekomendasikan. Nilai bobot diperoleh dari jumlah kata dari *query* yang ditemukan dalam dokumen dibagi dengan jumlah total kata dari dokumen tersebut dengan rumus :

$$W_{bm} = \frac{\sum(Lq_i * Fq_i)}{L} \quad (1)$$

Dimana :

W_{bm} = bobot *Boyer-Moore*

L_{qi} = panjang query ke- i

F_{qi} = jumlah pattern ke- i yang cocok dengan *query*

L = jumlah kata di dalam dokumen

2.3.5. Content Based Filtering

Konsep *information filtering* dimanfaatkan pada sistem rekomendasi. Pengguna sudah mempunyai profil pada *information filtering* untuk merepresentasikan kepentingan jangka panjang dan sistem memberikan kepada setiap pengguna untuk item yang relevan. Berdasarkan ukuran kesamaan profil, sistem memilih dan membuat peringkat atas item yang relevan, kemudian memberikan rekomendasi kepada pengguna. Pendekatan pada *information filtering* terdapat dua (2), yaitu *collaborative filtering* dan *content-based filtering*.

Menurut Ma Ke (2016), model akan mengeskrak kata kunci dari item dan mengkalkulasi bobot TF-IDF. Misal k_i sebagai kata kunci ke- i dari item d_j dan w_{ij} adalah bobot dari k_i untuk d_j . Maka konten d_j dapat di definisikan sebagai berikut:

$$\text{Content}(d_j) = \{w_{1j}, w_{2j}, \dots\} \quad (2)$$

Sistem rekomendasi berbasis konten akan menyarankan item mirip dengan apa yang disukai pengguna sebelumnya. Sehingga selera pengguna dapat dimodelkan menurut riwayat yang di sukai pengguna. Jika *ContentBasedProfile* (u) sebagai vektor preferensi pengguna u , maka dapat di definisikan :

$$\text{ContentBasedProfile}(u) = \frac{1}{|N(u)|} \sum_{d \in N(u)} \text{Content}(d) \quad (3)$$

$N(u)$ adalah apa yang disukai pengguna sebelumnya. Setelah menghitung konten vektor *Content*(\cdot) dan vektor konten *ContentBasedProfile*(\cdot) dari semua pengguna, diberikan ke setiap pengguna u dan item d , bagaimana pengguna menyukai item tersebut di definisikan sebagai kesamaan antara *ContentBasedProfile* (u) dan *Content* (d).

$$p(u,d) = \text{sim}(\text{ContentBasedProfile}(u), \text{Content}(d)) \quad (4)$$

Keuntungan dari pendekatan ini adalah mendapatkan wawasan tentang

suatu item yang dianggap relevan oleh penggunanya dikarenakan konten dari setiap item diketahui dari presentasinya. Profil pemakai diwujudkan dengan karakteristik sebuah produk dan kemudian diberi angka bobot menurut suatu parameter. Hasil rekomendasi adalah vektor yang memiliki *cosine similarity* tinggi.

2.3.6. Algoritma Term Frequency-Invers Document Frequency

Metode TF-IDF merupakan metode untuk menghitung bobot setiap kata yang paling umum digunakan pada *information retrieval*. Metode ini juga terkenal efisien, mudah dan memiliki hasil yang akurat. Metode ini akan menghitung nilai *Term Frequency* (TF) dan *Inverse Document Frequency* (IDF) pada setiap token (kata) di setiap dokumen dalam korpus (Robertson,2005). Metode ini akan menghitung bobot setiap token t di dokumen d dengan rumus:

$$W_{dt} = tf_{dt} * IDF_t \quad (5)$$

Dimana :

d : dokumen ke- d

t : kata ke- t dari kata kunci

W : bobot dokumen ke- d terhadap kata ke- t

tf : banyaknya kata yang dicari pada sebuah dokumen

Setelah bobot (W) masing-masing dokumen diketahui, maka dilakukan proses pengurutan dimana semakin besar nilai W , semakin besar tingkat similaritas dokumen tersebut terhadap kata kunci, demikian sebaliknya. Algoritma *Term Frequency-Invers Dokument Frequency* TF-IDF akan membangun profil berdasarkan bobot vektor pada komponen pembentuknya. Berdasarkan preferensi pengguna maka sistem akan memperkirakan penilaian suka atau tidak suatu produk berdasarkan analisis kemiripan dari profil pengguna untuk mencapai tujuan yang telah ditetapkan. Nilai IDF dapat dihitung berdasarkan rumus:

$$(idf_i)^n = \log\left(\frac{n}{idf_i}\right) \quad (6)$$

Dimana :

n = jumlah semua produk

df = n jumlah produk yang memiliki term i .

Metode *Term Frequency-Inverse Document Frequency* (TF-IDF) adalah cara pemberian bobot hubungan suatu kata (*term*) terhadap dokumen. TF-IDF ini adalah sebuah ukuran statistik yang digunakan untuk mengevaluasi seberapa penting sebuah kata di dalam sebuah dokumen atau dalam sekelompok kata. Untuk dokumen tunggal tiap kalimat dianggap sebagai dokumen.

Frekuensi kemunculan kata di dalam dokumen yang diberikan menunjukkan seberapa penting kata itu di dalam dokumen tersebut. Frekuensi dokumen yang mengandung kata tersebut menunjukkan seberapa umum kata tersebut. Bobot kata semakin besar jika sering muncul dalam suatu dokumen dan semakin kecil jika muncul dalam banyak dokumen.

2.3.7. Algoritma *Cosine Similarity*

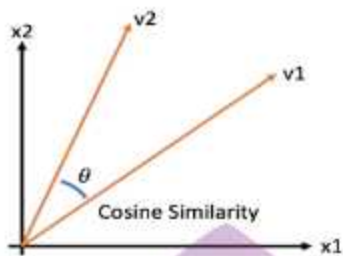
Cosine similarity merupakan metode yang bekerja mengukur jarak kedekatan antara dua buah dokumen dan berfungsi untuk membandingkan kemiripan antar dokumen, dalam penelitian ini yang dibandingkan adalah *query* dengan dokumen latih. Dalam menghitung *cosine similarity*, nilai skalar antara *query* dengan dokumen dijumlahkan. Hasil penjumlahan tersebut akan dikalikan antara panjang dokumen dengan panjang *query* yang telah dikuadratkan, setelah itu di hitung akar pangkat dua. Selanjutnya hasil perkalian skalar tersebut di bagi dengan hasil perkalian panjang dokumen dan *query*. *Cosine similarity* dapat dilihat pada persamaan 7 berikut :

$$similarity = \cos(\theta) = \frac{A \cdot B}{\|A\| \|B\|} = \frac{\sum_{i=1}^n A_i B_i}{\sqrt{\sum_{i=1}^n A_i^2} \sqrt{\sum_{i=1}^n B_i^2}} \quad (7)$$

dimana :

- A_i : term ke- i dalam vektor dokumen A
- B_i : term ke- i dalam vektor dokumen B
- n : jumlah *term* unik dalam dataset

Dalam implementasinya, *cosine similarity* dianalogikan sebagai sudut antar garis vektor seperti gambar 2.1 berikut :

Gambar 2.1 Vektor *Cosine Similarity*

2.3.8. Recall, Precision dan F1-Score

Recall menurut Lancaster dalam Pendit adalah proporsi jumlah dokumen yang dapat ditemukan kembali oleh sebuah proses pencarian informasi (Pendit, 2008) sedangkan menurut Mihalcea, *recall* adalah kemampuan pencarian untuk menemukan semua item yang relevan dalam *corpus*. Dalam klasifikasi biner, *recall* sering disebut sensitivitas. Jadi dapat dilihat sebagai probabilitas bahwa dokumen yang relevan diambil oleh query. Sangat mudah untuk mencapai nilai *retrieval* 100% dengan mengembalikan semua dokumen sebagai tanggapan atas permintaan apa pun. Untuk mengukur *recall* dapat menggunakan persamaan 8 sebagai berikut:

$$recall = \frac{\text{jumlah dokumen relevan yang terpanggil}}{\text{jumlah dokumen relevan yang ada pada database}} \quad (8)$$

Precision merupakan sebuah ukuran yang mengukur tingkat proporsi jumlah dokumen yang dapat ditemukan kembali oleh sebuah proses pencarian dan dianggap relevan untuk kebutuhan pencarian informasi atau rasio jumlah dokumen relevan yang ditemukan dengan total jumlah dokumen yang ditemukan (Pendit, 2008). Mihalcea mendefinisikan *precision* sebagai kemampuan untuk mengambil dokumen peringkat teratas yang sebagian besar relevan. Dalam klasifikasi biner, fall-out terkait erat dengan spesifisitas. Ini dapat dilihat sebagai probabilitas bahwa dokumen yang tidak relevan diambil oleh kueri. Untuk mengukur *precision* dapat diukur dengan menggunakan persamaan 9 sebagai berikut:

$$precision = \frac{\text{jumlah dokumen relevan yang terpanggil}}{\text{jumlah dokumen yang terpanggil}} \quad (9)$$

Perhitungan *recall* dan *precision* pada *Information Retrieval Evaluation* oleh Mihalcea dicontohkan dengan 6 dokumen relevan pada tabel 2.6 berikut :

Tabel 2.6 *Recall* dan *Precision* pada *Information Retrieval Evaluation*

n	Doc #	Relevan	Recall	Precision
1	F	Relevan	1/6 = 0.167	1/1 = 1
2	Y	Tidak Relevan		
3	H	Relevan	2/6 = 0.33	2/3 = 0.67
4	M	Relevan	3/6 = 0.5	3/4 = 0.75
5	U	Tidak relevan		
6	W	Relevan	4/6 = 0.67	4/6 = 0.67
7	D	Tidak Relevan		
8	A	Relevan	5/6 = 0.83	5/8 = 0.625
9	C	Relevan	6/6 = 1	6/9 = 0.67
10	I	Tidak Relevan		
Rata-rata <i>Recall</i> dan <i>Precision</i>			3.497 / 6 = 0.582	4.385 / 6 = 0.730

F1-score merupakan perbandingan rata-rata presisi dan recall yang dibobotkan dengan persamaan 10 seperti berikut :

$$F = 2 * \frac{\text{recall} * \text{precision}}{\text{recall} + \text{precision}} \quad (10)$$

Lancaster memberikan penilaian untuk mengukur tingkat precision pada penilaian efektivitas suatu sistem temu kembali informasi dengan ukuran angka dan mengkategorikannya menjadi precision rendah yaitu tidak efektif, precision sedang berarti kurang efektif, dan precision tinggi yang berarti efektif (Lancaster, 1979). Berikut tabel penilaiannya:

Tabel 2.7 Penilaian efektivitas *Precision* pada *Information Retrieval System*

Range	Efektifitas
0 – 0.33	Precision rendah, Tidak Efektif
0.34 – 0.67	Precision sedang, Kurang Efektif
0.68 – 1.00	Precision tinggi, Efektif

2.3.9. Mean Average Precision

Mean Average Precision (MAP) adalah salah satu teknik evaluasi pada sistem *Information Retrieval*. Teknik ini menggunakan nilai *precision* rata-rata yang diperoleh dari dokumen k teratas setiap kali suatu dokumen yang relevan diperoleh. MAP terbukti memiliki diskriminasi yang sangat baik dan stabil. Jika sekumpulan dokumen yang relevan untuk kebutuhan $q_j \in Q$ adalah $\{d_1, \dots, d_m\}$ dan R_{jk} adalah kumpulan peringkat yang terambil dari hasil tertinggi sampai mendokumentasikan d_k , maka

$$MAP(Q) = \frac{1}{|Q|} \sum_{j=1}^{|Q|} \frac{1}{m_j} \sum_{k=1}^{m_j} Precision(R_{jk}) \quad (11)$$

Jika dokumen yang relevan tidak terambil sama sekali, nilai *precision* pada persamaan diatas diambil menjadi 0. Untuk kebutuhan informasi tunggal, rata-rata *precision* kira-kira daerah dibawah kurva *uninterpolated precision-recall*, sehingga MAP secara kasar dapat dikatakan bahwa daerah rata-ratanya terletak di bawah kurva *precision-recall* untuk sekumpulan *query*. Dengan menggunakan MAP, *recall levels* tetap tidak terpilih dan tidak ada interpolasi. Nilai MAP untuk koleksi pengujian adalah rata-rata aritmatik dari nilai rata-rata *precision* untuk kebutuhan informasi individual. Menghitung nilai MAP secara normal sangat bervariasi disamping kebutuhan informasi individual. Sekumpulan kebutuhan informasi pengujian harus besar dan cukup beragam untuk merepresentasikan keefektifan dari sistem menggunakan *query* yang berbeda.

Contoh MAP :

Terdapat 10 buah dokumen dengan kondisi relevansi sebagai berikut :

R - R - N - R - N - R - N - N - R - N

dimana R adalah dokumen relevan N adalah dokumen tidak relevan. Dari 10 dokumen tersebut terdapat 5 dokumen yang relevan, sehingga $|Q| = 5$. Berdasarkan rumus MAP, maka MAP yang diperoleh adalah sebagai berikut :

$$MAP(Q) = \frac{1}{5} \left(\frac{1}{1} + \frac{2}{2} + \frac{3}{4} + \frac{4}{6} + \frac{5}{9} \right) = 0.749$$

BAB III METODE PENELITIAN

3.1. Jenis, Sifat dan Pendekatan Penelitian

Jenis, sifat dan pendekatan pada yang akan dilaksanakan pada penelitian ini adalah sebagai berikut :

a. Jenis Penelitian Eksperimen

Penelitian ini dilakukan untuk memperoleh perbandingan bobot rata-rata kandungan bahan makanan dan nilai relevansi penerapan algoritma *Boyer-Moore* pada *content base filtering* terhadap data resep masakan.

b. Sifat Penelitian Deskriptif

Penelitian ini menghitung rata-rata bobot bahan makanan yang dihasilkan serta membandingkan nilai relevansi hasil rekomendasi *Content Based Filtering* sebelum dan sesudah penerapan algoritma *Boyer-Moore* agar diperoleh kesimpulan penelitian.

c. Pendekatan Kuantitatif

Hasil penelitian berupa nilai yang menunjukkan bobot rata-rata komposisi dan tingkat relevansi hasil rekomendasi.

Penelitian ini memiliki pendekatan kuantitatif dimana peneliti mengambil langkah eksperimen untuk memperoleh suatu metode pemberian saran resep masakan dengan komposisi sesuai dengan bahan makanan yang terkandung didalamnya. Peneliti akan menganalisa dataset resep masakan yang diolah dengan metode *Content-Based Filtering* dengan penambahan algoritma *Boyer-Moore* untuk mendapatkan tingkat relevansi yang lebih tinggi. Peneliti akan menguji model rekomendasi yang dibuat untuk diperoleh fakta-fakta ilmiah yang sesuai dengan rumusan masalah dan tujuan penelitian.

3.2. Metode Pengumpulan Data

Metode pengumpulan data dilakukan dengan cara observasi. Observasi dilakukan dengan pengamatan langsung terhadap data yang akan diteliti terkait permasalahan yang akan dibahas yaitu dataset resep masakan. Dalam penelitian ini,

penulis memakai data resep masakan indonesia dari pusat data Kaggle. Berikut adalah data yang penulis unduh :

1. Data resep masakan <https://www.kaggle.com/canggih/indonesian-food-recipes> pada tanggal 23 Maret 2021 yang terdapat sekitar 14000 data.
2. Data stoplist bahasa indonesia dari kaggle yang penulis diunduh dari url <https://www.kaggle.com/oswinrh/indonesian-stoplist> pada tanggal 12 Agustus 2021 terdapat 756 daftar stoplist.
3. Dataset bahan masakan diperoleh dari ekstraksi dan translasi resep masakan pada data resep berbahasa inggris di url <https://www.kaggle.com/kaggle/recipe-ingredients-dataset> diunduh pada bulan Agustus 2021 yang terdapat 335 data training dan 360 data testing.

3.3. Analisa Data

Metode analisa data yang akan dilakukan adalah

1. Preprocessing

Tahapan preprocessing yang akan digunakan adalah :

a. Case Folding

Tidak semua dokumen teks konsisten dalam penggunaan huruf kapital. Oleh karena itu, peran *case folding* dibutuhkan dalam mengkonversi keseluruhan teks dalam dokumen menjadi suatu bentuk standar (biasanya huruf kecil atau *lowercase*).

b. Tokenizing

Tokenisasi secara garis besar memecah sekumpulan karakter dalam suatu teks ke dalam satuan kata, bagaimana membedakan karakter-karakter tertentu yang dapat diperlakukan sebagai pemisah kata.

c. Filtering

Dalam penelitian ini, *stopword* yang digunakan adalah pustaka sastrawi yang banyak digunakan oleh sistem rekomendasi berbahasa indonesia. Data resep masakan yang memiliki bentuk text dan kata yang heterogen banyak memuat unsur singkatan, ukuran, berat serta *volume* untuk

mendesripsikan kebutuhan resep masakan. Pada tahap *filtering* ini, data ukuran, singkatan, berat serta *volume* pada item resep masakan akan dihilangkan.

d. *Stemming*

Metode *stemming* yang dilakukan pada penelitian ini tetap menggunakan pustaka dari sastrawi. Tujuan dari *stemming* pada penelitian ini adalah mentransformasikan kata menjadi kata dasarnya (*root*), menghilangkan semua imbuhan kata (*affixes*) meliputi awalan kata (*prefixes*), sisipin kata (*infixes*), akhiran kata (*suffixes*) dan atau menghilangkan awalan dan akhiran kata (*confixes*) pada kata turunan.

2. Kemiripan Dokumen

Data akan diolah algoritma TF-IDF untuk mendapatkan vektor resep masakan dan menghasilkan jarak terdekat dari *query*. Pada beberapa literasi, vektor yang di bentuk oleh TF-IDF masih memiliki tingkat akurasi yang rendah untuk merekomendasikan dokumen terhadap *query* yang diinputkan. Algoritma *cosine similarity* digunakan untuk meningkatkan akurasi dan menutupi kekurangan vektor yang dihasilkan TF-IDF. Vektor yang dibentuk oleh algoritma TF-IDF akan digunakan pada persamaan *cosine similarity* untuk mendapatkan nilai kemiripan dan kedekatan antar vektor.

3. Pembobotan

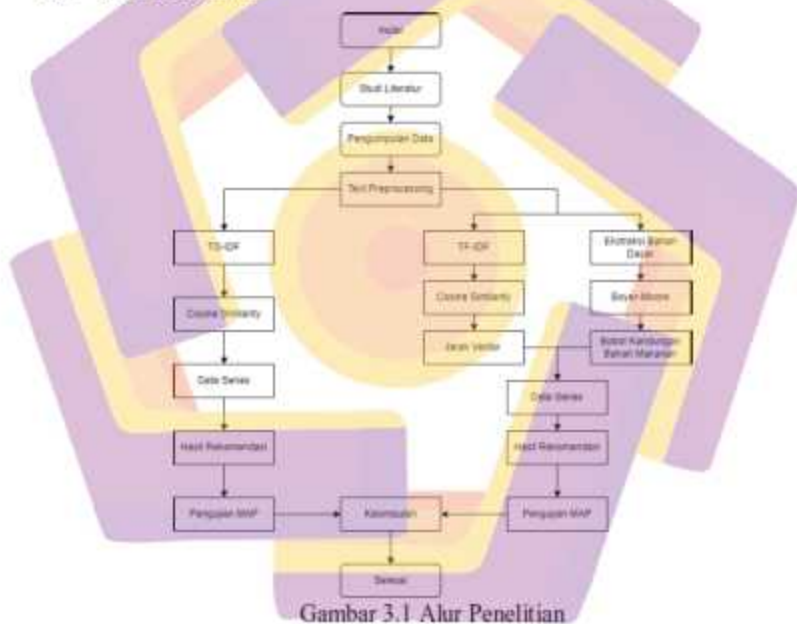
Dalam kasus resep masakan, item sebuah bahan masakan tidak dapat dilepaskan dari kombinasi kata sebagai kata kunci. Algoritma *Boyer-Moore* digunakan untuk menemukan *pattern* kombinasi kata dalam sebuah dokumen. Dalam penelitian ini, hasil perhitungan algoritma *Boyer-Moore* digunakan untuk menghitung bobot kandungan kata kunci pada dokumen tersebut.

3.4. Pengujian

Pada penelitian ini pengukuran rata-rata bobot kandungan bahan makanan dan relevansi penggunaan algoritma *Boyer-Moore* terhadap rekomendasi resep masakan dengan pendekatan *Content Based Filtering* adalah dengan menggunakan pengujian *Mean Average Precision (MAP)* untuk *Information Retrieval*. Pengujian

ini dilakukan untuk mengetahui perbandingan rata-rata bobot kandungan bahan makanan dan nilai relevansi sebelum dan setelah menggunakan algoritma *Boyer-Moore*. Pengukuran *Recall*, *Precision*, *F1-score* dan MAP dilakukan dengan menggunakan sampel query sebanyak 10 (sepuluh). Setiap sampel *query* akan dimasukkan kedalam tabulasi terhadap dokumen rekomendasi, sehingga dapat ditarik kesimpulan seberapa tinggi peningkatan rata-rata bobot kandungan bahan makanan dan nilai relevansi sistem rekomendasi berbasis konten pada data resep masakan sebelum dan sesudah penerapan algoritma *Boyer-Moore*.

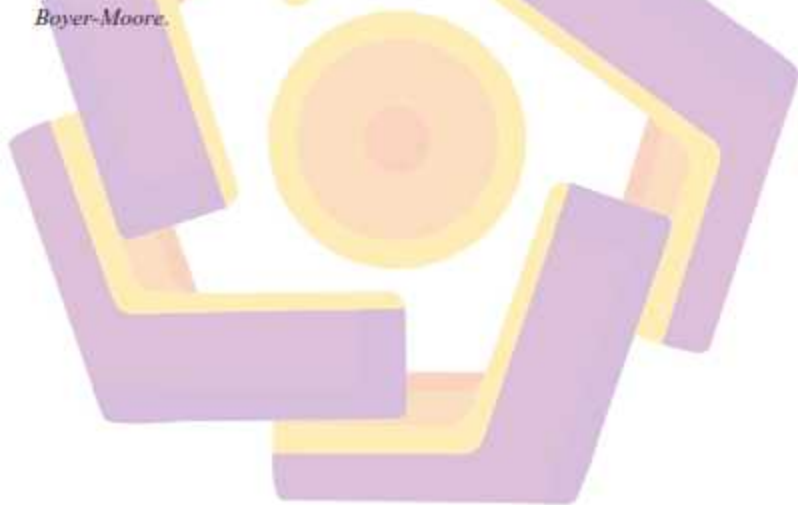
3.5. Alur Penelitian



Gambar 3.1 Alur Penelitian

Penelitian ini dimulai dari tahap studi literatur, dimana dalam tahap ini dilakukan identifikasi masalah dengan menganalisa latar belakang masalah, menentukan arah tujuan penelitian, merumuskan masalah dan kebermanfaatannya. Pustaka yang terkait dengan masalah yang akan diteliti akan dicari dan dipelajari untuk menjadi sumber referensi penelitian.

Metode pengumpulan data dilakukan dengan cara observasi. Observasi dilakukan dengan pengamatan langsung terhadap data yang akan diteliti terkait permasalahan yang akan dibahas yaitu dataset resep masakan. Data akan melalui preprocessing dengan algoritma *case folding*, *tokenizing*, *filtering* dan *stemming* untuk menentukan index dari setiap dokumen. Setiap data akan diberikan bobot oleh TF-IDF dan diolah untuk mencari nilai *similarity*. Perangkingan hasil *cosine similarity* akan dibandingkan tingkat akurasi antara sebelum dan sesudah dikalikan dengan hasil pembobotan *Boyer-Moore*. Setelah tahapan penelitian tersebut diselesaikan, maka dilakukan pengujian *Recall*, *Precision*, *F1-score* dan *Mean Average Precision (MAP)* untuk dapat disimpulkan seberapa tinggi nilai rata-rata bobot kandungan bahan makanan dari setiap hasil rekomendasi dan seberapa tinggi peningkatan nilai relevansi rekomendasi resep masakan dengan algoritma *Boyer-Moore*.



BAB IV HASIL PENELITIAN DAN PEMBAHASAN

4.1. Pengumpulan Data

Hasil yang diperoleh pada penelitian ini adalah sebuah model rekomendasi resep masakan yang menggabungkan kedekatan jarak vektor dengan bobot bahan makanan sebagai obyek, sehingga direpresentasikan bahwa resep masakan yang direkomendasikan memiliki komposisi bahan makanan yang sesuai. Peneliti menggunakan dataset bahan makanan indonesia, resep masakan indonesia dan *stopword* bahasa indonesia dari kaggle. Data resep masakan dengan format CSV yang terdiri dari 8 file dengan kategori yang berbeda dengan total puluhan ribu baris data. Pada penelitian ini, penulis hanya menggunakan 570 data resep masakan pilihan yang diambil dari setiap file kategori resep masakan sebagai dataset. Dataset yang telah disiapkan tersebut disimpan dalam bentuk CSV baru dan diimport kedalam database dengan nama "tesis_bm" pada tabel "resep". Berikut adalah hasil import dataset resep masakan yang digunakan dalam penelitian ini dapat dilihat pada gambar 4.1 berikut:



id_resep	nama_resep	ingredients
1	1. Ayam Goreng	1. Ayam, 2. Bawang Putih, 3. Bawang Merah, 4. Cabe Merah, 5. Cabe Hijau, 6. Gula Merah, 7. Gula Putih, 8. Minyak Goreng, 9. Air, 10. Garam
2	2. Ayam Bakar	1. Ayam, 2. Bawang Putih, 3. Bawang Merah, 4. Cabe Merah, 5. Cabe Hijau, 6. Gula Merah, 7. Gula Putih, 8. Minyak Goreng, 9. Air, 10. Garam
3	3. Ayam Bakar	1. Ayam, 2. Bawang Putih, 3. Bawang Merah, 4. Cabe Merah, 5. Cabe Hijau, 6. Gula Merah, 7. Gula Putih, 8. Minyak Goreng, 9. Air, 10. Garam
4	4. Ayam Bakar	1. Ayam, 2. Bawang Putih, 3. Bawang Merah, 4. Cabe Merah, 5. Cabe Hijau, 6. Gula Merah, 7. Gula Putih, 8. Minyak Goreng, 9. Air, 10. Garam
5	5. Ayam Bakar	1. Ayam, 2. Bawang Putih, 3. Bawang Merah, 4. Cabe Merah, 5. Cabe Hijau, 6. Gula Merah, 7. Gula Putih, 8. Minyak Goreng, 9. Air, 10. Garam
6	6. Ayam Bakar	1. Ayam, 2. Bawang Putih, 3. Bawang Merah, 4. Cabe Merah, 5. Cabe Hijau, 6. Gula Merah, 7. Gula Putih, 8. Minyak Goreng, 9. Air, 10. Garam
7	7. Ayam Bakar	1. Ayam, 2. Bawang Putih, 3. Bawang Merah, 4. Cabe Merah, 5. Cabe Hijau, 6. Gula Merah, 7. Gula Putih, 8. Minyak Goreng, 9. Air, 10. Garam
8	8. Ayam Bakar	1. Ayam, 2. Bawang Putih, 3. Bawang Merah, 4. Cabe Merah, 5. Cabe Hijau, 6. Gula Merah, 7. Gula Putih, 8. Minyak Goreng, 9. Air, 10. Garam
9	9. Ayam Bakar	1. Ayam, 2. Bawang Putih, 3. Bawang Merah, 4. Cabe Merah, 5. Cabe Hijau, 6. Gula Merah, 7. Gula Putih, 8. Minyak Goreng, 9. Air, 10. Garam
10	10. Ayam Bakar	1. Ayam, 2. Bawang Putih, 3. Bawang Merah, 4. Cabe Merah, 5. Cabe Hijau, 6. Gula Merah, 7. Gula Putih, 8. Minyak Goreng, 9. Air, 10. Garam

Gambar 4.1 Tabel Resep Masakan

Dataset bahan makanan didapatkan dari kaggle berupa data dengan format JSON dari resep masakan berbahasa inggris namun memiliki format bahan makanan yang tersusun rapi pada baris *array* sehingga memudahkan untuk melakukan ekstraksi. Untuk menjadikan dataset tersebut menjadi dataset bahan makanan, peneliti menempuh langkah untuk *translate* kedalam bahasa indonesia

dengan menggunakan bantuan Google Translate. Selanjutnya, hasil *translate* diekstraksi dan dimasukkan ke dalam tabel “bahan” pada database “tesis_bm” dengan format kolom *unique* agar tidak terjadi redundansi data. Data bahan makanan dengan jumlah sebanyak 3748 *record* ditunjukkan pada gambar 4.2 berikut :



Gambar 4.2 Tabel Bahan Makanan

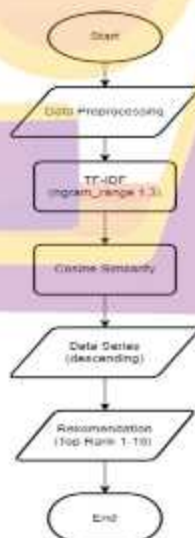
4.2. Skenario Model Rekomendasi

Dalam penelitian ini, penerapan Algoritma *Boyer-Moore* dapat digunakan untuk membobotkan dan juga dapat digunakan untuk ekstraksi bahan makanan yang terkandung. Bahan makanan yang dimiliki pada dataset mayoritas memiliki gabungan 2 suku kata dan selebihnya berupa satu suku kata dan tiga suku kata. Sehingga untuk mendapatkan hasil terbaik dari model yang diusulkan, penulis melakukan 5 skenario penelitian dengan spesifikasi model sebagai berikut :

1. Data diolah menggunakan pendekatan *Content Based Filtering* standar dengan algoritma *TF_IDF* dan *Cosine Similarity* yang ada pada library *NLTK*. Hasil uji metode ini akan menjadi tolok ukur dari skenario penelitian yang dilakukan.
2. Pendekatan *Content Based Filtering* dan *Boyer-Moore* dengan nilai *ngram_range* 1,2 dengan ekstraksi bahan dasar hanya pada query.
3. Pendekatan *Content Based Filtering* dan *Boyer-Moore* dengan nilai *ngram_range* 1,3 dengan ekstraksi bahan dasar hanya pada query.

4. Pendekatan *Content Based Filtering* dan *Boyer-Moore* dengan nilai *ngram_range* 1,2 dengan ekstraksi bahan dasar pada query dan data koleksi.
5. Pendekatan *Content Based Filtering* dan *Boyer-Moore* dengan nilai *ngram_range* 1,3 dengan ekstraksi bahan dasar pada query dan data koleksi.

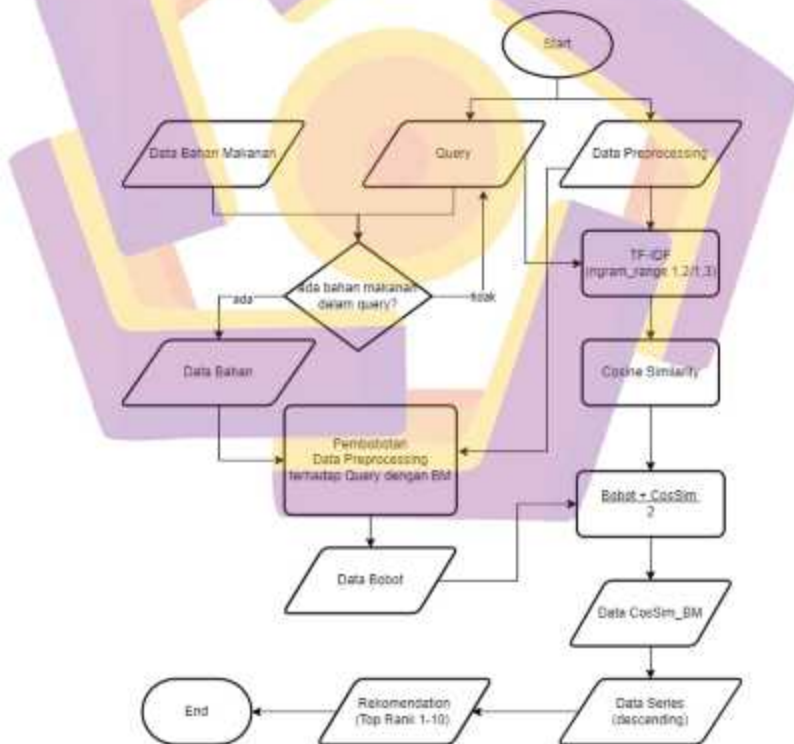
Pada skenario pertama, alur pertama adalah Algoritma TF-IDF mengekstrak *Term Frequency* dari dataset resep masakan yang telah melalui *text preprocessing*. Hasil ekstraksi data *Term Frequency* diolah untuk menentukan nilai *Inverse Document Frequency* setiap resep yang ada pada koleksi. *Cosine Similarity* akan menghitung kedekatan dari setiap data pada koleksi berdasarkan nilai TF-IDF sebelumnya. Hasil perhitungan *Cosine Similarity* tersebut akan menghasilkan *matrix* kedekatan setiap data pada koleksi. Data *matrix* diakses untuk membuat *dataseries* dengan pengurutan nilai dari terbesar ke terkecil dan diambil 10 urutan teratas sebagai hasil rekomendasi. Alur pada skenario pertama dapat dilihat pada gambar 4.3 berikut :



Gambar 4.3 Skenario Model Pertama

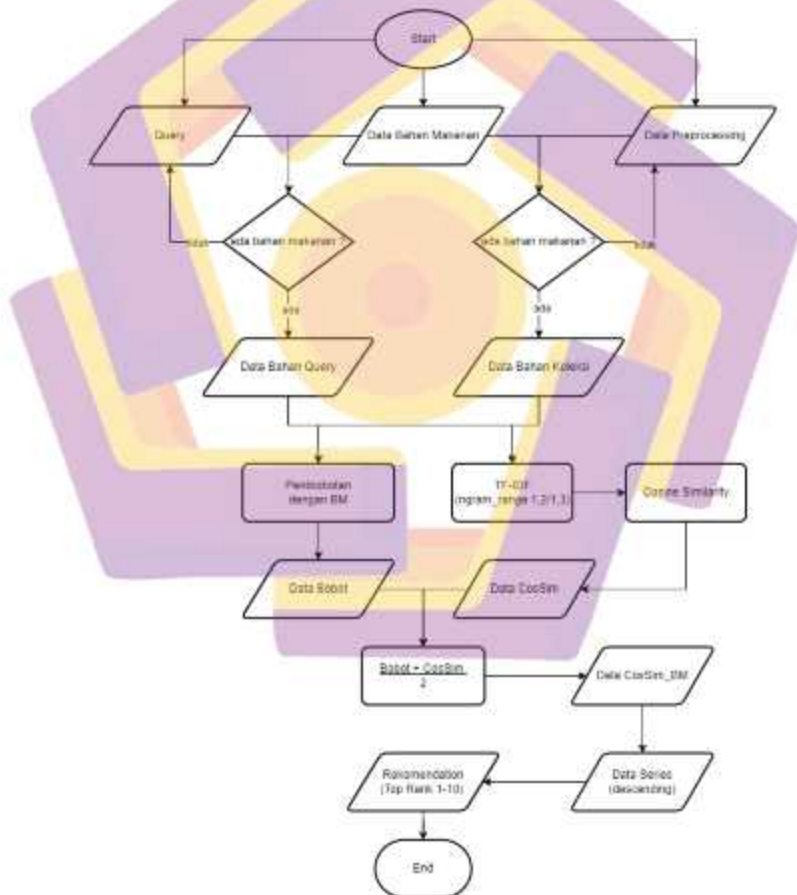
Dalam penerapan algoritma *Boyer-Moore* penulis memiliki beberapa skenario penelitian untuk mendapatkan hasil yang maksimal.

Pada skenario kedua dan ketiga, data hasil preprocessing diolah dengan membandingkan kemiripan antara data query dan data koleksi. Model rekomendasi pada skenario memiliki langkah ekstraksi bahan makanan hanya pada data query. Data bahan makanan hasil ekstraksi digunakan untuk mencari nilai bobot bahan pada setiap dokumen koleksi. Hasil perhitungan kemiripan ditambah dengan bobot dibagi 2 sebagai hasil pengolahan dan menjadi nilai untuk menentukan urutan pada data series yang diurutkan secara *descending*. Pada skenario kedua dan ketiga, yang membedakan adalah nilai "*ngram_range*" pada TF-IDF. Berikut gambar 4.4 adalah alur skenario kedua dan ketiga :



Gambar 4.4 Skenario Model Kedua dan Ketiga

Pada skenario keempat dan kelima, data query dan data koleksi dilakukan ekstraksi bahan makanan. Hasil ekstraksi akan diolah untuk menghitung bobot dan kemiripan. Hasil perhitungan kemiripan ditambah dengan bobot dibagi 2 sebagai hasil pengolahan dan menjadi nilai untuk menentukan urutan pada data series yang diurutkan secara *descending*. Pada skenario keempat dan kelima memiliki perbedaan pada nilai *ngram_range* pada TF-IDF, dimana skenario keempat menggunakan nilai *ngram_range* 1,2 dan skenario kelima menggunakan nilai *ngram_range* 1,3. Berikut ini gambar 4.5 adalah alur skenario keempat dan kelima:



Gambar 4.5 Skenario Model Keempat dan Kelima

Setiap skenario model rekomendasi memiliki beberapa perbedaan, diantaranya dapat dilihat pada tabel 4.1 berikut :

Tabel 4.1 Perbedaan Skenario Model Rekomendasi

Skenario	Algoritma	Ekstraksi Bahan Makanan	TF-IDF ngram	Target Keluaran
1	CBF	Tidak ada	Monogram	Hasil rekomendasi berdasarkan kemiripan dokumen query dan data koleksi
2	CBF + BM	Query	Bigram	Hasil rekomendasi berdasarkan kemiripan ekstraksi bahan makanan pada dokumen query dengan data koleksi menggunakan <i>bigram</i>
3	CBF + BM	Query	Trigram	Hasil rekomendasi berdasarkan kemiripan ekstraksi bahan makanan pada dokumen query dengan data koleksi menggunakan <i>trigram</i>
4	CBF + BM	Query dan Data Koleksi	Bigram	Dokumen query dan data koleksi sama-sama dilakukan ekstraksi bahan makanan untuk dicari kemiripan bahan makanannya dengan <i>bigram</i>
5	CBF + BM	Query dan Data Koleksi	Trigram	Dokumen query dan data koleksi sama-sama dilakukan ekstraksi bahan makanan untuk dicari kemiripan bahan makanannya dengan <i>trigram</i>

Pada penelitian *Content Based Filtering* ini, peneliti menggunakan bahasa pemrograman Python dengan *console jupyter* dan data berada pada database MySQL. Versi Python yang digunakan yaitu 3.8.5 dengan jupyter notebook versi 6.2.0 dan MySQL 5.0.12-dev yang running diatas framework XAMPP 7.1.6. Akses database localhost dengan engine PHPMyAdmin menggunakan browser Chrome.

Dalam penulisan model program rekomendasi, penulis menggunakan beberapa *library* yaitu :

1. *Pymysql* yang digunakan untuk menghubungkan Python dengan database MySQL.
2. *Pandas* digunakan untuk membentuk dataframe.
3. *String* adalah modul yang digunakan untuk mengolah string.
4. *Re* adalah modul library regular expression yang digunakan untuk mengolah karakter selain text.
5. *NLTK* adalah *Natural Language Tool Kit* yang bekerja untuk memodelkan teks atau *text preprocessing* sebelum digunakan pada *machine learning* atau algoritma *deep learning*.
6. *Numpy* adalah library python yang biasa digunakan untuk komputasi dengan tipe data numerik.
7. *Sklearn* atau *scikit learn* adalah library python yang membantu dalam melakukan *processing* data ataupun melakukan *training* data untuk kebutuhan *machine learning*.

Penulis menuliskan kode program dengan menggunakan *jupyter notebook*. Pada header program, penulis mengimport *library* *pymysql*, *pandas*, *string*, *re* dan *nlk*. Pada baris berikutnya adalah kode fungsi program dengan nama “*connection*” untuk menghubungkan ke database “*tesis_bm*” dengan modul *connect* milik *library* *pymysql* dapat dilihat pada gambar 4.6.

```

In [17]: import pymysql.cursor
import pandas as pd
from pandas import DataFrame
import string
import re

from nltk.tokenize import word_tokenize

# connect database
connection = pymysql.connect(host='localhost', user='root', password='',
                             database='tesis_bm', charset='utf8mb4',
                             cursorclass=pymysql.cursors.DictCursor)

with connection:
    with connection.cursor() as cursor:
        sql = "SELECT * FROM user"
        cursor.execute(sql)
        result = cursor.fetchall()
        sql_data = pd.DataFrame(result)

    with connection.cursor() as cursor:
        sql = "SELECT word FROM kata"
        cursor.execute(sql)
        result_kata = cursor.fetchall()
        sql_kata = pd.DataFrame(result_kata)

```

Gambar 4.6 Header Program, Koneksi Database dan Select Data

dilihat pada gambar 4.13. Kalimat hasil *text preprocessing* disimpan pada kolom *array* "clean". Pada tahap ini, tahap *text processing* dianggap telah selesai dan dapat dilakukan pengolahan proses rekomendasi.

```

In [14]: def detokenizer():
        return " ".join(tokens)
        #clean_text["clean"] = detokenizer(clean_text["tokens"])
        #print(clean_text["clean"])

0  akan akan memang datang buah jeruk manis gara...
1  akan datang datang buah jeruk manis akan jeruk ...
2  akan buah manis jeruk buah manis akan manis ...
3  agung akan manis jeruk gula jeruk kuning ...
4  bukit akan beraneka kuning kuning putih kuning ...
...
100 kuning kuning putih kuning putih kuning buah ...
101 kuning kuning buah jeruk datang buah akan ...
102 lemon putih kuning putih kuning kuning ...
103 teh kuning kuning ...
104 kuning kuning buah jeruk manis jeruk ...
105 jeruk jeruk jeruk jeruk jeruk jeruk jeruk ...
106 jeruk jeruk jeruk jeruk jeruk jeruk jeruk ...
107 kuning kuning datang buah jeruk jeruk ...
108 jeruk jeruk jeruk jeruk jeruk jeruk jeruk ...
name: clean, dtype: object

```

Gambar 4.13 *Detokenizer*

Dalam skenario pertama *Term* yang ditemukan pada dataset digambarkan pada tabel "vectorizer" seperti pada pada gambar 4.14 berikut :

	abc	abla	abon	acar	ad	adax	adonan	aduk	aga	agh	...	ahen	wings	woriel	wulu	wuluh	yaa	yach	yo	yyet	zafun	
0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
...
100	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
101	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
102	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
103	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
104	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
105	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
106	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
107	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
108	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
109	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
110	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	

Gambar 4.14 *Term Frequency Pada Program*

4.4. Model Rekomendasi

Pada Skenario 1, teks hasil pengolahan *preprocessing* dihitung kemunculan kata dengan TF-IDF kemudian ditransformasikan kedalam *matrix* dan diolah dengan fungsi *Cosine Similarity*. *Matrix* keluaran dari proses TF-IDF dan *Cosine Similarity* terdapat pada gambar 4.15 berikut :

```

In [102]: import numpy as np
from sklearn.metrics.pairwise import cosine_similarity
from sklearn.feature_extraction.text import TfidfVectorizer

isi_data = pd.read_csv('data', encoding='utf-8')
tf = TfidfVectorizer(analyzer='word', ngram_range=(1, 1), min_df=1)
tfidf_matrix = tf.fit_transform(isi_data['content'])
cos_sim = cosine_similarity(tfidf_matrix, tfidf_matrix)
cos_sim

Out[102]: array([[1.00000000, 0.8692149, 0.862080, ..., 0.8008891, 0.8294999,
0.8017902],
[0.8692149, 1.0000000, ..., 0.8027087, 0.8122084,
0.8017907],
[0.862080, 0.8698761, 1.0000000, ..., 0.8007828, 0.8078888,
0.8017909],
...,
[0.8008891, 0.8027087, 0.8007828, ..., 1.0000000,
0.01940044],
[0.8294999, 0.8122084, 0.8078888, ..., 0.01940044, 1.0000000],
[0.8017902, 0.8017907, 0.8017909, ..., 0.01940044,
0.01940044, 1.0000000],
...])

```

Gambar 4.15 Matrix Cosine Similarity Pada Program

Matrix diolah menjadi data series berdasarkan index data dan diurutkan berdasarkan nilai *Cosine Similarity* yang paling besar ke paling kecil. Data series yang dibuat akan digunakan untuk proses rekomendasi CBF. Pada gambar 4.16 berikut adalah 20 data index teratas dari *Matrix Cosine Similarity*.

```

In [103]: pd.Series(cos_sim.index)
Out[103]:
4      Ayam atau bebek
1      Ayam goreng tulang lunak
3      Ayam masak bumbu
2      Ayam goreng
6      Ayam goreng bumbu
8      Ayam masak bumbu
4      Ayam masak bumbu
7      Ayam masak bumbu
9      Ayam masak bumbu
10     Ayam masak bumbu
11     Ayam masak bumbu
12     Ayam masak bumbu
13     Ayam masak bumbu
14     Ayam masak bumbu
15     Ayam masak bumbu
16     Ayam masak bumbu
17     Ayam masak bumbu
18     Ayam masak bumbu
19     Ayam masak bumbu
Name: title, dtype: object

```

Gambar 4.16 Data Series Untuk Rekomendasi Model Skenario 1

Data series yang telah dibuat digunakan di dalam fungsi "*recom_CBF_skenario_1*" dimana fungsi ini akan menampilkan 10 rekomendasi teratas. Pada fungsi "*recom_CBF_skenario_1*" di gambar 4.17, variabel "*recommended_resep*" adalah *array* yang digunakan untuk menampilkan data resep yang direkomendasikan diambil dari skor pada data series yang telah disortir.

```

In [124]: def recom_CBF_skenario_1(isi_data, cos_sim = cos_sim):
recommended_resep = []
# Mengambil nama resep berdasarkan skor index
isi = pd.Series(isi_data['content'])
# Mengurutkan skor berdasarkan isi_data
score_series = pd.Series(cos_sim).sort_values(ascending = False)
recom_CBF_skenario_1 = []
# Mengambil 10 resep skor teratas
top_10_resep = list(score_series.index[0:10])
for i in top_10_resep:
recommended_resep.append(isi_data[isi_data.index[i]])
return recommended_resep

```

Gambar 4.17 Fungsi Model Rekomendasi Skenario 1

Fungsi tersebut diakses dengan menuliskan nama resep masakan atau ketika diimplementasikan pada sistem dapat mengambil variabel id atau judul resep masakan, sehingga proses yang terjadi pada gambar 4.17 akan menampilkan 10 nama resep masakan lain.

Model berikutnya yaitu penggunaan algoritma *Boyer-Moore* yang digabungkan dengan hasil perhitungan *Cosine Similarity*, dimana lebih mengedepankan pencarian string berdasarkan obyek yaitu bahan masakan. Pada gambar 4.18 adalah fungsi untuk menemukan string *Boyer-Moore*, dimana terdapat dua buah fungsi yaitu *badCharHeuristic* dan *bm_search*. Fungsi *badCharHeuristic* berfungsi untuk menggeser karakter dan mendeteksi kemiripan string, kemudian fungsi *bm_search* berfungsi untuk mencari kemiripan *pattern*.

```

34 [104]: def BM_CHAR = 255

35 def badCharHeuristic(str, size):
36     badChar = [-1]*256
37     for i in range(size):
38         badChar[str[i]] = i
39     return badChar

40 def bm_search(text, pat):
41     m = len(pat)
42     i = len(text)
43     badChar = badCharHeuristic(text, m)
44     k = 0
45     while i >= 0:
46         j = m - 1
47         while text[i+j] == pat[j]:
48             j -= 1
49         if j <= -1:
50             return pat
51         k += (badChar[text[i+j]]+1) if k <= 0 else 0
52     return None
53
54 def main():
55     k = bm_search('Cuba Hitam', 'C')
56     print k

```

Gambar 4.18 Fungsi Pencarian String *Boyer Moore*

```

34 [105]: #Fungsi untuk mendapatkan data resep yang telah melalui tahap preprocessing
35 def get_clean(index):
36     return pd.read_sql(sql_data_base + 'select * from ' + str(index) + ' where recipe = 1')

```

Gambar 4.19 Fungsi Untuk Menuliskan Data Resep Hasil *Preprocessing*

Pada gambar 4.19, fungsi *get_clean* digunakan untuk mengakses kolom data hasil *preprocessing*. Hal ini bertujuan untuk membatasi penggunaan data yang tidak diperlukan sehingga tidak mengakses banyak kolom untuk diolah.

Selanjutnya fungsi “*query_term*” pada gambar 4.20 berfungsi untuk mengekstrak bahan dasar dari masakan yang sedang diakses dengan bantuan fungsi *get_clean* untuk mendapatkan kalimat bersih yang telah melalui proses *text preprocessing*, dicocokkan dengan bahan dasar pada dataset “*sql_bahan*” dengan

fungsi *Boyer-Moore*. Berbeda dengan fungsi “ekstraksibahan” yang langsung mengolah text untuk diekstrak bahan makanannya. Pada tahap ini didapatkan fungsi ekstraksi komposisi bahan makanan dari resep masakan atau text untuk digunakan pada proses pencarian kedekatan dan pembobotan *Boyer-Moore*.

```
In [142]: #fungsi untuk mengextrak bahan makanan yang ada pada resep masakan yang dimasak
def query_harcama():
    bahandasar = []
    for dataresep in sql_bahan('resep'):
        ada = db_search(get_cleanname)(dataresep)
        if ada==dataresep:
            bahandasar.append(dataresep)
    return bahandasar

def ekstraksibahan(text):
    bahandasar = []
    for dataresep in sql_bahan('resep'):
        ada = db_search(get_cleanname)(dataresep)
        if ada==dataresep:
            bahandasar.append(dataresep)
    return bahandasar
```

Gambar 4.20 Fungsi Untuk Mengekstrak Bahan Dasar Resep Masakan

Fungsi selanjutnya yaitu pembobotan *Boyer-Moore*. fungsi *Term* yang sebelumnya telah didapatkan dari fungsi ekstraksi *query_term* digunakan ke dalam fungsi “bobot_bm” pada gambar 4.21 dimana setiap data koleksi akan dihitung kandungan *term* atau bahan masakan di dalamnya dibandingkan dengan panjang kata yang ada pada setiap data koleksi sehingga di dapatkan prosentase *term* yang ada pada data tersebut. Tujuan dari fungsi *bobot_bm* adalah untuk menghitung seberapa banyak bahan makanan terdapat pada setiap data koleksi.

Fungsi “bobot_bahan” memiliki proses yang sama dengan fungsi “bobot_bm” namun berbeda inputan. Fungsi “bobot_bahan” digunakan untuk mengekstraksi bahan dari text yang dimasukkan dengan *Boyer-Moore*, kemudian mengukur bobotnya dengan *Boyer-Moore* berdasarkan term yang dimasukkan. Fungsi “bobot_bahan” digunakan pada skenario model keempat dan kelima.


```

In [14]: #Fungsi rekomendasi berbasis filter menggunakan algoritma based cosine
def reco_3d_1(senarai_s_nama):
    #Fungsi array rekomendasi based cosine
    rekomendasi_3d = []
    #mengambil data dasar movie yang sedang di akses
    term = query_term(senarai_s_nama)
    #mendefinisikan corpus dengan based cosine berdasarkan data dasar movie yang diakses
    def get_3dmm(term):
        #mencari hasil kombinasi data array dengan key based_3d
        #get_data('based_3d'] = get_data('data').apply(get_bahan)
        #definisikan nama dasar array
        data_3d = {}
        #fungsi untuk mengfiltering cosine similarity antara data training dan data testing
        def cosang_3d(text):
            #mendefinisikan dengan based cosine dan mendefinisikan untuk mengkonversi text
            #tittle = {}
            #vectorizer = TfidfVectorizer(analyzer='word', ngram_range=(1, 2), min_df=1)
            #fit = vectorizer.fit_transform([tittle, data_3d])
            #return (fit.get_feature_names()[0], cosang)
        #mendefinisikan hasil cosine similarity ke dalam array dengan key cos_3d
        #get_data('cos_3d'] = get_data('data').apply(cosang_3d)
        #mendefinisikan hasil cosine similarity dengan based cosine array dan di dalam array dengan key cos_3d_1
        #get_data('cos_3d_1'] = get_data('cos_3d'] * get_data('based_3d')
        # Mendefinisikan string berdasarkan nilai cos_3d_1 untuk mendapatkan
        sorted_get_data = get_data.sort_values(by='cos_3d_1', ascending = False)
        # Mendefinisikan array hasil sorting
        top_3d_1 = sorted(sorted_get_data.index)
        # Mengambil index dari 20 basis rekomendasi berbasis
        top_3d_1 = list(top_3d_1.index[1:21])
        for i in top_3d_1:
            rekomendasi_3d.append(list(sorted_get_data.index[1:]))
    return rekomendasi_3d

```

Gambar 4.24 Fungsi Model Rekomendasi Skenario 4

Model Skenario 5 memiliki fungsi yang sama dengan Model Skenario 4, yang membedakan adalah pada nilai *ngram_range*. Model Skenario 5 menggunakan nilai *ngram_range* 1,3 dengan model skenario seperti pada gambar 4.25 sebagai berikut :

```

In [15]: #Fungsi rekomendasi berbasis filter menggunakan algoritma based cosine
def reco_3d_2(senarai_s_nama):
    #Fungsi array rekomendasi based cosine
    rekomendasi_3d = []
    #mengambil data dasar movie yang sedang di akses
    term = query_term(senarai_s_nama)
    #mendefinisikan corpus dengan based cosine berdasarkan data dasar movie yang diakses
    def get_3dmm(term):
        #mencari hasil kombinasi data array dengan key based_3d
        #get_data('based_3d'] = get_data('data').apply(get_bahan)
        #definisikan nama dasar array
        data_3d = {}
        #fungsi untuk mengfiltering cosine similarity antara data training dan data testing
        def cosang_3d(text):
            #mendefinisikan dengan based cosine dan mendefinisikan untuk mengkonversi text
            #tittle = {}
            #vectorizer = TfidfVectorizer(analyzer='word', ngram_range=(1, 3), min_df=1)
            #fit = vectorizer.fit_transform([tittle, data_3d])
            #return (fit.get_feature_names()[0], cosang)
        #mendefinisikan hasil cosine similarity ke dalam array dengan key cos_3d
        #get_data('cos_3d'] = get_data('data').apply(cosang_3d)
        #mendefinisikan hasil cosine similarity dengan based cosine array dan di dalam array dengan key cos_3d_1
        #get_data('cos_3d_1'] = get_data('cos_3d'] * get_data('based_3d')
        # Mendefinisikan string berdasarkan nilai cos_3d_1 untuk mendapatkan
        sorted_get_data = get_data.sort_values(by='cos_3d_1', ascending = False)
        # Mendefinisikan array hasil sorting
        top_3d_1 = sorted(sorted_get_data.index)
        # Mengambil index dari 20 basis rekomendasi berbasis
        top_3d_1 = list(top_3d_1.index[1:21])
        for i in top_3d_1:
            rekomendasi_3d.append(list(sorted_get_data.index[1:]))
    return rekomendasi_3d

```

Gambar 4.25 Fungsi Model Rekomendasi Skenario 5

4.5. Hasil Model Rekomendasi

4.5.1. Skenario 1

Hasil skenario 1 yaitu hasil yang diperoleh dari pendekatan Content Based Filtering. Model rekomendasi 1 sebagai tolok ukur dalam penelitian dengan hasil rekomendasi pada resep “Ayam goreng tulang lunak” pada gambar 4.26 berikut :

```
In [140]: recom_CBF_skenario_1('Ayam goreng tulang lunak')
Out[140]: ['Tongseng kambing dadakan :D',
           'Krengsengan Tulangan Kambing',
           'Gulai Ikan Gurami',
           'Pais Patin Banjar',
           'Soto Ayam',
           'AYAM ungkep',
           'Tongseng Kambing',
           'Ayam goreng serundeng lengkuas',
           'Ayam suwir',
           'Tongseng Kambing Jogja (otentik warisan mertua)']
```

Gambar 4.26 Hasil Model Rekomendasi Skenario 1

Grafik kesamaan bahan makanan skenario 1 yang terdapat pada hasil rekomendasi terdapat pada gambar 4.27 berikut :



Gambar 4.27 Grafik Kesamaan Bahan Makanan Model Rekomendasi Skenario 1

Kandungan bahan makanan diperoleh dari perbandingan antara query dengan hasil rekomendasi. Pada skenario 1, kesamaan bahan makanan tertinggi berada pada resep “Tongseng Kambing Jogja (otentik warisan mertua)” yang

terdapat pada urutan ke-10 rekomendasi. Sedangkan kesamaan terendah terdapat pada resep “Pais Patin Banjar” pada urutan ke-4 hasil rekomendasi.

Pada *library* NLTK, *Content Based Filtering* dengan TF-IDF menggunakan monogram sehingga frekuensi kata yang ditemukan dihitung setiap satu kata. Sebagai contoh sebuah resep memiliki bahan makanan “bawang putih, cabai rawit merah dan putih telur” dengan “bawang merah, cabai merah keriting dan susu putih”. Secara awam dapat dilihat bahwa kedua resep tersebut memiliki bahan makanan yang jauh berbeda, namun sistem TF-IDF monogram akan merepresentasikan kemiripan yang tinggi karena banyak ditemukan kata yang sama di kedua resep tersebut. Jika dilihat dari hasil rekomendasi, model ini belum efektif digunakan pada rekomendasi yang mengacu pada urutan kata atau gabungan kata.

4.5.2. Skenario 2

Hasil skenario 2 yaitu hasil yang diperoleh dari pendekatan Content Based Filtering dengan penerapan Algoritma *Boyer-Moore* pada sisi query. Pada skenario 2, TF-IDF menggunakan rentang *ngram_range* 1,2 yang menghasilkan rekomendasi seperti pada gambar 4.28 berikut.

```
In [141]: recon_CBF_BM_skenario_2('Ayam goreng tulang lunak')
Out[141]: ['Telur Goreng Gurah',
           'Ayam Goreng Lengkuas ala Those',
           'AYAM ungkep',
           'Soto Ayam',
           'Sayap goreng',
           'Ayam bumbu kuning',
           'Gulai Ikan Gurah',
           'Ayam Goreng Lengkuas',
           'Terik daging kambing simpel',
           'Ayam goreng lengkuas kremes']
```

Gambar 4.28 Hasil Model Rekomendasi Skenario 2

Grafik kesamaan bahan makanan skenario 2 yang terdapat pada hasil rekomendasi terdapat pada gambar 4.29 berikut :



Gambar 4.29 Grafik Kesamaan Bahan Makanan Model Rekomendasi Skenario 2

Pada skenario 2, terdapat tiga resep dengan bahan makanan tertinggi yaitu “Ayam goreng lengkuas kremes, Ayam Goreng Lengkuas dan Ayam Goreng Lengkuas ala those”, sedangkan kesamaan terendah terdapat pada resep “lele goreng gurih” yang terdapat pada urutan rekomendasi paling atas. Resep “lele goreng gurih” hanya memiliki 5 bahan yang sama dengan query, namun bahan yang tidak sama hanya 1, sehingga bobot kandungannya tinggi dan sistem merekomendasikan paling atas.

Pada skenario 2, Algoritma *Boyer-Moore* melakukan ekstraksi kepada dokumen *query* sehingga menghasilkan data bahan makanan yang terkandung. Hasil dari ekstraksi tersebut digunakan sebagai *term* oleh algoritma *Boyer-Moore* untuk melakukan ekstraksi kepada data koleksi sehingga diperoleh bobot kandungan bahan makanannya. TF-IDF dengan *bigram* dan cosine similarity pada skenario 2 masih menghitung jarak kedekatan dokumen query dan data koleksi berdasarkan text hasil preprocessing sebelumnya. Hasil pembobotan dan jarak kedekatan akan ditambahkan kemudian dibagi 2 untuk mendapatkan nilai tengahnya sehingga menghasilkan nilai CBF+BM.

4.5.3. Skenario 3

Hasil skenario 3 memiliki kesamaan pada skenario 2 yaitu hasil yang diperoleh dari pendekatan *Content Based Filtering* dengan penerapan Algoritma *Boyer-Moore* pada sisi query. Pada Model Skenario 3, nilai *ngram_range* dinaikkan menjadi 1,3 dengan hasil rekomendasi pada gambar 4.30 berikut :

```
In [142]: recon_CBF_BM_skenario_3('Ayam goreng tulang lunak')
Out[142]: ['Lele goreng Gurih',
           'AYAM unkep',
           'Ayam Goreng Lengkuas ala Those',
           'Sayap goreng',
           'Ayam bumbu kuning',
           'Soto Ayam',
           'Nila Goreng Gurih',
           'Terik daging kambing simpel',
           'Rica-rica ayam kampung',
           'Gulai Ikan Gurami']
```

Gambar 4.30 Hasil Model Rekomendasi Skenario 3

Grafik kesamaan bahan makanan skenario 3 yang terdapat pada hasil rekomendasi terdapat pada gambar 4.31 berikut :



Gambar 4.31 Grafik Kesamaan Bahan Makanan Model Rekomendasi Skenario 3

Pada skenario 3 hasil rekomendasi pada skenario tidak jauh berbeda dengan skenario 2, terdapat dua resep dengan bahan makanan tertinggi yaitu “Rica-rica ayam kampung dan Ayam Goreng Lengkuas ala those”, sedangkan kesamaan terendah terdapat pada resep “lele goreng gurih dan Nila goreng gurih. Pada skenario 3, Algoritma *Boyer-Moore* melakukan ekstraksi seperti pada skenario 2 yaitu kepada dokumen *query*. Hasil dari ekstraksi tersebut digunakan sebagai *term* oleh algoritma *Boyer-Moore* untuk melakukan ekstraksi kepada data koleksi sehingga diperoleh bobot kandungan bahan makanannya. TF-IDF dengan *trigram* dan cosine similarity masih menghitung jarak kedekatan dokumen query dan data koleksi berdasarkan text hasil preprocessing sebelumnya. Hasil pembobotan dan jarak kedekatan akan ditambahkan kemudian dibagi 2 untuk mendapatkan nilai tengahnya sehingga menghasilkan nilai CBF+BM.

4.5.4. Skenario 4

Hasil skenario 4 diperoleh dari pendekatan *Content Based Filtering* dengan penerapan Algoritma *Boyer-Moore* pada sisi query dan sisi data koleksi. Pada Model Skenario 4, nilai *ngram_range* menggunakan rentang 1,2 dengan hasil rekomendasi pada gambar 4.32 berikut :

```
In [143]: recon_CBF_BM_skenario_4('Ayam goreng tulang lunak')
Out[143]: ['Lele Goreng Gurih',
'Ayam Goreng Simple',
'Ayam Goreng lengkuas ala those',
'Ayam pop',
'Ayam Geprek',
'Minyak Ayam',
'Rica-rica ayam kampung',
'Lele bumbu kuning',
'Lele Goreng Renyah',
'Kari Ayam']
```

Gambar 4.32 Hasil Model Rekomendasi Skenario 4

Grafik kesamaan bahan makanan skenario 4 yang terdapat pada hasil rekomendasi terdapat pada gambar 4.33 berikut :



Gambar 4.33 Grafik Kesamaan Bahan Makanan Model Rekomendasi Skenario 4

Pada skenario 4, resep “Kari Ayam” memiliki kesamaan tertinggi dengan *query*, namun memiliki ketidaksamaan tertinggi juga terhadap *query* sehingga bobot kandungan bahan makanan kecil. Sedangkan kesamaan terendah terdapat pada “Ayam Geprek”.

Pada skenario 4, *Content Based Filtering* tidak lagi menghitung kedekatan berdasarkan text hasil preprocessing melainkan data obyek bahan makanan. Dokumen *query* dan setiap data koleksi akan diekstrak bahan makanan yang terkandung di dalamnya. CBF dengan *bigram* digunakan untuk menghitung jarak kedekatan dan BM digunakan untuk membobotkan kandungan bahan makanan. Jarak dan bobot yang diperoleh ditambahkan, kemudian dibagi 2 untuk dicari nilai tengahnya, hasil perhitungan akan dijadikan pengurutan data secara descending.

4.5.5. Skenario 5

Hasil skenario 5 memiliki kesamaan dengan skenario 4 dengan hasil rekomendasi diperoleh dari pendekatan *Content Based Filtering* dengan penerapan Algoritma *Boyer-Moore* pada sisi *query* dan sisi data koleksi. Pada Model Skenario 4, nilai *ngram_range* menggunakan rentang 1,3 dengan hasil rekomendasi pada gambar 4.34 berikut

```
In [144]: recom_CBF_BM_skenario_5('Ayam goreng tulang lunak')
```

```
Out[144]: ['Ayam Goreng Simple',
            'Lele Goreng Gurih',
            'Ayam Geprek',
            'Ayam pop',
            'Ayam Goreng Lengkuas ala Those',
            'Minyak Ayam',
            'Lele Goreng Renyah',
            'Lele bumbu kuning',
            'Rica-rica ayam kampung',
            'xari Ayam']
```

Gambar 4.34 Hasil Model Rekomendasi Skenario 5

Grafik kesamaan bahan makanan skenario 5 yang terdapat pada hasil rekomendasi terdapat pada gambar 4.35 berikut :



Gambar 4.35 Grafik Kesamaan Bahan Makanan Model Rekomendasi Skenario 5

Pada skenario 5, kandungan kesamaan bahan makanan tidak jauh berbeda dengan skenario 4, sehingga belum dapat merepresentasikan perubahan yang signifikan terkait kandungan bahan makanan. Untuk itu perlu adanya pengujian dengan query lain agar didapatkan nilai yang lebih presisi.

Alur yang terjadi pada skenario 5 memiliki kesamaan dengan alur pada penelitian skenario 4 namun memiliki perbedaan pada data rekomendasi yang dihasilkan. Pada skenario 5, CBF menggunakan *trigram* untuk mengolah data

hingga urutan 3 suku kata sedangkan pada skenario 4 hanya menggunakan bigram yang hanya mengolah dari satu hingga dua suku kata.

4.6. Evaluasi Model Rekomendasi

Untuk membuktikan bahwa Algoritma *Boyer-Moore* dapat menaikkan nilai bobot rata-rata kandungan bahan makanan dan relevansi dari hasil rekomendasi, maka diperlukan pengujian. Penelitian ini menggunakan metode pengujian *Recall*, *Precision*, *F1-score*, *MAP (Mean Average Precision)* dan *AWIC (Average Weight Ingredient Contained)* dengan menggunakan 570 data training dan 10 (sepuluh) data testing sebagai Query Pengujian. Setiap hasil rekomendasi akan dinilai relevansinya berdasarkan *keyword* pencarian. Hasil rekomendasi dianggap relevan apabila memenuhi sebagian besar bahan makanan yang ada pada *query* atau kandungan bahan yang tidak sama tidak lebih dari bahan yang sama. Berikut adalah *query* yang digunakan dalam pengujian:

Tabel 4.2 Query Pengujian

Query	Judul	Bahan Dasar
Q1	Ayam goreng tulang lunak	['bawang putih', 'garam', 'daun jeruk', 'minyak goreng', 'serai', 'kemiri', 'ketumbar', 'air', 'ayam', 'jeruk']
Q2	Semur Daging Sapi	['bawang merah', 'bawang putih', 'gula pasir', 'daun jeruk', 'lada', 'jeruk nipis', 'kecap manis', 'royco', 'bawang bombay', 'ladaku', 'daging sapi', 'air', 'gula', 'daging', 'jeruk']
Q3	Sambal telur ayam	['bawang merah', 'bawang putih', 'cabe rawit', 'garam', 'cabe merah', 'telur', 'lada', 'kecap manis', 'gula', 'gula putih']
Q4	Perkedel Tempe	['bawang merah', 'bawang putih', 'garam', 'tempe', 'ketumbar', 'tepung', 'air', 'cabai', 'cabai rawit', 'bumbu halus', 'papa']
Q5	Ayam panggang madu	['bawang merah', 'bawang putih', 'jahe', 'gula merah', 'garam', 'lengkuas', 'cabe keriting', 'udang', 'lada', 'ker', 'ketumbar', 'air', 'gula', 'madu', 'ayam', 'es']
Q6	Nila Pedas Manis	['bawang merah', 'bawang putih', 'cabe rawit', 'gula merah', 'garam', 'daun jeruk', 'ikan nila', 'tomat', 'serai', 'air', 'gula', 'daun salam', 'jeruk', 'ikan', 'nila']
Q7	Kari Gulai Gurame	['garam', 'daun jeruk', 'santan', 'kaldu bubuk', 'jeruk nipis', 'kari', 'serai', 'air', 'cabai', 'gula', 'cabai rawit', 'kayu manis', 'buah kapulaga', 'jeruk', 'ikan', 'kaldu', 'daun jeruk purut']

Tabel 4.2 Query Pengujian (lanjutan)

Query	Judul	Bahan Dasar
Q8	Tongseng Kambing	['bawang merah', 'bawang putih', 'garam', 'daun jeruk', 'santan', 'merica', 'kecap manis', 'daun bawang', 'tomat', 'serai', 'kol', 'ketumbar', 'air', 'gula', 'daun salam', 'daging', 'jeruk', 'kambing']
Q9	Sop iga kambing	['bawang merah', 'bawang putih', 'garam', 'seledri', 'lada', 'daun bawang', 'wortel', 'bawang goreng', 'tomat', 'timun', 'mentimun', 'kol', 'pala', 'air', 'iga', 'acar', 'acar mentimun', 'daun seledri', 'rebus', 'kambing']
Q10	Tongseng daging sapi pedas	['bawang merah', 'bawang putih', 'cabe rawit', 'jahe', 'daun jeruk', 'cabe merah', 'daging sapi', 'tomat', 'kol', 'kemiri', 'daun salam', 'daging', 'bumbu halus', 'jeruk']

4.6.1. Recall, Precision dan F1-score

Pada pengukuran *recall* dan *precision* diperoleh rekapitulasi yang ditabulasikan seperti pada tabel-tabel berikut :

Tabel 4.3 Recall and Precision Skenario 1

S1	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
recall	0,63	0,67	0,6	0,75	0,75	0,56	0,75	0,56	1	0,57
precision	0,43	0,4	0,5	0,42	0,19	0,98	0,625	0,92	1	0,89

Tabel 4.4 Recall and Precision Skenario 2

S2	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
recall	0,56	0,56	0,56	0,6	0,56	0,55	0,6	0,55	0,556	0,55
precision	0,99	0,95	0,95	0,45	0,88	1	0,764	1	1	1

Tabel 4.5 Recall and Precision Skenario 3

S3	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
recall	0,56	0,56	0,56	0,6	0,56	0,55	0,6	0,55	0,55	0,55
precision	0,98	0,96	0,93	0,51	0,88	1	0,943	1	1	1

Tabel 4.6 Recall and Precision Skenario 4

S4	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
recall	0,56	0,55	0,55	0,56	0,56	0,55	0,583	0,55	0,556	0,56
precision	0,93	1	1	0,99	0,75	1	0,788	1	0,906	0,88

Tabel 4.7 Recall and Precision Skenario 5

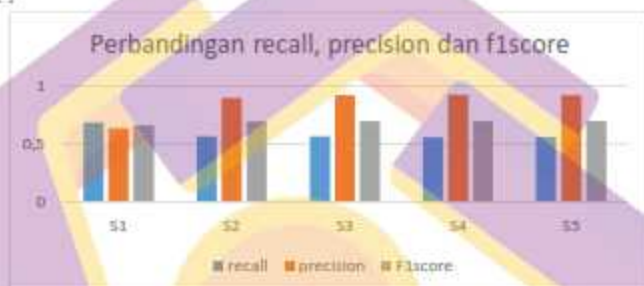
S5	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
recall	0,56	0,55	0,55	0,56	0,56	0,55	0,583	0,56	0,556	0,56
precision	0,88	1	1	0,99	0,77	1	0,788	1	0,906	0,88

Dari perhitungan recall dan precision pada setiap Skenario Model Rekomendasi, diperoleh rata-rata perhitungan dan F1-score seperti pada tabel 4.8 berikut :

Tabel 4.8 Nilai rata-rata recall, precision dan f1-score

	S1	S2	S3	S4	S5
recall	0,683115	0,562778	0,562222	0,5575	0,558056
precision	0,634823	0,897738	0,919398	0,924093	0,920224
f1-score	0,658084	0,691847	0,697758	0,695443	0,694775

Grafik perbandingan nilai rata-rata dapat divisualisasikan pada gambar 4.36 berikut :



Gambar 4.36 Perbandingan Nilai Recall, Precision Dan F1-Score

Dapat dilihat pada tabel 4.8 dan grafik pada gambar 4.36, diperoleh pengukuran bahwa Model Rekomendasi skenario ke-3 memiliki nilai F1-score tertinggi yaitu 0,697758 dibandingkan skenario lainnya. Sebagai tolok ukur, skenario ke-1 mendapatkan nilai pengukuran F1-score 0,658084.

4.6.2. Mean Average Precision

Pada setiap Model Skenario, rekapitulasi hasil rekomendasi dari setiap query memiliki nilai relevansi pada tabel-tabel berikut :

Tabel 4.9 Relevansi Hasil Rekomendasi Model Skenario 1

Rekom ke	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
1	TR	TR	TR	TR	TR	R	R	R	R	R
2	TR	R	TR	TR	TR	R	TR	R	TR	R
3	R	TR	R	R	TR	R	TR	R	TR	R
4	TR	TR	R	R	TR	R	TR	R	TR	R
5	R	TR	R	TR	TR	R	TR	R	TR	TR
6	R	R	TR	TR	R	R	TR	TR	TR	R
7	TR	TR	TR	TR	TR	R	TR	TR	TR	TR
8	R	R	R	TR	TR	TR	R	R	TR	TR
9	TR	TR	R	TR	R	R	TR	R	TR	R
10	TR	TR	TR	TR	TR	R	TR	R	TR	R

Tabel 4.10 Relevansi Hasil Rekomendasi Model Skenario 2

Rekom ke	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
1	R	R	R	TR	R	R	R	R	R	R
2	R	R	R	R	R	R	R	R	R	R
3	R	R	R	TR	TR	R	TR	R	R	R
4	R	R	R	TR	R	R	R	R	R	R
5	R	R	R	TR	R	R	TR	R	R	R
6	R	TR	TR	R	R	R	TR	R	R	R
7	R	R	R	R	R	R	R	R	R	R
8	R	R	R	R	R	R	TR	R	R	R
9	TR	R	R	TR	R	R	TR	R	R	R
10	R	R	R	R	R	R	R	R	TR	R

Tabel 4.11 Relevansi Hasil Rekomendasi Model Skenario 3

Rekom ke	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
1	R	R	R	TR	R	R	R	R	R	R
2	R	R	R	R	R	R	R	R	R	R
3	R	R	R	TR	TR	R	R	R	R	R
4	R	R	R	TR	R	R	R	R	R	R
5	R	R	TR	R	R	R	TR	R	R	R
6	R	R	R	R	R	R	TR	R	R	R
7	R	TR	R	R	R	R	R	R	R	R
8	TR	R	R	TR	R	R	TR	R	R	R
9	R	R	R	R	R	R	TR	R	R	R
10	R	R	R	TR	R	R	TR	R	R	R

Tabel 4.12 Relevansi Hasil Rekomendasi Model Skenario 4

Rekom ke	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
1	R	R	R	R	TR	R	R	R	R	R
2	R	R	R	R	R	R	R	R	R	R
3	R	R	R	R	R	R	R	R	R	TR
4	R	R	R	R	R	R	TR	R	TR	R
5	TR	R	R	R	R	R	TR	R	R	R
6	R	R	R	R	R	R	TR	R	R	R
7	R	R	R	R	R	R	R	R	R	R
8	R	R	R	R	TR	R	TR	R	R	R
9	R	R	R	TR	R	R	R	R	R	R
10	TR	R	R	R	R	R	R	R	R	R

Tabel 4.13 Relevansi Hasil Rekomendasi Model Skenario 5

Rekom ke	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
1	R	R	R	R	R	R	R	R	R	R
2	R	R	R	R	TR	R	R	R	R	R
3	TR	R	R	R	R	R	R	R	R	TR
4	R	R	R	R	R	R	TR	R	TR	R

Tabel 4.13 Relevansi Hasil Rekomendasi Model Skenario 5 (lanjutan)

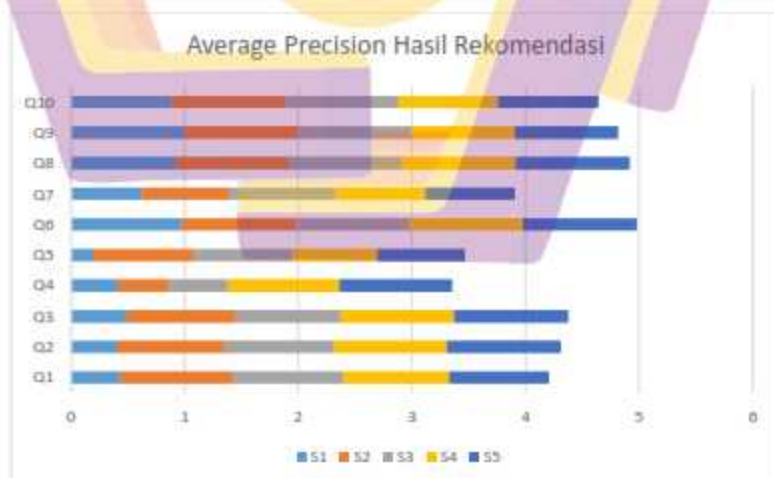
Rekom ke	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
5	R	R	R	R	TR	R	TR	R	R	R
6	R	R	R	R	R	R	TR	R	R	R
7	R	R	R	R	R	R	R	R	R	R
8	R	R	R	R	R	R	TR	R	R	R
9	R	R	R	TR	R	R	R	R	R	R
10	TR	R	R	R	R	R	R	TR	R	R

Dari tabel relevansi diatas, rekomendasi yang relevan dengan query diberi nilai R dan rekomendasi yang tidak relevan terhadap query diberi nilai TR. Setelah dilakukan perhitungan pada sepuluh dokumen rekomendasi dari setiap query, rata-rata nilai *Average Precision* dari hasil rekomendasi setiap query pada tabel 4.14 berikut :

Tabel 4.14 Hasil Pengujian AP

	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
S1	0,433	0,403	0,498	0,417	0,194	0,977	0,625	0,916	1	0,886
S2	0,989	0,947	0,947	0,452	0,878	1	0,764	1	1	1
S3	0,977	0,963	0,928	0,505	0,878	1	0,943	1	1	1
S4	0,932	1	1	0,989	0,748	1	0,788	1	0,906	0,878
S5	0,876	1	1	0,989	0,766	1	0,788	1	0,906	0,878

Dari tabel 4.13 dapat di visualisasikan kedalam grafik seperti pada gambar 4.37 berikut :

Gambar 4.37 Grafik *Average Precision* Hasil Rekomendasi

Nilai *Average Precision* terbanyak diperoleh pada pengujian query ke-6 yaitu “Nila Pedas Manis” dikarenakan memiliki komposisi bahan makanan yang banyak ditemukan pada data koleksi sedangkan nilai terendah diperoleh pada pengujian ke-4 yaitu “Perkedel Tempe” karena memiliki komposisi yang tidak banyak ditemukan pada dokumen koleksi.

Tabel 4.15 *Mean Average Precision* Semua Model Rekomendasi

	MAP
S1	0,635
S2	0,898
S3	0,919
S4	0,924
S5	0,920

Seluruh hasil pengujian *Average Precision* dilakukan perhitungan rata-rata yang ditabulasikan pada tabel 4.15. Diperoleh nilai MAP yang divisualisasikan seperti pada gambar 4.38 berikut :



Gambar 4.38 Grafik Pengujian *Mean Average Precision*

Pada pengujian yang telah dilakukan, MAP terendah diperoleh pada Model Rekomendasi Skenario 1 dengan nilai MAP 0,635. Skenario 1 adalah model standar metode *Content Based Filtering* dengan menggunakan algoritma TF-IDF dan *Cosine Similarity*. Pengukuran nilai MAP terbaik dengan nilai 0,924 diperoleh pada Model Rekomendasi Skenario 4. Model Rekomendasi Skenario 4 yaitu model penerapan algoritma *Boyer-Moore* dalam *Content Based Filtering* dengan TF-IDF *bigram*. Pada skenario 4 pengukuran bobot dan kesamaan menggunakan data hasil

ekstraksi bahan makanan pada query dan data koleksi menggunakan *Boyer-Moore*. Dari pengukuran ini diperoleh kesimpulan bahwa Model Rekomendasi skenario 4 dapat meningkatkan nilai MAP pendekatan CBF sebesar 45,51%.

4.6.3. Average Weight Ingredient Contained

Pada tahap selanjutnya yaitu perhitungan rata-rata bobot bahan makanan yang terkandung didalam hasil rekomendasi atau *Average Weight Ingredient Contained*. Perhitungan yang dilakukan berdasarkan jumlah bahan makanan yang sama dengan query dibagi dengan total bahan makanan pada setiap hasil rekomendasi. *Average Weight Ingredient Contained* menghitung total kandungan bobot yang dihasilkan dari seluruh query pada setiap skenario pengujian, sehingga didapatkan rata-rata bobot kandungan bahan makanan pada setiap model. Dari hasil pengujian diperoleh tabel perhitungan persentase bobot dengan pembulatan 2 angka dibelakang koma pada tabel 4.16 berikut :

Tabel 4.16 Bobot Kandungan Bahan Makanan Hasil Rekomendasi

	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
S1	41,21	43,85	48,01	38,68	39,05	61,21	41,97	54,3	42,93	50,6
S2	57,39	54,92	60,01	47,51	58,2	66,6	51,43	75,93	59,73	64,22
S3	56,95	54,47	60,01	47,63	63,2	66,6	52,24	80,37	61,81	63,72
S4	63,69	60,75	68,36	55,42	66,58	73,35	53,66	78,75	61,34	65,08
S5	63,69	60,06	68,36	55,26	66,25	73,35	54,28	75,66	61,34	64,75
TOTAL	282,9	274,0	304,8	244,5	293,3	341,1	253,6	365,0	287,1	308,4

Grafik hasil pembobotan terbaik dari skenario penelitian dapat dilihat pada gambar 4.39 berikut:



Gambar 4.39 Grafik Bobot Kandungan Makanan Hasil Rekomendasi

Bobot kandungan bahan makanan terbesar yang dihasilkan dari proses rekomendasi diperoleh pada pengujian query ke-8 yaitu “Tongseng Kambing” dengan nilai 365,0. Sedangkan kandungan bahan makanan hasil rekomendasi terkecil diperoleh pada pengujian dengan query ke-4 yaitu “Perkedel Tempe” yaitu 244,5. Bobot rata-rata dari setiap skenario pengujian dapat dilihat pada tabel 4.17 berikut :

Tabel 4.17 Bobot Rata-rata Bahan Makanan Terkandung

	Bobot Rata-rata Bahan Makanan Terkandung
Skenario 1	46,18103
Skenario 2	59,59564
Skenario 3	60,70134
Skenario 4	64,69783
Skenario 5	64,29947

Grafik bobot rata-rata kandungan bahan makanan dari keseluruhan skenario penelitian dapat dilihat pada gambar 4.40 berikut



Gambar 4.40 Grafik *Average Weight Ingredient Contained*

Dari tabel 4.17 dan Gambar 4.40 dapat dilihat bahwa bobot rata-rata kandungan bahan makanan terbaik yang dihasilkan oleh penelitian ini terdapat pada skenario ke 4 dengan nilai *Average Weight Ingredient Contained* sebesar 64, 69783. Pada skenario 1 yang merupakan tolok ukur penelitian menunjukkan nilai *Average Weight Ingredient Contained* sebesar 46,18103. Sehingga dapat ditarik kesimpulan bahwa Model Rekomendasi skenario ke-4 dapat meningkatkan nilai kandungan bahan makanan sebesar 40,1 %.

Dari keseluruhan pengujian didapatkan nilai evaluasi yang ditabulasikan pada tabel 4.18 berikut :

Tabel 4.18 Evaluasi Skenario Model Rekomendasi

Skenario	Recall	Precision	F1-Score	MAP	AWIC
1	0,683	0,634	0,658	0,635	46,18103
2	0,562	0,897	0,691	0,898	59,59564
3	0,562	0,919	0,697	0,919	60,70134
4	0,557	0,924	0,695	0,924	64,69783
5	0,558	0,920	0,694	0,920	64,29947

Keluaran dari skenario model rekomendasi memiliki perbedaan yang sangat signifikan, hasil rekomendasi CBF dengan menggunakan TF-IDF dan *Cosine Similarity* masih memiliki nilai MAP dan AWIC rendah. Dengan penambahan algoritma *Boyer-Moore* nilai MAP dan AWIC meningkat. Nilai *ngram_range* sangat mempengaruhi hasil rekomendasi dapat dilihat pada perbedaan hasil rekomendasi skenario ke 2,3,4 dan 5. Hal ini dapat terjadi karena *ngram_range* membatasi panjang urutan kata dalam *text* yang diolah. *Bigram* dengan rentang *ngram* 1,2 membatasi pengolahan panjang urutan kata dari 1 kata hingga 2 kata sedangkan *trigram* memiliki rentang 1,3 sehingga mengolah urutan kata hingga 3 kata.

Pada penelitian R. Janani dan S. Vijarani, dengan inputan file .txt, *Boyer-Moore* memiliki nilai relevansi 92% sedangkan dengan inputan file .docx *Boyer-Moore* memiliki nilai relevansi 90%. Pada penelitian yang sama, R. Janani dan S. Vijarani, melakukan evaluasi terhadap Algoritma *Enhanced Boyer-Moore* menghasilkan nilai relevansi 95% pada inputan .txt file, sedangkan inputan .docx menghasilkan nilai relevansi 99%. Dibandingkan dengan hasil penelitian ini, nilai relevansi (MAP) yang dihasilkan 0,924 atau 92,4% sehingga dapat dikatakan sama dengan *Boyer-Moore* pada penelitian R. Janani dan S. Vijarani pada inputan file .txt, namun lebih baik dibandingkan inputan file .docx. Namun, penelitian ini tidak lebih baik jika dibandingkan dengan hasil evaluasi *Enhanced Boyer-Moore* pada penelitian R. Janani dan S. Vijarani.

BAB V PENUTUP

5.1. Kesimpulan

Penelitian ini menggunakan pendekatan *Content Based Filtering* (CBF) dengan penambahan algoritma pencarian string *Boyer-Moore* (BM) dalam mempertimbangkan bahan makanan sebagai obyek yang terkandung didalam hasil rekomendasi resep masakan. Dari hasil penelitian dapat ditarik kesimpulan bahwa:

1. Penambahan algoritma BM pada pendekatan CBF dapat diterapkan sebagai model rekomendasi resep masakan yang menggabungkan kedekatan jarak vektor dengan bobot bahan makanan sebagai obyek, sehingga direpresentasikan bahwa resep masakan yang direkomendasikan memiliki kedekatan dan memiliki komposisi bahan makanan yang sesuai.
2. Penerapan algoritma pencarian string BM kedalam pendekatan CBF mampu menaikkan nilai relevansi sebesar 45,51% dan nilai bobot rata-rata kandungan bahan makanan sebesar 40,1% pada sistem rekomendasi resep masakan dengan Model Rekomendasi Skenario ke-4. Model Rekomendasi Skenario ke-4 menggunakan TF-IDF *bigram* dan ekstraksi pada query dan data koleksi mampu menghasilkan nilai MAP sebesar 0,9240 dan AWIC sebesar 64,69783 dengan nilai F1-score 0,695443. Sedangkan pada model CBF standar hanya mampu menghasilkan nilai MAP sebesar 0,6350 dan nilai AWIC sebesar 46,18103 dengan nilai F1-score 0,658084. Dalam pengukuran evaluasi diperoleh fakta bahwa pada penelitian berbasis urutan kata atau obyek, *trigram* pada TF-IDF tidak selalu memberikan hasil rekomendasi terbaik dibanding *bigram*. Dibuktikan dengan nilai evaluasi *bigram* yang lebih besar dibandingkan *trigram*. Hal ini dipengaruhi oleh mayoritas obyek bahan makanan cenderung memiliki gabungan sebanyak 2 kata.

5.2. Saran

Penelitian ini dapat dikembangkan untuk rekomendasi masakan yang mempertimbangkan kebutuhan khusus pengguna, bahan makanan yang tidak dianjurkan untuk dikonsumsi bagi penderita penyakit tertentu dan kebiasaan pengguna seperti vegan dan vegetarian. Pengembangan lain dari penelitian ini dapat diimplementasikan pada sistem rekomendasi di bidang kesehatan, wisata, peternakan, pertanian dan bidang lain yang menitikberatkan kandungan obyek dan kemiripan informasi. Penelitian selanjutnya diharapkan dapat menggali fitur lain seperti gambar, kuantitas dan ukuran satuan bahan makanan, sehingga diperoleh saran resep yang lebih sesuai dengan tampilan dan dengan ukuran penyajian yang sama.

Optimalisasi menggunakan varian lain dari Algoritma *Boyer-Moore* seperti *Turbo Boyer-Moore*, *Boyer-Moore-Horspool* dan *Enhanced Boyer-Moore*, diharapkan dapat lebih meningkatkan relevansi rekomendasi. Penerapan kesamaan fonetik seperti "cabe = cabai" atau penerapan sinonim seperti "Lengkuas, Laos, Gadamala, Laja" dapat digunakan untuk menjembatani perbedaan penamaan yang merujuk pada obyek yang sama. Penerapan metode fonetik atau sinonim tersebut diharapkan dapat membantu model mengurangi bias dalam melakukan ekstraksi sehingga diperoleh bobot kandungan obyek dan nilai relevansi yang lebih baik.

DAFTAR PUSTAKA

PUSTAKA BUKU

- Karina, M.,S and Endang T.,A. 2017. "Pengembangan Kuliner". Pusat Pendidikan Sumber Daya Manusia Kesehatan. Kementrian Kesehatan Republik Indonesia.
- Kusrini., E T Luthfi. 2009. "Algoritma Data Mining". Yogyakarta: Penerbit Andi.
- Lancaster, F. W. (1979). "Information Retrieval System: Characteristics, Testing, and Evaluation (2 nd Editi)". New York: John Wiley.
- Pendit, P. L. (2008). *Perpustakaan Digital dari A sampai Z*. Jakarta: Citra Karya Karsa Mandiri.
- Ricii, F. et al., 2011. *Recommender System Handbook*, New York:Springer.
- Sutarman. 2012. "Pengantar Teknologi Informasi". Jakarta: Bumi Aksara.

PUSTAKA JURNAL

- Achakulvisut, T., D. E. Acuna, T. Ruangrong, and K. Kording. 2016. "Science Concierge: A fast contentbased recommendation system for scientific," *PLoS One*. doi: doi.org/10.1371/ journal.pone.0158423.
- Agung, Halim dan Yogyakarta. 2016. "Implementasi Boyer-Moore Pada Aplikasi Pencarian Rumus Matematika Dan Fisika". *Jurnal Ilmiah Teknologi Informasi Terapan Volume III, No I*. ISSN : 2407 - 3911.
- Bobadilla, J., F. Ortega, A. Hernando, and A. Gutiérrez. 2013. "Recommender Systems Survey. *KnowledgeBased Systems*," vol. 46, pp. 109–132.
- Deshpande, M., A. V. Muddebihalkar, A. B. Jadhav, and S. Kokate. 2016. "Hybrid Content-Based Filtering Recommendation Algorithm on Hadoop," *Int. J. Adv. Res. Comput. Eng. Technol.*, vol. 5.
- Hartatik. Rosyid. 2020. "Pengaruh User Profiling Pada Rekomendasi Sistem Menggunakan K-Means dan KNN. *Jurnal Of Information System Management*. Vol.2, No.1.
- Janani, R., and S. Vijayarani. 2016. "An Efficient Text Pattern Matching Algorithm for Retrieving Information from Desktop". *Indian Journal of Science and Technology*, Vol 9(43), DOI: 10.17485/ijst/2016/v9i43/95454
- Kalita, J., V. E. Balas, S. Borah, and R. Pradhan. 2019. "Recent Developments in Machine Learning and Data Analytics," Springer Singapore, vol. 740, 2019, doi: 10.1007/978- 981-13-1280-9.
- Melita, Ria dkk. 2018. "Penerapan Metode Term Frequency Inverse Document Frequency (TF-IDF) Dan Cosine Similarity Pada Sistem Temu Kembali Informasi Untuk Mengetahui Syarah Hadits Berbasis Web (Studi Kasus:

- Syarah Umdatil Ahkam)". *Jurnal Teknik Informatika* Vol 11 No. 2. Universitas Islam Negeri Syarif Hidayatullah Jakarta.
- Nastiti, P. 2019. "Penerapan Metode Content Based Filtering Dalam Implementasi Sistem Rekomendasi Tanaman Pangan," *Teknika*, vol. 8, no. 1, pp. 1–10, 2019, doi: 10.34148/teknika.v8i1.139.
- Parwita, WGS. 2019. "Penguujian Akurasi Sistem Rekomendasi Berbasis Content-Based Filtering". *Jurnal Ilmiah Ilmu Komputer*, vol. 14, pp. 1, doi : 10.30872/jim.v14i1.1272
- Prasetya, Candra S.D. 2017. "Sistem Rekomendasi Pada E-Commerce Menggunakan K-Nearest Neighbor". *Jurnal Teknologi Informasi dan Ilmu Komputer* 4(3):194. DOI:10.25126/jtiik.201743392
- Rahmanita, E. 2014. "Pencarian String Menggunakan Algoritma Boyer Moore Pada Dokumen". *Jurnal Ilmiah NERO* Vol.1. No.1. hal 15-26.
- Robertson, Stephen. 2005. "Understanding Inverse Document Frequency: On theoretical arguments for IDF". England : *Journal of Documentation*, Vol. 60, pp. 502–520
- Saadah, Munjiah., dkk. 2013. "Sistem Temu Kembali Dokumen Teks Dengan Pembobotan Tf-Idf Dan LCS". *JUTI*. Volume 11, Nomor 1, Januari 2013 : 17 – 20
- Sagita, V., & Prasetyowati, M. (2013). Studi Perbandingan Implementasi Algoritma Boyer-Moore, Turbo Boyer-Moore, dan Tuned Boyer-Moore dalam Pencarian String. *Ultimatics : Jurnal Teknik Informatika*, 5(1), 31-37. <https://doi.org/https://doi.org/10.31937/ti.v5i1.311>
- Setiawan, E. Soegiarto. 2017. "Aplikasi Rekomendasi Pemilihan Menu Makanan Sehat Untuk Anak Penderita Obesitas Menggunakan Metode Fuzzy Tahani". *Progresif: Jurnal Ilmiah Komputer*, Vol. 13, Hal.1.

PUSTAKA LAPORAN PENELITIAN

- Ma, Ke. 2016. "Content-based Recommender System for Movie Website". KTH Royal Institute of Technology. Swedia.

PUSTAKA ELEKTRONIK

- Mihalcea, Rada. "Information Retrieval and Web Search, IR Evaluation and IR Standard Text Collections" diakses pada 15 desember 2021 dari <http://web.eecs.umich.edu/~mihalcea/498IR/Lectures/IREvaluation.pdf>

LAMPIRAN

Hasil Rekomendasi Q1

Di [10]: `rekomendasi_10["Ayam goreng (saling layak)"]`

```
out[10]: ['Lenggang kambing rambutan 10',
         'Kemping Telangan Kuning',
         'Gulai Ikan Gurami',
         'Rasa Petak Gurami',
         'Kopi Ayam',
         'Ayam Ungkep',
         'Mangkok Cendol',
         'Ayam goreng serunding lenggang',
         'Ayam Goreng',
         'Lenggang kambing jagga (statis varian mertak)']
```

Di [11]: `rekomendasi_11["Ayam goreng (saling layak)"]`

```
out[11]: ['Lada goreng putih',
         'Ayam goreng lenggang ala Thai',
         'Ayam Ungkep',
         'Kopi Ayam',
         'Lada goreng',
         'Ayam bumbu kuning',
         'Gulai Ikan Gurami',
         'Ayam goreng Lenggang',
         'Terdid daging kambing klasik',
         'Ayam goreng lenggang klasik']
```

Di [12]: `rekomendasi_12["Ayam goreng (saling layak)"]`

```
out[12]: ['Lada goreng putih',
         'Ayam Ungkep',
         'Ayam goreng lenggang ala Thai',
         'Kopi Ayam',
         'Ayam bumbu kuning',
         'Kopi Ayam',
         'Ayam goreng putih',
         'Terdid daging kambing klasik',
         'Rice-rice ayam kampung',
         'Gulai Ikan Gurami']
```

Di [13]: `rekomendasi_13["Ayam goreng (saling layak)"]`

```
out[13]: ['Lada goreng putih',
         'Ayam goreng putih',
         'Ayam goreng lenggang ala Thai',
         'Ayam Goreng',
         'Ayam goreng',
         'Rice-rice ayam kampung',
         'Lada bumbu kuning',
         'Lada Goreng Rayak',
         'Kopi Ayam']
```

Di [14]: `rekomendasi_14["Ayam goreng (saling layak)"]`

```
out[14]: ['Ayam goreng putih',
         'Lada Goreng Putih',
         'Ayam Goreng',
         'Ayam Goreng',
         'Ayam Goreng',
         'Ayam goreng lenggang ala Thai',
         'Kopi Ayam',
         'Lada Goreng Putih',
         'Lada bumbu kuning',
         'Rice-rice ayam kampung',
         'Kopi Ayam']
```


Hasil rekomendasi Q2

```
In [10]: recom_Q2_skenario_1('Samar Tengah Laut')
```

```
Out[10]: ['Kap talang lada hitam',
          'Ayam cincalol',
          'Sate lambang',
          'Surasa naco',
          'Tongkol sapi, tongkol kambing hitam',
          'Ayam masak putih lada hitam',
          'Bakul ikan',
          'Sate goreng kambing 50%',
          'Usang tongkol pedas',
          'Ayam goreng setengah']
```

```
In [11]: recom_Q2_skenario_2('Samar Tengah Laut')
```

```
Out[11]: ['Ayam goreng setengah simple',
          'Sate goreng kambing 50%',
          'Lambel usang goreng',
          'Usang kambing kacang asam manis simple',
          'Ayam cincalol',
          'Kap talang kambing',
          'Sate kambing bumbu',
          'Ayam caur lada hitam',
          'Lambel telur ayam',
          'Tongkol caur (campur) asam manis']
```

```
In [12]: recom_Q2_skenario_3('Samar Tengah Laut')
```

```
Out[12]: ['Ayam goreng setengah simple',
          'Ayam cincalol',
          'Lambel usang goreng',
          'Sate goreng kambing 50%',
          'Usang kambing kacang asam manis simple',
          'Sate kambing bumbu',
          'Kap talang kambing',
          'Lambel telur ayam',
          'Ayam caur lada hitam',
          'Telur lada hitam panggang']
```

```
In [13]: recom_Q2_skenario_4('Samar Tengah Laut')
```

```
Out[13]: ['Tongkol kambing lada hitam',
          'Sate goreng kambing 50%',
          'Bakul ikan gigit panggang',
          'Ayam goreng setengah simple',
          'Tongkol Lada',
          'Ayam masak putih lada hitam',
          'Garudan = sosis',
          'Ikan bakar campur telur ala mas house',
          'Ayam cincalol',
          'Sate kambing bakar']
```

```
In [14]: recom_Q2_skenario_5('Samar Tengah Laut')
```

```
Out[14]: ['Tongkol kambing lada hitam',
          'Sate goreng kambing 50%',
          'Bakul ikan gigit panggang',
          'Ayam goreng setengah simple',
          'Ikan bakar campur telur ala mas house',
          'Garudan = sosis',
          'Ayam masak putih lada hitam',
          'Ayam cincalol',
          'Tongkol kambing',
          'Ivonnee goreng daging']
```

Hasil Rekomendasi Q3

Di [14]: `recom_14_scenario_1('lantai telur ayam')`

```
out[14]: ['lantai bertelur dan ayam kelor',
         'lantai-lantai ayam',
         'lantai bambu lada ayam manis simple',
         'lantai terong-jarden',
         'lantai jarden',
         'lantai ulang gunung',
         'ayam masak jarden',
         'lantai telur pedas mentah',
         'lantai terong-jarden',
         'lantai lada ayam']
```

Di [15]: `recom_15_scenario_2('lantai telur ayam')`

```
out[15]: ['lantai telur telur jarden',
         'lantai all simple ayam',
         'ayam masak mentah-lada',
         'lantai terong-jarden',
         'lantai telur pedas mentah',
         'ayam lada ayam jarden',
         'lantai bambu lada ayam manis simple',
         'lantai all terong-jarden lada ayam',
         'ayam lada ayam',
         'lantai telur-jarden']
```

Di [16]: `recom_16_scenario_3('lantai telur ayam')`

```
out[16]: ['lantai telur telur jarden',
         'lantai all simple ayam',
         'ayam masak mentah-lada',
         'lantai terong-jarden',
         'ayam lada ayam jarden',
         'lantai bambu lada ayam manis simple',
         'lantai telur pedas mentah',
         'lantai all terong-jarden lada ayam',
         'lantai telur-jarden',
         'lantai lada ayam']
```

Di [17]: `recom_17_scenario_4('lantai telur ayam')`

```
out[17]: ['lantai telur telur jarden',
         'lantai telur telur (jard)',
         'ayam lada ayam',
         'lantai terong-jarden',
         'lantai bambu lada ayam',
         'lantai bambu lada ayam manis simple',
         'lantai telur pedas mentah',
         'lantai bambu lada ayam',
         'lantai telur telur lada ayam manis',
         'lantai telur telur lada ayam manis']
```

Di [18]: `recom_18_scenario_4('lantai telur ayam')`

```
out[18]: ['lantai telur telur jarden',
         'lantai telur telur (jard)',
         'lantai bambu lada ayam',
         'ayam lada ayam',
         'lantai bambu lada ayam',
         'lantai terong-jarden',
         'lantai bambu lada ayam manis simple',
         'lantai telur pedas mentah',
         'lantai bambu lada ayam',
         'lantai telur telur lada ayam manis']
```

Hasil Rekomendasi Q4

In (43): `recom_cas_skenario_1('skenario: teapa')`

```
out(43): ['Gabus asam sambal teapa',
         'Mact goreng kari kembang',
         'Nyax teapa sarat nambu aiek (recom)',
         'Klotok cabe ijo',
         'Maggat reka tigoja',
         'Meris asam teapa teapa',
         'Serdon buah asam',
         'Mujon asam pedas meris',
         'Mact rebut kembang (sax_sitarwood)',
         'Seta goreng kembang 01***']
```

In (44): `recom_CSP_BI_skenario_2('skenario: teapa')`

```
out(44): ['Seta kembang goreng',
         'Lada Garing Rongga',
         'Kembang tangkil',
         'Kembang oven',
         'Asam kacang ulolasa',
         'Nyax goreng ungka prattis',
         'Asam ungka',
         'Majak Tongkol',
         'Maggat Teapa Simala',
         'Lada tangkil kembang sager']
```

In (45): `recom_CSP_BI_skenario_3('skenario: teapa')`

```
out(45): ['Seta kembang goreng',
         'Lada kembang sapan',
         'Kembang tangkil',
         'Kembang tangkil',
         'Nyax goreng ungka prattis',
         'Kembang goreng teapa',
         'Asam ungka',
         'Kembang oven',
         'Majak tangkil',
         'Mact asam kembang teapa']
```

In (46): `recom_CSP_BI_skenario_4('skenario: teapa')`

```
out(46): ['Lada tangkil kembang sager',
         'Tangkil sapan',
         'Klotok cabe ijo',
         'Majak Tongkol',
         'Lada kembang sapan',
         'Lada kembang sapan',
         'Dua asam sarat sapan',
         'Nyax goreng ungka prattis',
         'Lada kembang goreng',
         'Nyax teapa asam kembang sager (recom)']
```

In (47): `recom_CSP_BI_skenario_5('skenario: teapa')`

```
out(47): ['Lada tangkil kembang sager',
         'Tangkil sapan',
         'Klotok cabe ijo',
         'Lada Garing Rongga',
         'Majak Tongkol',
         'Lada kembang sapan',
         'Lada kembang sapan',
         'Nyax goreng ungka prattis',
         'Lada kembang goreng',
         'Nyax kembang sapan']
```


Hasil Rekomendasi Q6

In [42]: `recom_04_01_recoms_01('Hila Haba Nasta')`

```
Out[42]: ['Ayan Samud',
          'Ayan Samud 001',
          'Jean Hajar: Samud 001',
          'Jean Hening 0000',
          'Sali Jean Hening',
          'Kalado: Jean pari asap',
          'Mali Jean Hajar 00000',
          'Mali Hajar 00000',
          'Ayan Asit: Samud Nasta',
          'Ayan Asit 0000']
```

In [43]: `recom_04_01_recoms_01('Hila Haba Nasta')`

```
Out[43]: ['Tinggi Naur (Samud) Ayan Nasta',
          'Layang Kembang Catur Ayan Nasta Samud',
          'Ayan Samud 001',
          'Jean Hening 0000',
          'Sali Jean Hajar',
          'Jean Tinggi Kalado',
          'Tinggi Naur: Jean Hening',
          'Layang Sali Hening Ayan',
          'Ayan Samud',
          'Kalado: Jean pari asap']
```

In [44]: `recom_04_01_recoms_01('Hila Haba Nasta')`

```
Out[44]: ['Tinggi Naur (Samud) Ayan Nasta',
          'Layang Kembang Catur Ayan Nasta Samud',
          'Ayan Samud 001',
          'Layang Sali Hening Ayan',
          'Jean Tinggi Kalado',
          'Jean Hening 0000',
          'Sali Jean Hajar',
          'Kalado: Jean pari asap',
          'Tinggi Naur: Jean Hening',
          'Ayan Samud']
```

In [45]: `recom_04_01_recoms_01('Hila Haba Nasta')`

```
Out[45]: ['Jean Hajar: Samud 001',
          'Jean Tinggi: Jean Haba 001',
          'Jean Hening 0000',
          'Mali Jean Hajar 0000',
          'Tinggi Naur (Samud) Ayan Nasta',
          'Sali Jean Hajar',
          'Mali Jean Hajar 0000',
          'Jean Hajar: Samud 001',
          'Jean Tinggi: Jean Haba',
          'Mali Jean Hajar: Hening Nasta']
```

In [46]: `recom_04_01_recoms_01('Hila Haba Nasta')`

```
Out[46]: ['Jean Hajar: Samud 001',
          'Jean Tinggi: Jean Haba 001',
          'Jean Hening 0000',
          'Mali Jean Hajar 0000',
          'Tinggi Naur (Samud) Ayan Nasta',
          'Sali Jean Hajar',
          'Jean Hajar: Samud 001',
          'Jean Tinggi: Jean Haba',
          'Mali Jean Hajar: Hening Nasta']
```

Hasil Rekomendasi Q7

70 [10]	recom_C07_01_konvensi_11('hari Gulet Garwa')
0A130	['Dian tengali lauir pado', 'Malar' dan pado malar', 'Bukane peggang tepian buko padang', 'Mangut dan Gebu', 'Amur ayan kawang simple', 'Dian karama ayan mangal', 'Bali dan kawang', 'Bulu dan pado peggang', 'Petr' haka kuring kawang malar', 'Mara' (top kawang dan' (top' (top')
70 [17]	recom_C07_01_konvensi_0('hari Gulet Garwa')
0A137	['Dian' dia gung' anektis', 'Dian' tengali lauir pado', 'Mangut dan malar', 'Ayan' eldir' amal' malar', 'Pala' haka kuring kawang malar', 'Mangut' haka pado malar', 'Lala' dan pado', 'Amur' ayan kawang simple', 'Mali' dan kawang', 'Dian' haka kawang malar' haka' dia' malar' kawang']
70 [18]	recom_C07_01_konvensi_0('hari Gulet Garwa')
0A138	['Dian' dia gung' anektis', 'Dian' tengali lauir pado', 'Lala' dan pado', 'Ayan' eldir' amal' malar', 'Mangut' dan malar', 'Malar' dan pado malar', 'Dian' haka kawang malar' haka' dia' malar' kawang', 'Pala' haka kuring kawang malar', 'Mali' dan kawang', 'Mali' dan kawang']
71 [40]	recom_C07_01_konvensi_0('hari Gulet Garwa')
0A139	['Tengali' dia' dia', 'Karam' kawang simple', 'Ayan' pado' (top' haka', 'Mali' kawang kawang', 'Kari' kawang' dia' haka', 'Kari' kawang' dia' haka', 'Kari' kawang' dia' haka', 'Tengali' kawang', 'Lala' dan pado', 'Kari' tengali lauir pado']
71 [40]	recom_C07_01_konvensi_0('hari Gulet Garwa')
0A140	['Tengali' dia' dia', 'Karam' kawang simple', 'Ayan' pado' (top' haka', 'Mali' kawang kawang', 'Kari' kawang' dia' haka', 'Kari' kawang' dia' haka', 'Kari' kawang' dia' haka', 'Tengali' kawang', 'Lala' dan pado', 'Kari' tengali lauir pado']

Hasil Rekomendasi Q8

in (41) recom_sas_skenario_1('longeng kaming')

```
out(41) ['Ayun girang longeng kaming',
        'lora' yan',
        'longeng ayah',
        'longeng kaming',
        'Ayun kaming balok',
        'Ayun kamin kamin',
        'Ayun kamin kamin',
        'Ayun kamin kamin',
        'Ayun kamin kamin',
        'Ayun kamin kamin']
```

D> (42) recom_cpf_01_skenario_2('longeng kaming')

```
out(42) ['Teris kaming longeng kaming',
        'kaming girang kamin',
        'Ayun kamin kamin',
        'longeng kaming kamin',
        'longeng kaming kamin',
        'kamin kamin kamin kamin kamin kamin',
        'kamin kamin',
        'Ayun kamin kamin',
        'Ayun kamin kamin',
        'longeng kamin',
        'kamin kamin kamin kamin',
        'Teris kamin kamin']
```

in (43) recom_sas_skenario_2('longeng kaming')

```
out(43) ['Teris kaming kamin kamin',
        'kamin girang kamin',
        'Ayun kamin kamin',
        'longeng kamin kamin kamin kamin',
        'longeng kamin kamin',
        'kamin kamin',
        'Ayun kamin kamin kamin',
        'longeng kamin kamin',
        'kamin kamin kamin',
        'Ayun kamin kamin']
```

in (44) recom_id_01_skenario_4('longeng kaming')

```
out(44) ['Ayun kamin kamin',
        'kamin kamin kamin',
        'longeng kamin kamin',
        'longeng kamin kamin kamin',
        'longeng kamin kamin kamin',
        'longeng kamin',
        'longeng kamin kamin',
        'kamin kamin',
        'Ayun kamin kamin kamin']
```

in (45) recom_id_01_skenario_4('longeng kaming')

```
out(45) ['Ayun kamin kamin',
        'longeng kamin kamin',
        'kamin kamin kamin',
        'longeng kamin kamin kamin',
        'longeng kamin kamin kamin',
        'longeng kamin',
        'longeng kamin kamin',
        'Ayun kamin kamin kamin']
```


Hasil Rekomendasi Q10

10 [70] rccm_200_1000010_11 [Tongpang (Matah Sasi Pasa)]

001(70) ['Sakai ikan tongpang',
 'Tongpang saming (tanga kemas)',
 'Ajam saming',
 'Tongpang kumbang (tanga saming)',
 'Ikan katin pedas manis',
 'Ikan katin pedas',
 'Ikan katin pedas (katin).... W. Dapet Langer',
 'Ikan katin (katin) katin kating',
 'Ikan katin (katin)',
 'Ikan katin (katin)',
 'Ikan tongpang (ikan katin)']

10 [70] rccm_200_1000010_12 [Tongpang (Matah Sasi Pasa)]

001(70) ['Ikan tongpang saming',
 'Ajam katin saming',
 'Ikan katin saming dan ati kumbang',
 'Tongpang saming saming',
 'Ikan katin saming saming saming',
 'Ikan katin saming',
 'Tanda kating saming',
 'Ikan katin saming',
 'Ikan katin saming',
 'Tongpang saming (saming) saming saming']

10 [70] rccm_200_1000010_13 [Tongpang (Matah Sasi Pasa)]

001(70) ['Ikan tongpang saming',
 'Ajam katin saming',
 'Tongpang saming saming',
 'Ikan katin saming dan ati kumbang',
 'Ikan katin saming saming saming saming',
 'Tanda kating saming',
 'Ikan katin saming',
 'Ikan katin saming',
 'Ajam saming']

10 [70] rccm_200_1000010_14 [Tongpang (Matah Sasi Pasa)]

001(70) ['Ikan tongpang saming saming saming',
 'Tongpang saming saming',
 'Tongpang saming',
 'Ikan katin saming saming saming',
 'Ikan katin saming dan ati kumbang',
 'Tongpang saming saming saming saming',
 'Ikan katin saming saming saming',
 'Ikan katin saming saming saming',
 'Ikan katin saming saming saming']

10 [70] rccm_200_1000010_15 [Tongpang (Matah Sasi Pasa)]

001(70) ['Tongpang saming saming',
 'Ikan tongpang saming saming saming',
 'Tongpang saming saming',
 'Ikan katin saming saming saming',
 'Ikan katin saming saming saming saming',
 'Ikan katin saming saming saming',
 'Ikan katin saming saming saming',
 'Ikan katin saming saming saming']

Hasil Rekomendasi Pengujian Query 1

Skenario 1					
R	Title	Sama	Tidak Sama	Relevansi	Bobot
1	Tongseng kambing dadakaan :D	7	16	TR	30,43478
2	Krengsengan Tulangan Kambing	6	15	TR	28,57143
3	Gulai Ikan Gurami	6	6	R	50
4	Pais Patin Banjar	3	9	TR	25
5	Soto Ayam	7	7	R	50
6	AYAM ungkep	6	3	R	66,66667
7	Tongseng Kambing	7	11	TR	38,88889
8	Ayam goreng serundeng lengkuas	8	8	R	50
9	Ayam suwir	8	10	TR	44,44444
10	Tongseng Kambing Jogja (otentik warisan mertua)	9	23	TR	28,125

Skenario 2					
R	Title	Sama	Tidak Sama	Relevansi	Bobot
1	Lele Goreng Gurih	5	1	R	83,33333
2	Ayam Goreng Lengkuas ala Those	8	4	R	66,66667
3	AYAM ungkep	6	3	R	66,66667
4	Soto Ayam	7	7	R	50
5	Sayap goreng	7	7	R	50
6	Ayam bumbu kuning	6	5	R	54,54545
7	Gulai Ikan Gurami	6	6	R	50
8	Ayam Goreng Lengkuas	8	8	R	50
9	Terik daging kambing simpel	7	10	TR	41,17647
10	Ayam goreng lengkuas kremes	8	5	R	61,53846

Skenario 3					
R	Title	Sama	Tidak Sama	Relevansi	Bobot
1	Lele Goreng Gurih	5	1	R	83,33333
2	AYAM ungkep	6	3	R	66,66667
3	Ayam Goreng Lengkuas ala Those	8	4	R	66,66667
4	Sayap goreng	7	7	R	50
5	Ayam bumbu kuning	6	5	R	54,54545
6	Soto Ayam	7	7	R	50
7	Nila Goreng Gurih	5	5	R	50
8	Terik daging kambing simpel	7	10	TR	41,17647
9	Rica-rica ayam kampung	8	6	R	57,14286
10	Gulai Ikan Gurami	6	6	R	50

Skenario 4					
R	Title	Sama	Tidak Sama	Relevansi	Bobot
1	Lele Goreng Gurih	5	1	R	83,33333
2	Ayam Goreng Simple	4	0	R	100
3	Ayam Goreng Lengkuas ala Those	8	4	R	66,66667
4	Ayam pop	4	1	R	80
5	Ayam Geprek	2	9	TR	18,18182
6	Minyak Ayam	4	2	R	66,66667
7	Rica-rica ayam kampung	8	6	R	57,14286
8	Lele bumbu kuning	7	4	R	63,63636
9	Lele Goreng Renyah	4	2	R	66,66667
10	Kari Ayam	9	17	TR	34,61538

Skenario 5					
R	Title	Sama	Tidak Sama	Relevansi	Bobot
1	Ayam Goreng Simple	4	0	R	100
2	Lele Goreng Gurih	5	1	R	83,33333
3	Ayam Geprek	2	9	TR	18,18182
4	Ayam pop	4	1	R	80
5	Ayam Goreng Lengkuas ala Those	8	4	R	66,66667
6	Minyak Ayam	4	2	R	66,66667
7	Lele Goreng Renyah	4	2	R	66,66667
8	Lele bumbu kuning	7	4	R	63,63636
9	Rica-rica ayam kampung	8	6	R	57,14286
10	Kari Ayam	9	17	TR	34,61538

Hasil Rekomendasi Pengujian Query 2

Skenario 1					
R	Title	Sama	Tidak Sama	Relevansi	Bobot
1	Sop tulang iga kambing	5	11	TR	31,25
2	Ayam coca cola	7	5	R	58,33333
3	Sate Kambing	9	13	TR	40,90909
4	Gurame Kecap	6	14	TR	30
5	Tongseng sapi, tongseng kambing santan	8	13	TR	38,09524
6	Ayam saos teriyaki Lada Hitam	6	4	R	60
7	Bistik ayam	7	10	TR	41,17647
8	Sate goreng kambing 89"	10	6	R	62,5
9	Oseng tongkol pedas	6	8	TR	42,85714
10	Ayam Goreng Mentega	5	10	TR	33,33333

Skenario 2					
R	Title	Sama	Tidak Sama	Relevansi	Bobot
1	Ayam goreng mentega simple	6	4	R	60
2	Sate goreng kambing 89"	10	6	R	62,5
3	Sambal udang goreng	7	6	R	53,84615
4	Oseng kambing kecap asam manis simple	6	4	R	60
5	Ayam coca cola	7	5	R	58,33333
6	Sop tulang kambing	4	6	TR	40
7	Sate Kambing Rumahan	6	4	R	60
8	Ayam suwir lada hitam	8	8	R	50
9	Sambal telur ayam	5	5	R	50
10	Tongkol suwir (pampis) asem manis	6	5	R	54,54545

Skenario 3					
R	Title	Sama	Tidak Sama	Relevansi	Bobot
1	Ayam goreng mentega simple	6	4	R	60
2	Ayam coca cola	7	5	R	58,33333
3	Sambal udang goreng	7	6	R	53,84615
4	Sate goreng kambing 89"	10	6	R	62,5
5	Oseng kambing kecap asam manis simple	6	4	R	60
6	Sate Kambing Rumahan	6	4	R	60
7	Sop tulang kambing	4	6	TR	40
8	Sambal telur ayam	5	5	R	50

9	Ayam suwir lada hitam	8	8	R	50
10	Gulai ikan gabus panggang	7	7	R	50

Skenario 4					
R	Title	Sama	Tidak Sama	Relevansi	Bobot
1	Daging Kambing Lada Hitam	10	6	R	62,5
2	Sate goreng kambing ØY"	10	6	R	62,5
3	Gulai ikan gabus panggang	7	7	R	50
4	Ayam goreng mentega simple	6	4	R	60
5	Pecak Lele	5	3	R	62,5
6	Ayam saos teriyaki Lada Hitam	6	4	R	60
7	Sarden + sosis	5	3	R	62,5
8	Ikan sarden campur telur ala anak kosan	2	1	R	66,66667
9	Ayam coca cola	7	5	R	58,33333
10	Pete Kambing Kecap	5	3	R	62,5

Skenario 5					
R	Title	Sama	Tidak Sama	Relevansi	Bobot
1	Daging Kambing Lada Hitam	10	6	R	62,5
2	Sate goreng kambing ØY"	10	6	R	62,5
3	Gulai ikan gabus panggang	7	7	R	50
4	Ayam goreng mentega simple	6	4	R	60
5	Ikan sarden campur telur ala anak kosan	2	1	R	66,66667
6	Sarden + sosis	5	3	R	62,5
7	Ayam saos teriyaki Lada Hitam	6	4	R	60
8	Ayam coca cola	7	5	R	58,33333
9	Pete Kambing Kecap	5	3	R	62,5
10	Bandeng Presto Balado	5	4	R	55,55556

Hasil Rekomendasi Pengujian Query 3

Skenario 1					
R	Title	Sama	Tidak Sama	Relevansi	Bobot
1	Nasi brokoli dan ayam kecap	3	9	TR	25
2	Rica-Rica Ayam	7	9	TR	43,75
3	Oseng kambing kecap asam manis simple	6	4	R	60
4	Tumis terong+ sarden	6	3	R	66,66667
5	Sambal Sarden	5	3	R	62,5
6	Sambal udang goreng	6	7	TR	46,15385
7	Ayam Cabe Garam	6	10	TR	37,5
8	Tongkol suwir pedas manisd'Y''	7	6	R	53,84615
9	Tumis Tongkol Petai	5	3	R	62,5
10	Somay ikan tuna	6	21	TR	22,22222

Skenario 2					
R	Title	Sama	Tidak Sama	Relevansi	Bobot
1	Orak arik telur buncis	7	1	R	87,5
2	Oseng ati ampela ayam	6	6	R	50
3	Ayam cabe manis ~Leana	6	5	R	54,54545
4	Tumis terong+ sarden	6	3	R	66,66667
5	Tongkol suwir pedas manisd'Y''	7	6	R	53,84615
6	Ayam kentang bumbu paniki	6	8	TR	42,85714
7	Oseng kambing kecap asam manis simple	6	4	R	60
8	Ikan mas terbang sambal dabu-dabu	5	3	R	62,5
9	Ayam kecap pedas	5	4	R	55,55556
10	Bandeng suwir sambel korek	4	2	R	66,66667

Skenario 3					
R	Title	Sama	Tidak Sama	Relevansi	Bobot
1	Orak arik telur buncis	7	1	R	87,5
2	Oseng ati ampela ayam	6	6	R	50
3	Ayam cabe manis ~Leana	6	5	R	54,54545
4	Tumis terong+ sarden	6	3	R	66,66667
5	Ayam kentang bumbu paniki	6	8	TR	42,85714
6	Oseng kambing kecap asam manis simple	6	4	R	60
7	Tongkol suwir pedas manisd'Y''	7	6	R	53,84615
8	Ikan mas terbang sambal dabu-dabu	5	3	R	62,5

9	Bandeng suwir sambel korek	4	2	R	66,66667
10	Bledek kambing	5	4	R	55,55556

Skenario 4					
R	Title	Sama	Tidak Sama	Relevansi	Bobot
1	Orak arik telur buncis	7	1	R	87,5
2	Mangut Ikan Phe (pari)	6	2	R	75
3	Ayam Sambel	6	4	R	60
4	Tumis terong+ sarden	6	3	R	66,66667
5	Kambing mercon ala ovi	6	3	R	66,66667
6	Oseng-oseng tulang kambing pedas	6	2	R	75
7	Pindang Tongkol Bumbu Tomat	6	3	R	66,66667
8	Tongkol sarden pete uendesss	6	4	R	60
9	TTKK (tahu telur timun tauge kuah kacang)	5	3	R	62,5
10	Gurame Asam Manis Seadanya	7	4	R	63,63636

Skenario 5					
R	Title	Sama	Tidak Sama	Relevansi	Bobot
1	Orak arik telur buncis	7	1	R	87,5
2	Mangut Ikan Phe (pari)	6	2	R	75
3	Kambing mercon ala ovi	6	3	R	66,66667
4	Ayam Sambel	6	4	R	60
5	Tongkol sarden pete uendesss	6	4	R	60
6	Tumis terong+ sarden	6	3	R	66,66667
7	Oseng-oseng tulang kambing pedas	6	2	R	75
8	Gurame Asam Manis Seadanya	7	4	R	63,63636
9	Pindang Tongkol Bumbu Tomat	6	3	R	66,66667
10	TTKK (tahu telur timun tauge kuah kacang)	5	3	R	62,5

Hasil Rekomendasi Pengujian Query 4

Skenario 1					
R	Title	Sama	Tidak Sama	Relevansi	Bobot
1	Gabus asin sambal pete	6	8	TR	42,85714
2	Nasi goreng kari kambing	3	13	TR	18,75
3	Orek tempe basah bumbu ulek (recook)	8	6	R	57,14286
4	Klotok cabe ijo	6	4	R	60
5	Nugget Tempe Simple	6	12	TR	33,33333
6	Terik ayam tempe telur	8	11	TR	42,10526
7	Sarden kuah asam	5	7	TR	41,66667
8	Mujaer asam pedas manis	6	10	TR	37,5
9	Nasi Kebuli Kambing (#PR AsianFood)	6	21	TR	22,22222
10	Sate goreng kambing d'Y"	5	11	TR	31,25

Skenario 2					
R	Title	Sama	Tidak Sama	Relevansi	Bobot
1	Sate kambing goreng	4	10	TR	28,57143
2	Lele Goreng Renyah	4	2	R	66,66667
3	Pempek tengiri	4	6	TR	40
4	Kambing Oven	5	9	TR	35,71429
5	Ayam Kecap ulalaaa	4	5	TR	44,44444
6	Ayam Goreng Ungkep Praktis	5	3	R	62,5
7	AYAM ungkep	5	4	R	55,55556
8	Balado Tongkol	6	6	R	50
9	Nugget Tempe Simple	6	12	TR	33,33333
10	Ikan Tenggiri Pesmol Segar	7	5	R	58,33333

Skenario 3					
R	Title	Sama	Tidak Sama	Relevansi	Bobot
1	Sate kambing goreng	4	10	TR	28,57143
2	Lele Goreng Renyah	4	2	R	66,66667
3	Pempek tengiri	4	6	TR	40
4	Ayam Kecap ulalaaa	4	5	TR	44,44444
5	Ayam Goreng Ungkep Praktis	5	3	R	62,5
6	Kambing goreng kecap	3	3	R	50
7	AYAM ungkep	5	4	R	55,55556
8	Kambing Oven	5	9	TR	35,71429
9	Balado Tongkol	6	6	R	50
10	Gabus asin sambal pete	6	8	TR	42,85714

Skenario 4					
R	Title	Sama	Tidak Sama	Relevansi	Bobot
1	Ikan Tenggiri Pesmol Segar	7	5	R	58,33333
2	Tongkol suwir	4	3	R	57,14286
3	Klotok cabe ijo	6	4	R	60
4	Balado Tongkol	6	6	R	50
5	Lele sambal simpel	6	4	R	60
6	Lele Goreng Renyah	4	2	R	66,66667
7	Ikan kuwe merah merona	7	6	R	53,84615
8	Ayam Goreng Ungkep Praktis	5	3	R	62,5
9	Sate kambing goreng	4	10	TR	28,57143
10	Orek tempe basah bumbu ulek (recook)	8	6	R	57,14286

Skenario 5					
R	Title	Sama	Tidak Sama	Relevansi	Bobot
1	Ikan Tenggiri Pesmol Segar	7	5	R	58,33333
2	Tongkol suwir	4	3	R	57,14286
3	Klotok cabe ijo	6	4	R	60
4	Lele Goreng Renyah	4	2	R	66,66667
5	Balado Tongkol	6	6	R	50
6	Lele sambal simpel	6	4	R	60
7	Ikan kuwe merah merona	7	6	R	53,84615
8	Ayam Goreng Ungkep Praktis	5	3	R	62,5
9	Sate kambing goreng	4	10	TR	28,57143
10	Ayam cabai kawin	5	4	R	55,55556

Hasil Rekomendasi Pengujian Query 5

Skenario 1					
R	Title	Sama	Tidak Sama	Relevansi	Bobot
1	Ikan mas bakar terasi	6	9	TR	40
2	Tongseng Kambing	6	12	TR	33,33333
3	Tongseng Kambing (NO MSG)	6	15	TR	28,57143
4	Omlet ayam sayur	2	13	TR	13,33333
5	Bandeng kecil bumbu acar	7	9	TR	43,75
6	Opor Ayam Jawa	9	6	R	60
7	Mujaer asam pedas manis	5	11	TR	31,25
8	Tongseng kambing dadakaan :D	9	14	TR	39,13043
9	Ayam bumbu kuning	7	4	R	63,63636
10	Bawal bakar pedas	6	10	TR	37,5

Skenario 2					
R	Title	Sama	Tidak Sama	Relevansi	Bobot
1	Oseng ati ampela ayam	8	4	R	66,66667
2	Ayam bumbu kuning	7	4	R	63,63636
3	Ayam Geprek	3	8	TR	27,27273
4	Ayam kecap pedas	6	3	R	66,66667
5	Sate kambing goreng	7	7	R	50
6	Ayam Sambel	7	3	R	70
7	Ayam Goreng Lengkuas	8	8	R	50
8	Oseng kambing kecap asam manis simple	6	4	R	60
9	Bandeng Presto Balado	7	2	R	77,77778
10	Udang saus asam manis	8	8	R	50

Skenario 3					
R	Title	Sama	Tidak Sama	Relevansi	Bobot
1	Oseng ati ampela ayam	8	4	R	66,66667
2	Ayam bumbu kuning	7	4	R	63,63636
3	Ayam Geprek	3	8	TR	27,27273
4	Ayam kecap pedas	6	3	R	66,66667
5	Sate kambing goreng	7	7	R	50
6	Bandeng Presto Balado	7	2	R	77,77778
7	Oseng kambing kecap asam manis simple	6	4	R	60
8	Ayam Sambel	7	3	R	70
9	Ayam Goreng Lengkuas	8	8	R	50
10	Ayam pop	5	0	R	100

Skenario 4					
R	Title	Sama	Tidak Sama	Relevansi	Bobot
1	Ayam Geprek	3	8	TR	27,27273
2	Ayam pop	5	0	R	100
3	Bandeng Presto Balado	7	2	R	77,77778
4	Ayam kecap pedas	6	3	R	66,66667
5	Krecek Kambing	8	6	R	57,14286
6	Ayam Bacem Pedas ala DeeðŸŒ,	10	6	R	62,5
7	Oseng kulit ayam (debm)	3	0	R	100
8	Ayam suwir	8	10	TR	44,44444
9	Ayam Sambel	7	3	R	70
10	Opor Ayam Jawa	9	6	R	60

Skenario 5					
R	Title	Sama	Tidak Sama	Relevansi	Bobot
1	Ayam pop	5	0	R	100
2	Ayam Geprek	3	8	TR	27,27273
3	Bandeng Presto Balado	7	2	R	77,77778
4	Oseng kulit ayam (debm)	3	0	R	100
5	Ayam suwir	8	10	TR	44,44444
6	Krecek Kambing	8	6	R	57,14286
7	Ayam kecap pedas	6	3	R	66,66667
8	Opor Ayam Jawa	9	6	R	60
9	Ayam Bacem Pedas ala DeeðŸŒ,	10	6	R	62,5
10	Oseng ati ampela ayam	8	4	R	66,66667

Hasil Rekomendasi Pengujian Query 6

Skenario 1					
R	Title	Sama	Tidak Sama	Relevansi	Bobot
1	Ayam Sambel	7	3	R	70
2	Ayam Bumbu Bali	8	6	R	57,14286
3	Ikan Mujair Sambal Pete	11	6	R	64,70588
4	Ikan kembung balado	11	5	R	68,75
5	Bali ikan Bandeng	7	6	R	53,84615
6	Balado Ikan pari asap	9	3	R	75
7	Gulai ikan gabus panggang	9	5	R	64,28571
8	Pais Patin Banjar	5	7	TR	41,66667
9	Ayam sisit sambal matah	4	4	R	50
10	Ayam cabai kawin	6	3	R	66,66667

Skenario 2					
R	Title	Sama	Tidak Sama	Relevansi	Bobot
1	Tongkol suwir (pampis) asem manis	8	3	R	72,72727
2	Oseng kambing kecap asam manis simple	7	3	R	70
3	Ayam suwir Bali	9	7	R	56,25
4	Ikan Kembung Sarden	6	3	R	66,66667
5	Sup Ikan Patin	11	6	R	64,70588
6	Ikan tongkol balado	10	4	R	71,42857
7	Tongkol suwir pedas manis ^{oY}	9	4	R	69,23077
8	Oseng ati ampela ayam	6	6	R	50
9	Ayam Sambel	7	3	R	70
10	Balado Ikan pari asap	9	3	R	75

Skenario 3					
R	Title	Sama	Tidak Sama	Relevansi	Bobot
1	Tongkol suwir (pampis) asem manis	8	3	R	72,72727
2	Oseng kambing kecap asam manis simple	7	3	R	70
3	Ayam suwir Bali	9	7	R	56,25
4	Oseng ati ampela ayam	6	6	R	50
5	Ikan tongkol balado	10	4	R	71,42857
6	Ikan Kembung Sarden	6	3	R	66,66667
7	Sup Ikan Patin	11	6	R	64,70588
8	Balado Ikan pari asap	9	3	R	75

9	Tongkol suwir pedas manisðY”	9	4	R	69,23077
10	Ayam Sambel	7	3	R	70

Skenario 4					
R	Title	Sama	Tidak Sama	Relevansi	Bobot
	IKAN GURAME SAMBEL				
1	JELETOT.... BY. Dapoer Linggar	10	3	R	76,92308
2	Ikan tongkol cuwe cabe ijo	11	4	R	73,33333
3	Ikan psmol nila	7	2	R	77,77778
4	Balado Ikan pari asap	9	3	R	75
5	Tongkol suwir (pampis) asem manis	8	3	R	72,72727
6	Sup Ikan Patin	11	6	R	64,70588
7	Pindang asem tongkol	6	1	R	85,71429
8	Ikan patin asam padeh (pedas)	8	3	R	72,72727
9	Ikan tongkol balado	10	4	R	71,42857
10	Patin Masak Kuning Belimbing Wuluh	12	7	R	63,15789

Skenario 5					
R	Title	Sama	Tidak Sama	Relevansi	Bobot
	IKAN GURAME SAMBEL				
1	JELETOT.... BY. Dapoer Linggar	10	3	R	76,92308
2	Ikan tongkol cuwe cabe ijo	11	4	R	73,33333
3	Ikan psmol nila	7	2	R	77,77778
4	Pindang asem tongkol	6	1	R	85,71429
5	Tongkol suwir (pampis) asem manis	8	3	R	72,72727
6	Balado Ikan pari asap	9	3	R	75
7	Ikan tongkol balado	10	4	R	71,42857
8	Sup Ikan Patin	11	6	R	64,70588
9	Ikan patin asam padeh (pedas)	8	3	R	72,72727
10	Patin Masak Kuning Belimbing Wuluh	12	7	R	63,15789

Hasil Rekomendasi Pengujian Query 7

Skenario 1					
R	Title	Sama	Tidak Sama	Relevansi	Bobot
1	Ikan Tongkol Suwir Pedas	9	7	R	56,25
2	Mujair saus pedas manis	7	8	TR	46,66667
3	Gurame panggang teplon bumbu padang*	7	9	TR	43,75
4	Mangut Ikan Gabus	9	10	TR	47,36842
5	Semur ayam kentang simple	8	13	TR	38,09524
6	Ikan Gurame Acar Kuning	6	12	TR	33,33333
7	Bali ikan Bandeng	6	7	TR	46,15385
8	Gulai ikan gabus panggang	7	7	R	50
9	Patin Masak Kuning Belimbing Wuluh	8	11	TR	42,10526
10	Marak (Sop Kambing Khas Timur Tengah)	4	21	TR	16

Skenario 2					
R	Title	Sama	Tidak Sama	Relevansi	Bobot
1	Ikan nila goreng praktis	4	3	R	57,14286
2	Ikan Tongkol Suwir Pedas	9	7	R	56,25
3	Mangut Ikan Gabus	9	10	TR	47,36842
4	Ayam sisit sambal matah	5	3	R	62,5
5	Patin Masak Kuning Belimbing Wuluh	8	11	TR	42,10526
6	Mujair saus pedas manis	7	8	TR	46,66667
7	Lele kuah pedas	10	8	R	55,55556
8	Semur ayam kentang simple	8	13	TR	38,09524
9	Bali ikan Bandeng	6	7	TR	46,15385
10	Ikan bawal sambel mata+kecap ala bumill aziipppp	5	3	R	62,5

Skenario 3					
R	Title	Sama	Tidak Sama	Relevansi	Bobot
1	Ikan nila goreng praktis	4	3	R	57,14286
2	Ikan Tongkol Suwir Pedas	9	7	R	56,25
3	Lele kuah pedas	10	8	R	55,55556
4	Ayam sisit sambal matah	5	3	R	62,5
5	Mangut Ikan Gabus	9	10	TR	47,36842
6	Mujair saus pedas manis	7	8	TR	46,66667
7	Ikan bawal sambel mata+kecap ala bumill aziipppp	5	3	R	62,5

8	Patin Masak Kuning Belimbing Wuluh	8	11	TR	42,10526
9	Kripik kulit ayam By bubu	6	7	TR	46,15385
10	Bali ikan Bandeng	6	7	TR	46,15385

Skenario 4					
R	Title	Sama	Tidak Sama	Relevansi	Bobot
1	Tenggiri Rica Rica	8	3	R	72,72727
2	Sarden homemade simple	6	2	R	75
3	Asam padeh ikan tuna	9	5	R	64,28571
4	Nasi kebuli kambing	3	9	TR	25
5	Kari kambing ala bundo	12	15	TR	44,44444
6	Kari kambing ala bundo	12	15	TR	44,44444
7	Ikan kembung balado maknyos	7	6	R	53,84615
8	Tengkleng kambing	9	11	TR	45
9	Lele kuah pedas	10	8	R	55,55556
10	Ikan Tongkol Suwir Pedas	9	7	R	56,25

Skenario 5					
R	Title	Sama	Tidak Sama	Relevansi	Bobot
1	Tenggiri Rica Rica	8	3	R	72,72727
2	Sarden homemade simple	6	2	R	75
3	Asam padeh ikan tuna	9	5	R	64,28571
4	Nasi kebuli kambing	3	9	TR	25
5	Kari kambing ala bundo	12	15	TR	44,44444
6	Kari kambing ala bundo	12	15	TR	44,44444
7	Ikan kembung balado maknyos	7	6	R	53,84615
8	Tengkleng kambing	9	11	TR	45
9	Tongkol balado simpel	5	3	R	62,5
10	Lele kuah pedas	10	8	R	55,55556

Hasil Rekomendasi Pengujian Query 8

Skenario 1					
R	Title	Sama	Tidak Sama	Relevansi	Bobot
1	Ayam goreng lengkuas kremes	8	5	R	61,53846
2	Opor Ayam	10	6	R	62,5
3	Tongseng ayam	13	8	R	61,90476
4	Tongseng Kambing	16	11	R	59,25926
5	Ikan kembung balado	9	7	R	56,25
6	Asem Asem Patin	9	10	TR	47,36842
7	Nasi kebuli kambing	4	8	TR	33,33333
8	Opor Ayam Pedas	11	11	R	50
9	Ayam Goreng Bumbu Ungkep	5	5	R	50
10	Gulai Kambing	14	9	R	60,86957

Skenario 2					
R	Title	Sama	Tidak Sama	Relevansi	Bobot
1	Terik daging kambing simpel	12	5	R	70,58824
2	Kambing goreng kecap	5	1	R	83,33333
3	Sop tulang kambing	7	3	R	70
4	Tongseng Kambing Praktis	10	1	R	90,90909
5	Oseng kambing kecap asam manis simple	8	2	R	80
6	BakaRica Kambing	7	3	R	70
7	Nila Pedas Manis	10	5	R	66,66667
8	Tongseng kambing	15	12	R	55,55556
9	Sate Kambing (#Bumbu Manis)	8	1	R	88,88889
10	Tumis kambing kecap	10	2	R	83,33333

Skenario 3					
R	Title	Sama	Tidak Sama	Relevansi	Bobot
1	Terik daging kambing simpel	12	5	R	70,58824
2	Kambing goreng kecap	5	1	R	83,33333
3	Sop tulang kambing	7	3	R	70
4	Oseng kambing kecap asam manis simple	8	2	R	80
5	Tongseng Kambing Praktis	10	1	R	90,90909
6	BakaRica Kambing	7	3	R	70
7	Sate Kambing (#Bumbu Manis)	8	1	R	88,88889
8	Goreng kambing simple	4	0	R	100
9	Tumis kambing kecap	10	2	R	83,33333
10	Nila Pedas Manis	10	5	R	66,66667

Skenario 4					
R	Title	Sama	Tidak Sama	Relevansi	Bobot
1	Pete Kambing Kecap	8	0	R	100
2	Tumis kambing kecap	10	2	R	83,33333
3	Tongseng Kambing Praktis	10	1	R	90,90909
4	Tongseng Kambing (tanpa santan)	15	7	R	68,18182
5	Tongseng Kambing Tanpa Santan	15	7	R	68,18182
6	Tongseng Kambing	18	0	R	100
7	Tongseng kambing tanpa santan	15	7	R	68,18182
8	Tongseng kambing	15	12	R	55,55556
9	Gulai kambing	9	5	R	64,28571
10	Sate Kambing (#Bumbu Manis)	8	1	R	88,88889

Skenario 5					
R	Title	Sama	Tidak Sama	Relevansi	Bobot
1	Pete Kambing Kecap	8	0	R	100
2	Tongseng Kambing Praktis	10	1	R	90,90909
3	Tumis kambing kecap	10	2	R	83,33333
4	Tongseng Kambing (tanpa santan)	15	7	R	68,18182
5	Tongseng Kambing Tanpa Santan	15	7	R	68,18182
6	Tongseng Kambing	18	0	R	100
7	Sate Kambing (#Bumbu Manis)	8	1	R	88,88889
8	Tongseng kambing tanpa santan	15	7	R	68,18182
9	Tongseng kambing	15	12	R	55,55556
10	Nasi kebuli kambing	4	8	TR	33,33333

Hasil Rekomendasi Pengujian Query 9

Skenario 1					
R	Title	Sama	Tidak Sama	Relevansi	Bobot
1	Sop kambing	12	9	R	57,14286
2	SOP Kambing Muda	9	11	TR	45
3	Soto bening ayam	8	13	TR	38,09524
4	Tongseng Kambing	8	10	TR	44,44444
5	Sop Iga Kambing	10	14	TR	41,66667
6	Nasi Kabsah kambing #pr_asianfood	9	10	TR	47,36842
7	Tongseng Kambing	8	10	TR	44,44444
8	Kambing masak kecap pedas	5	7	TR	41,66667
9	Tongseng Kambing	8	10	TR	44,44444
10	Tongseng Kambing Jogja (otentik warisan mertua)	8	24	TR	25

Skenario 2					
R	Title	Sama	Tidak Sama	Relevansi	Bobot
1	Sop tulang kambing	7	3	R	70
2	Sop kambing	12	9	R	57,14286
3	Sop ikan tenggiri	4	1	R	80
4	Sop kambing kuah bening	7	5	R	58,33333
5	Sayur Sutage (sop tahu ceker)	8	7	R	53,33333
6	Kambing goreng kecap	4	2	R	66,66667
7	Soup tulang kambing 𐄂𐄂 𐄂𐄂	8	6	R	57,14286
8	Kepala Kakap Masak Kuning	6	4	R	60
9	Bubur oatmeal kambing/syurbah lahm	7	5	R	58,33333
10	Perkedel Tempe	4	7	TR	36,36364

Skenario 3					
R	Title	Sama	Tidak Sama	Relevansi	Bobot
1	Sop tulang kambing	7	3	R	70
2	Sop kambing	12	9	R	57,14286
3	Sop ikan tenggiri	4	1	R	80
4	Sop kambing kuah bening	7	5	R	58,33333
5	Sayur Sutage (sop tahu ceker)	8	7	R	53,33333
6	Kambing goreng kecap	4	2	R	66,66667
7	Lele sambal kemangi delecious	4	3	R	57,14286
8	Soup tulang kambing 𐄂𐄂 𐄂𐄂	8	6	R	57,14286

9	Bubur oatmeal kambing/syurbah lahm	7	5	R	58,33333
10	Kepala Kakap Masak Kuning	6	4	R	60

Skenario 4					
R	Title	Sama	Tidak Sama	Relevansi	Bobot
1	Sop ikan tenggiri	4	1	R	80
2	Sop tulang kambing	7	3	R	70
3	Tongkol suwir	5	2	R	71,42857
4	Perkedel Tempe	4	7	TR	36,36364
5	Soup tulang kambing δY \square^2	8	6	R	57,14286
6	Sop kambing	12	9	R	57,14286
7	Kambing goreng kecap	4	2	R	66,66667
8	Sop kambing	12	9	R	57,14286
9	Sup ayam pedas	11	9	R	55
10	Ikan mas terbang sambal dabu-dabu	5	3	R	62,5

Skenario 5					
R	Title	Sama	Tidak Sama	Relevansi	Bobot
1	Sop ikan tenggiri	4	1	R	80
2	Sop tulang kambing	7	3	R	70
3	Tongkol suwir	5	2	R	71,42857
4	Perkedel Tempe	4	7	TR	36,36364
5	Soup tulang kambing δY \square^2	8	6	R	57,14286
6	Kambing goreng kecap	4	2	R	66,66667
7	Sop kambing	12	9	R	57,14286
8	Sop kambing	12	9	R	57,14286
9	Ikan mas terbang sambal dabu-dabu	5	3	R	62,5
10	Sup ayam pedas	11	9	R	55

Hasil Rekomendasi Pengujian Query 10

Skenario 1					
R	Title	Sama	Tidak Sama	Relevansi	Bobot
1	Gulai ikan tongkol	9	7	R	56,25
2	Tongseng Kambing (tanpa santan)	11	11	R	50
3	Ayam Sambel	6	4	R	60
4	Tongseng kambing tanpa santan	12	10	R	54,54545
5	Ikan Patin pedas manis	4	8	TR	33,33333
6	IKAN GURAME SAMBEL JELETOT.... BY. Dapoer Linggar	7	6	R	53,84615
7	Kepala Ikan Bawal Kuah Kuning	8	11	TR	42,10526
8	Rica Rica Ayam 2	6	11	TR	35,29412
9	Pesmol gembung (non msg)	6	6	R	50
10	Ikan tongkol segar kuah merah	12	5	R	70,58824

Skenario 2					
R	Title	Sama	Tidak Sama	Relevansi	Bobot
1	Ikan tongkol balado	9	5	R	64,28571
2	Ayam Kecap ulalaaa	6	3	R	66,66667
3	Balado kentang dan ati kambing	6	2	R	75
4	Tongkol Balado Rawit	8	2	R	80
5	Ikan tongkol segar kuah merah	12	5	R	70,58824
6	Bledak kambing	5	4	R	55,55556
7	Tumis terong+ sarden	5	4	R	55,55556
8	Sup Patin Gurih	8	7	R	53,33333
9	Ikan pesmol nila	6	3	R	66,66667
10	Tongkol suwir (pampis) asem manis	6	5	R	54,54545

Skenario 3					
R	Title	Sama	Tidak Sama	Relevansi	Bobot
1	Ikan tongkol balado	9	5	R	64,28571
2	Ayam Kecap ulalaaa	6	3	R	66,66667
3	Tongkol Balado Rawit	8	2	R	80
4	Balado kentang dan ati kambing	6	2	R	75
5	Ikan tongkol segar kuah merah	12	5	R	70,58824
6	Tumis terong+ sarden	5	4	R	55,55556
7	Sup Patin Gurih	8	7	R	53,33333
8	Gulai ikan tongkol	9	7	R	56,25
9	Bledak kambing	5	4	R	55,55556

10	Ayam Sambel	6	4	R	60
----	-------------	---	---	---	----

Skenario 4					
R	Title	Sama	Tidak Sama	Relevansi	Bobot
1	Ikan tongkol segar kuah merah	12	5	R	70,58824
2	Tongkol Balado Rawit	8	2	R	80
3	Tongseng kambing	13	14	TR	48,14815
4	Kambing mercon ala ovi	6	3	R	66,66667
5	Balado kentang dan ati kambing	6	2	R	75
6	Tongkol sarden pete uendesss	7	3	R	70
7	Mangut Lele (tanpa santan)	8	4	R	66,66667
8	Rica-Rica Ayam	10	6	R	62,5
9	Ikan Mujair Sambal Pete	10	7	R	58,82353
10	Gule kambing pedas #keto	11	10	R	52,38095

Skenario 5					
R	Title	Sama	Tidak Sama	Relevansi	Bobot
1	Tongkol Balado Rawit	8	2	R	80
2	Ikan tongkol segar kuah merah	12	5	R	70,58824
3	Tongseng kambing	13	14	TR	48,14815
4	Kambing mercon ala ovi	6	3	R	66,66667
5	Balado kentang dan ati kambing	6	2	R	75
6	Tongkol sarden pete uendesss	7	3	R	70
7	Gule kambing pedas #keto	11	10	R	52,38095
8	Mangut Lele (tanpa santan)	8	4	R	66,66667
9	Rica-Rica Ayam	10	6	R	62,5
10	Ikan Gurame Acar Kuning	10	8	R	55,55556

Perhitungan Average Precision Skenario 1

	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
R1						1	1	1	1	1
R2		0,5				1		1		1
R3	0,333		0,333	0,333		1		1		1
R4			0,5	0,5		1		1		1
R5	0,4		0,6			1		1		
R6	0,5	0,333			0,167	1				0,833
R7						1				
R8	0,5	0,375	0,5				0,25	0,75		
R9			0,556		0,222	0,889		0,778		0,667
R10						0,9		0,8		0,7

Perhitungan Average Precision Skenario 2

	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
R1	1	1	1		1	1	1	1	1	1
R2	1	1	1	0,5	1	1	1	1	1	1
R3	1	1	1			1		1	1	1
R4	1	1	1		0,75	1	0,75	1	1	1
R5	1	1	1		0,8	1		1	1	1
R6	1			0,333	0,833	1		1	1	1
R7	1	0,857	0,857	0,429	0,857	1	0,571	1	1	1
R8	1	0,875	0,875	0,5	0,875	1		1	1	1
R9		0,889	0,889		0,889	1		1	1	1
R10	0,9	0,9	0,9	0,5	0,9	1	0,5	1		1

Perhitungan Average Precision Skenario 3

	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
R1	1	1	1		1	1	1	1	1	1
R2	1	1	1	0,5	1	1	1	1	1	1
R3	1	1	1			1	1	1	1	1
R4	1	1	1		0,75	1	1	1	1	1
R5	1	1		0,4	0,8	1		1	1	1
R6	1	1	0,833	0,5	0,833	1		1	1	1
R7	1		0,857	0,571	0,857	1	0,714	1	1	1
R8		0,875	0,875		0,875	1		1	1	1
R9	0,889	0,889	0,889	0,556	0,889	1		1	1	1
R10	0,9	0,9	0,9		0,9	1		1	1	1

Perhitungan Average Precision Skenario 4

	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
R1	1	1	1	1		1	1	1	1	1
R2	1	1	1	1	0,5	1	1	1	1	1
R3	1	1	1	1	0,667	1	1	1	1	
R4	1	1	1	1	0,75	1		1		0,75
R5		1	1	1	0,8	1		1	0,8	0,8
R6	0,833	1	1	1	0,833	1		1	0,833	0,833
R7	0,857	1	1	1	0,857	1	0,571	1	0,857	0,857
R8	0,875	1	1	1		1		1	0,875	0,875
R9	0,889	1	1		0,778	1	0,556	1	0,889	0,889
R10		1	1	0,9	0,8	1	0,6	1	0,9	0,9

Perhitungan Average Precision Skenario 5

	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
R1	1	1	1	1	1	1	1	1	1	1
R2	1	1	1	1		1	1	1	1	1
R3		1	1	1	0,667	1	1	1	1	
R4	0,75	1	1	1	0,75	1		1		0,75
R5	0,8	1	1	1		1		1	0,8	0,8
R6	0,833	1	1	1	0,667	1		1	0,833	0,833
R7	0,857	1	1	1	0,714	1	0,571	1	0,857	0,857
R8	0,875	1	1	1	0,75	1		1	0,875	0,875
R9	0,889	1	1		0,778	1	0,556	1	0,889	0,889
R10		1	1	0,9	0,8	1	0,6		0,9	0,9