



ITS
Institut
Teknologi
Sepuluh Nopember

TUGAS AKHIR - 141501

RANCANG BANGUN SISTEM PENCARIAN DALAM APLIKASI HALAL NUTRITION FOOD MENGGUNAKAN ALGORITMA OKAPI BM25F

DESIGN IN-APP SEARCH SYSTEM IN HALAL NUTRITION FOOD APPLICATIONS USING OKAPI BM25F ALGORITHM

ADNAN MAULUDIN FAJRIYADI
NRP 5213100103

Dosen Pembimbing
Nur Aini Rakhmawati, S.Kom., M.Sc.Eng., Ph.D

DEPARTEMEN SISTEM INFORMASI
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember
Surabaya, 2017

Halaman ini sengaja dikosongkan

TUGAS AKHIR - 141501

RANCANG BANGUN SISTEM Pencarian dalam Aplikasi HALAL NUTRITION FOOD menggunakan ALGORITMA OKAPI BM25F

ADNAN MAULUDIN FAJRIYADI

NRP 5213100103

Dosen Pembimbing

Nur Aini Rakhmawati, S.Kom., M.Sc.Eng., Ph.D

DEPARTEMEN SISTEM INFORMASI

Fakultas Teknologi Informasi

Institut Teknologi Sepuluh Nopember

Surabaya, 2017

Halaman ini sengaja dikosongkan

UNDERGRADUATE THESIS - 141501

**DESIGN IN-APP SEARCH SYSTEM IN HALAL NUTRI-
TION FOOD APPLICATIONS USING OKAPI BM25F AL-
GORITHM**

ADNAN MAULUDIN FAJRIYADI

NRP 5213100103

Supervisor

Nur Aini Rakhmawati, S.Kom., M.Sc.Eng., Ph.D

DEPARTMENT OF INFORMATION SYSTEM

Faculty of Information Technology

Institut Teknologi Sepuluh Nopember

Surabaya, 2017

Halaman ini sengaja dikosongkan

LEMBAR PENGESAHAN
RANCANG BANGUN SISTEM Pencarian Dalam
APLIKASI HALAL NUTRITION FOOD MENGGUNAKAN
ALGORITMA OKAPI BM25F

TUGAS AKHIR

Diajukan Guna Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada

Bidang Studi Akuisisi Data dan Diseminasi Informasi
Program Studi S1 Departemen Sistem Informasi
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember

Oleh :

ADNAN MAULUDIN FAJRIYADI
NRP: 5213100103

Surabaya, Juli 2017

KEPALA
DEPARTEMEN SISTEM INFORMASI

DEPARTEMEN
SISTEM INFORMASI
Dr. Ir. Aris Cahyanto, M.Kom.
NIP. 19650310 199102 1 001

Halaman ini sengaja dikosongkan

LEMBAR PERSETUJUAN

**RANCANG BANGUN SISTEM Pencarian Dalam
Aplikasi Halal Nutrition Food Menggunakan
Algoritma Okapi BM25F**

TUGAS AKHIR

**Diajukan Guna Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada**

**Bidang Studi Akuisisi Data dan Diseminasi Informasi
Program Studi S1 Departemen Sistem Informasi
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember**

Oleh :

**ADNAN MAULUDIN FAJRIYADI
NRP: 5213100103**

Disetujui Tim Penguji: Tanggal Ujian: 10 Juli 2017

Periode Wisuda: September 2017

**Nur Aini Rakhmawati, S.Kom., M.Sc.Eng., Ph.D (Pembimbing
1)**

Renny Pradina Kusumawardani, S.T., M.T.

(Penguji 1)

Radityo Prasetyanto Wibowo, S.Kom., M.Kom.

(Penguji 2)

Halaman ini sengaja dikosongkan

RANCANG BANGUN SISTEM PENCARIAN DALAM APLIKASI HALAL NUTRITION FOOD MENGGUNAKAN ALGORITMA OKAPI BM25F

Nama : ADNAN MAULUDIN FAJRIYADI
NRP : 5213100103
Departemen : Sistem Informasi FTIf
Pembimbing I : Nur Aini Rakhmawati, S.Kom., M.Sc.Eng., Ph.D

Abstrak

Di Indonesia, Lembaga Pengkajian Pangan Obat-obatan dan Kosmetik (LPPOM) MUI adalah lembaga resmi yang menyediakan informasi mengenai halal atau tidaknya suatu produk. Namun informasi tersebut masih sulit diakses oleh pengguna yang termasuk masyarakat awam. Halal Nutrition Food adalah aplikasi yang memiliki fungsi untuk memudahkan pencarian produk halal yang disatukan dalam database yang terintegrasi dalam bentuk Linked Open Data. Untuk melakukan pencarian produk pada aplikasi menggunakan kueri SPARQL. Namun hasil yang didapatkan dari hasil pencarian kurang maksimal jika data yang dimiliki besar. Salah satu metode pencarian yang populer adalah BM25. BM25 memberikan ranking pada dokumen dengan menghitung kemunculan tiap term dalam kueri dalam setiap dokumen sehingga hasil yang didapatkan lebih relevan dengan kueri pencarian. Sementara itu penelitian ini menggunakan metode ekstensi dari BM25 yaitu BM25F. BM25F dapat memproses dokumen terstruktur seperti instance pada RDF. Penelitian ini berfokus pada penambahan fungsi pencarian menggunakan algoritma BM25F sehingga didapatkan hasil pencarian produk yang relevan dengan kueri yang dimasukkan pengguna. Perhitungan BM25F dilakukan dalam Apache Solr yang di dalamnya menghitung nilai pencarian suatu produk didasarkan pada nilai bobot tiap term dalam dokumen.

Kata kunci: Produk Halal, Perangkat lunak, Linked Data, BM25F, Apache Solr.

DESIGN IN-APP SEARCH SYSTEM IN HALAL NUTRITION FOOD APPLICATIONS USING OKAPI BM25F ALGORITHM

Name : ADNAN MAULUDIN FAJRIYADI
NRP : 5213100103
Major : Information System FTIf
Supervisor I : Nur Aini Rakhmawati, S.Kom., M.Sc.Eng., Ph.D

Abstract

In Indonesia, the Institute of Food and Drug Administration (LP-POM) MUI is an official institution that provides information about whether or not a product is legal. But the information is still difficult to access by users in general. Halal Nutrition Food is an application that has a function to facilitate the search for halal products that are incorporated in an integrated database in the form of Linked Open Data. To search a product in an app using a SPARQL query. But the results obtained from search results less than the maximum if the data held large. One of the popular search methods is BM25. BM25 ranks the document by calculating the appearance of each term in the query in each document so that the results obtained are more relevant to the search query. Meanwhile, this research use extension method from BM25 that is BM25F. BM25F can process structured documents such as instances in RDF. This study focuses on adding search functionality using the BM25F algorithm to obtain relevant product search results with user-entered queries. The calculation of BM25F is done in Apache Solr which it calculates the search value of a product based on the weight value of each term in the document.

Keywords: Halal Product, Software, Linked Data, BM25F.

Halaman ini sengaja dikosongkan

KATA PENGANTAR

Puji syukur kehadiran Allah SWT yang telah memberikan nikmat, anugerah, serta hidayah kepada penulis sehingga dapat menyelesaikan tugas akhir dengan judul “*Rancang Bangun Sistem Pencarian Dalam Aplikasi Halal Nutrition Food Menggunakan Algoritma Okapi BM25F*” sebagai salah satu syarat kelulusan pada Departemen Sistem Informasi, Fakultas Teknologi Informasi, Institut Teknologi Sepuluh Nopember Surabaya.

Shalawat serta salam semoga tercurahkan kepada Nabi Muhammad SAW beserta keluarga, para sahabat dan umatnya yang setia hingga akhir zaman.

Penyusunan tugas akhir ini senantiasa mendapatkan dukungan dari berbagai pihak. Untuk itu, secara khusus penulis ingin menyampaikan ucapan terima kasih yang sedalam-dalamnya kepada:

1. Segenap keluarga besar terutama kedua orang tua dan adik-adik penulis, Bapak Agus Mulyadi, Ibu Supriyaningsih, Mutiara Nur Rahmadhanty, dan Muhammad Hafizhul Haq yang senantiasa mendoakan, memberikan motivasi sehingga penulis mampu menyelesaikan pendidikan S1 ini dengan baik.
2. Bapak Dr. Ir. Aris Tjahyanto, M.Kom., selaku Kepala Departemen Sistem Informasi ITS, Bapak Nisfu Asrul Sani, S.Kom, M.Sc selaku KaProdi S1 Sistem Informasi ITS serta seluruh dosen pengajar beserta staf dan karyawan di Departemen Sistem Informasi, FTIF ITS Surabaya selama penulis menjalani kuliah
3. Ibu Nur Aini Rakhmawati, S.Kom., M.Sc., Eng. Ph.D selaku dosen pembimbing yang telah banyak meluangkan waktu untuk membimbing, mengarahkan, dan mendukung dengan memberikan ilmu, petunjuk, dan motivasi dalam penyelesaian

an Tugas Akhir.

4. Ibu Mahendrawathi ER, ST., MSc., PhD. sebagai dosen wali penulis selama menempuh pendidikan di Departemen Sistem Informasi.
5. Bapak Radityo Prasetianto Wibowo, S.Kom, M.Kom serta Ibu Renny Pradina Kusumawardani, S.T., M.T. selaku dosen penguji yang telah memberikan kritik, saran, dan masukan yang dapat menyempurnakan Tugas Akhir ini.
6. Keluarga Majelis Dzikirullah "Assalamu'alaikum" yang selalu mendoakan dan memberikan motivasi kepada penulis selama perkuliahan dan penyusunan tugas akhir.
7. Teman-teman Sistem Informasi angkatan 2013 (13ELTRANIS) yang senantiasa menemani dan memberikan motivasi bagi penulis selama perkuliahan hingga dapat menyelesaikan tugas akhir.
8. Rekan-rekan Badan Eksekutif Mahasiswa (BEM) ITS mulai dari kepengurusan Kolaborasi 2014/2015, Berani 2015/2016, dan Wahana Juang 2016/2017 yang telah memberikan semangat serta motivasi.
9. Rekan-rekan Kementerian Kominfo BEM ITS selama 3 periode mulai dari Kominfun, Kominstrong, dan Kominrangers.
10. Serta pihak-pihak lain yang tidak dapat disebutkan satu per satu yang telah banyak membantu penulis selama perkuliahan hingga dapat menyelesaikan tugas akhir ini.

Terima kasih atas segala bantuan, dukungan, serta doanya. Semoga Allah SWT senantiasa melimpahkan anugerah serta membalas kebaikan yang telah diberikan kepada penulis.

Penulis menyadari bahwa masih terdapat kekurangan dalam penyusunan tugas akhir ini, oleh karena itu penulis mengharapkan saran dan kritik yang membangun demi kebaikan penulis dan tugas akhir ini. Akhir kata, penulis berharap agar tugas akhir ini dapat memberikan kebermanfaatan.

DAFTAR ISI

ABSTRAK	xi
ABSTRACT	xiii
KATA PENGANTAR	xv
DAFTAR ISI	xvii
DAFTAR TABEL	xxi
DAFTAR GAMBAR	xxii
DAFTAR KODE	xxiii
1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Perumusan Masalah	3
1.3 Batasan Masalah	3
1.4 Tujuan Tugas Akhir	3
1.5 Manfaat Tugas Akhir	4
1.6 Relevansi Tugas Akhir	4

2	TINJAUAN PUSTAKA	5
2.1	Penelitian Sebelumnya	5
2.1.1	Rancang Bangun Perangkat Lunak Linked Open Data Halal Dan Gizi Pada Produk Ma- kanan Dan Minuman	5
2.1.2	Tuning Parameter dalam Fungsi Okapi BM25 pada Mesin Pencari Teks Bahasa Indonesia	6
2.1.3	Using BM25F for Semantic Search	6
2.1.4	Integrating the Probabilistic Model BM25/BM25F into Lucene	7
2.1.5	Sindice at semsearch 2010	8
2.2	Dasar Teori	8
2.2.1	Produk Halal	9
2.2.2	OKAPI BM25F	10
2.2.3	Semantic Web	11
2.2.4	Linked Data	12
2.2.5	RDF	13
2.2.6	Semantic Search	14
2.2.7	Apache Lucene dan Apache Solr	14
2.2.8	Plugin BM25F	16
2.2.9	Laravel	18
2.2.10	Solarium <i>Package</i>	19
2.2.11	<i>Analytical Hierarchial Process</i>	19
3	METODOLOGI	23
3.1	Tahapan Pengerjaan Tugas Akhir	23
3.1.1	Studi literatur	24
3.1.2	Pengumpulan data	24
3.1.3	Pengindeksan data	24
3.1.4	BM25F <i>Ranking</i>	24
3.1.5	Pengujian	28
3.1.6	<i>Deployment</i>	29

3.1.7	Pembuatan buku Tugas Akhir	29
3.2	Penjelasan Sistem	30
4	PERANCANGAN	33
4.1	Penyusunan Kuisisioner	33
4.2	Desain Sistem	33
4.3	Desain Basis Data	35
4.4	Desain Skema <i>Solr collection</i>	37
4.5	Desain Antarmuka Pengguna	39
5	IMPLEMENTASI	43
5.1	Lingkungan Implementasi	43
5.2	Implementasi	44
5.2.1	Konfigurasi Server	44
5.2.2	Pengindeksan Data MySQL	45
5.2.3	Perhitungan AHP	47
5.2.4	Pemrosesan Algoritma BM25F	49
5.2.5	Koneksi Apache Solr Dengan Aplikasi	52
6	HASIL DAN PEMBAHASAN	55
6.1	Hasil Pengujian	55
6.1.1	Pengujian Algoritma BM25F	55
6.1.2	Pengujian Kinerja	60
6.2	Pembahasan	61
6.2.1	Pembahasan Uji Algoritma BM25F	61
6.2.2	Pembahasan Uji Kinerja	62
7	KESIMPULAN DAN SARAN	63
7.1	Kesimpulan	63

7.2	Saran	64
	DAFTAR PUSTAKA	65
A	Kuisisioner	67
	BIODATA PENULIS	71

DAFTAR TABEL

2.1	Matriks AHP awal	20
2.2	Contoh hasil perhitungan <i>pairwise-comparison</i> . .	21
2.3	Contoh hasil penjumlahan kriteria	21
2.4	Contoh hasil normalisasi <i>pairwise-comparison</i> . . .	22
2.5	AHP hasil prioritas <i>field</i> awal	22
3.1	Dua instances RDF data produk halal	25
3.2	Skenario Pengujian Fungsional	28
3.3	Skenario Pengujian Non-Fungsional	29
5.1	Spesifikasi perangkat keras	43
5.2	Spesifikasi perangkat lunak	44
5.3	AHP hasil prioritas <i>field</i> awal	48
5.4	AHP hasil prioritas <i>field</i> setelah normalisasi	48
6.1	Penjelasan nilai untuk <i>entity</i> hasil pencarian teratas	56
6.2	Hasil pencarian dengan varian kueri	60
6.3	Hasil pengujian performa aplikasi berdasarkan kueri	61

DAFTAR GAMBAR

2.1	Layer aplikasi dengan Lucene	15
2.2	Struktur Project Plugin BM25F	17
3.1	Metodologi Tugas Akhir	23
4.1	Desain sistem dalam aplikasi	34
4.2	Skema basis data yang digunakan dalam aplikasi	36
4.3	Desain skema <i>solr collection</i>	38
4.4	Halaman utama antarmuka pengguna aplikasi	40
4.5	Halaman hasil pencarian dalam aplikasi	41
4.6	Halaman produk dalam aplikasi	41
6.1	Hasil pencarian uji coba algoritma BM25F	56
6.2	Pengujian relevansi hasil pencarian dengan kueri	59

DAFTAR KODE

5.1	<i>field</i> pada <i>schema.xml</i>	45
5.2	Data Import Handler pada collection data halal . . .	46
5.3	File <i>data-config.xml</i>	46
5.4	Query Parser untuk BM25F	49
5.5	Kueri SQL untuk mencari <i>length field</i>	50
5.6	Request Handler untuk BM25F	51
5.7	BM25F Similarity	52
5.8	Koneksi Solarium dengan aplikasi	52
5.9	Konfigurasi Solarium dalam aplikasi	53
6.1	Hasil debug kueri satu term satu field	57

Halaman ini sengaja dikosongkan

BAB 1

PENDAHULUAN

Pada bab pertama ini akan dibahas mengenai dasar utama mengapa tugas akhir ini dibuat yaitu membahas mengenai latar belakang masalah, perumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, dan relevansi terhadap pengerjaan tugas akhir.

1.1 Latar Belakang

Indonesia merupakan salah satu negara dengan mayoritas penduduk muslim terbesar di dunia. Prosentase Muslim Indonesia mencapai hingga 12,7 persen dari populasi dunia. Dari 205 juta penduduk Indonesia, dilaporkan sedikitnya 88,1 persen beragama Islam, [8]. Dalam agama islam terdapat hal-hal yang mengikat penganutnya salah satunya adalah mengenai makanan dan minuman yang dikonsumsi. Dalam agama islam makanan yang boleh di konsumsi adalah makanan yang halal, sehingga pemerintah Indonesia membentuk lembaga khusus untuk mengawasi produk makanan halal untuk masyarakat penganut agama islam di Indonesia.

Lembaga Pengawasan Pangan, Obat-obatan, dan Kosmetika Majelis Ulama Indonesia (LPPOM MUI) adalah lembaga dibawah naungan departemen agama yang mengeluarkan sertifikat halal bagi barang-barang yang dikonsumsi oleh penduduk Indonesia. Selain mengeluarkan sertifikat, LPPOM MUI juga bertanggung jawab atas penyediaan informasi produk halal baik melalui media cetak maupun elektronik. Saat ini LPPOM MUI sudah menyediakan informasi produk halal melalui website yang dapat di akses di <http://halalmui.org> dan juga melalui sms gateway.

Dari situs web LPPOM MUI, pengguna dapat mencari sertifikat halal sesuai dengan produk ataupun perusahaan yang mengajukan sertifikat halal dan juga mengunduhnya dalam bentuk *portable document format (pdf)*. Namun permasalahannya hasilnya masih kurang bagus untuk diberikan kepada masyarakat. Dalam penggunaannya, pengguna hanya dapat melihat nomor sertifikat, produsen, dan masa berlaku sertifikat produk halal sesuai dengan kategori yang disediakan oleh MUI.

Halal Nutrition Food adalah sebuah aplikasi yang dibuat oleh Jaubar Fatawi [4] yang berfungsi untuk memberikan kemudahan kepada pengguna dalam melakukan pencarian produk halal yang telah disertifikasi oleh MUI. Aplikasi ini menampung data produk halal dalam bentuk RDF. Dalam melakukan pencarian data produk halal, aplikasi ini mengirimkan kueri SPARQL ke database sehingga hasil yang didapatkan masih belum relevan sesuai dengan maksud dari kueri yang dimasukkan oleh pengguna.

Salah satu metode pencarian yang populer adalah Okapi BM25. Algoritma BM25 mengurutkan dokumen berdasarkan term kueri yang muncul pada setiap koleksi dokumen. Dari nilai ranking tersebut didapatkan dokumen yang paling relevan ditampilkan di urutan awal hasil pencarian. Dengan cara ini pencarian produk halal yang relevan akan mudah didapatkan.

Tugas akhir ini akan memberikan penambahan fungsi pada aplikasi Halal Nutrition Food yaitu fungsi pencarian produk halal menggunakan metode ranking dengan BM25F. BM25F sendiri adalah algoritma ekstensi dari BM25 yang memiliki kelebihan untuk memproses dokumen terstruktur, contohnya adalah dokumen dengan format RDF. Sehingga penambahan fungsi ini akan bermanfaat untuk mendapatkan informasi produk halal yang relevan sesuai dengan kueri dari pengguna.

1.2 Perumusan Masalah

Berdasarkan uraian latar belakang, maka rumusan permasalahan yang menjadi fokus dan akan diselesaikan dalam Tugas Akhir ini antara lain :

1. Bagaimana pengguna mendapatkan informasi mengenai komposisi gizi serta karakteristik zat pada produk halal secara relevan?
2. Bagaimana menerapkan algoritma BM25F untuk memudahkan pencarian produk halal?

1.3 Batasan Masalah

Batasan-batasan dalam pembuatan Usulan Tugas Akhir adalah sebagai berikut:

1. Hasil akhir dari penelitian ini adalah pengembangan aplikasi web berbasis linked open data yang memiliki fitur pencarian data produk.
2. Jumlah sampel data hanya dibatasi 150 produk halal yang telah mendapatkan sertifikat halal dari LPPOM MUI.
3. Metode pencarian data pada aplikasi web menggunakan algoritma BM25F.

1.4 Tujuan Tugas Akhir

Tujuan dari penelitian ini adalah membuat sistem pencarian produk halal dari aplikasi web yang sudah dibuat pada penelitian sebelumnya yaitu Halal Nutrition Food. Sistem pencarian akan menggu-

nakan metode ranking BM25F yang akan memudahkan dalam menemukan hasil pencarian yang relevan yaitu data produk yang telah diberi label halal oleh MUI.

1.5 Manfaat Tugas Akhir

Manfaat yang diharapkan dapat diperoleh dari tugas akhir ini adalah:

1. Bagi pengguna, dapat dengan mudah mengetahui karakteristik, komposisi, dan gizi dari produk halal pada yang disetujui oleh MUI.
2. Bagi peneliti, penelitian ini akan menghasilkan perangkat lunak berbasis Linked Open Data yang mana ke depannya bisa dikembangkan menjadi lebih baik lagi.

1.6 Relevansi Tugas Akhir

Tugas akhir ini berkaitan dengan pembangunan perangkat lunak dengan metode penyimpanan dan publikasi data berupa Linked Open Data ditambah dengan metode pencarian menggunakan algoritma BM25F. Pengerjaan tugas akhir ini berkaitan dengan mata kuliah Pemrograman Integratif, Pemrograman Berbasis Objek dan Pemrograman Berbasis Web di Departemen Sistem Informasi, Fakultas Teknologi Informasi, Institut Teknologi Sepuluh Nopember.

Tugas akhir ini layak dijadikan sebagai tugas akhir pada tingkat S1, karena memecahkan masalah yaitu dalam mempermudah pencarian informasi terkait produk makanan atau minuman yang halal.

BAB 2

TINJAUAN PUSTAKA

Bab ini akan menjelaskan mengenai penelitian sebelumnya dan dasar teori yang dijadikan acuan atau landasan dalam pengerjaan tugas akhir ini. Dasar teori akan memberikan gambaran secara umum dari landasan penjabaran tugas akhir ini.

2.1 Penelitian Sebelumnya

Pada subbab ini dijelaskan tentang referensi dari penelitian-penelitian yang berkaitan dengan tugas akhir.

2.1.1 Rancang Bangun Perangkat Lunak Linked Open Data Halal Dan Gizi Pada Produk Makanan Dan Minuman

Penelitian ini berjudul "*Rancang Bangun Perangkat Lunak Linked Open Data Halal Dan Gizi Pada Produk Makanan Dan Minuman*" oleh Jauhar Fatawi [4]. Dalam penelitian ini, peneliti membuat sebuah aplikasi web yang bernama Halal Nutrition Food yang memberikan rekomendasi tentang status gizi dari produk makanan atau minuman yang halal. Pada penelitian ini didapatkan dataset produk halal, dataset zat aditif, dan perangkat lunak "*Linked Open Data Halal dan Gizi*". Dataset produk halal menghimpun data-data produk halal secara detail beserta dengan komposisi produk, nutrisi produk, status halal, dan detail zat aditif bila terkandung dalam produk.

Tugas akhir ini akan melanjutkan penelitian tentang aplikasi Ha-

lal Nutrition Food dengan memberikan penambahan fungsi pada sistem pencarian produk halal di dalamnya agar lebih relevan dan akurat.

2.1.2 Tuning Parameter dalam Fungsi Okapi BM25 pada Mesin Pencari Teks Bahasa Indonesia

Penelitian ini berjudul "*Tuning Parameter dalam Fungsi Okapi BM25 pada Mesin Pencari Teks Bahasa Indonesia*" oleh Tedy Saputra [14]. Dalam penelitian ini, peneliti melakukan analisis tuning parameter terhadap parameter bebas dari metode BM25. Tuning parameter sendiri merupakan suatu variabel yang dapat diubah-ubah nilainya sesuai dengan kebutuhan dengan tujuan untuk mendapatkan hasil pencarian yang lebih baik. Tujuan dari penelitian ini adalah seberapa besar pengaruh modifikasi nilai dari tuning parameter yang ada dalam fungsi kesamaan Okapi BM25 terhadap evaluasi dari hasil pencarian.

Keterkaitannya dengan tugas akhir ini adalah pada proses pencarian kesamaan yang menggunakan Okapi BM25 yang juga digunakan sebagai metode pencarian dalam tugas akhir ini.

2.1.3 Using BM25F for Semantic Search

Penelitian ini berjudul "*Using BM25F for Semantic Search*" oleh Perez-Aguera et al [11]. Dalam penelitian ini, peneliti menjabarkan tentang sistem *information retrieval* pada Semantic Web Search. Tujuan utama penelitian ini adalah menganalisa tantangan untuk menerapkan *information retrieval* pada RDF dalam proses pencarian data.

Penelitian ini melakukan pendekatan *inverted index* pada RDF. Con-

toh penggunaannya adalah sebagai berikut:

Entri pada DBpedia:

The Godfather http://dbpedia.org/page/The_Godfather

Francis Ford Coppola http://dbpedia.org/page/Francis_Ford_Coppola

Penjelasan Triples:

The Godfather (Subject) - dbpprop:director (Predicate)- Francis Ford Coppola (Object)

Kata kunci indeks:

Kata "director" dapat digunakan sebagai keyword untuk meng-indeks entri yang dijelaskan dengan URI:

http://dbpedia.org/page/Francis_Ford_Coppola.

Keterkaitan dengan tugas akhir ini adalah penggunaan algoritma BM25F dan menerapkannya dalam Semantic Search yang menjadi metode utama dalam tugas akhir.

2.1.4 Integrating the Probabilistic Model BM25/BM25F into Lucene

Penelitian ini berjudul "*Integrating the Probabilistic Model BM25/BM25F into Lucene*" oleh Pérez-Iglesias et al [12]. Dalam penelitian ini, peneliti menjelaskan tentang penggunaan formula BM25 dalam memproses dokumen teks biasa dan BM25F yang memproses dokumen terstruktur kemudian mengimplementasikannya ke dalam Lucene Java Framework.

Keterkaitan dengan tugas akhir ini adalah penggunaan algoritma BM25F dengan mengimplementasikannya ke Lucene Java Framework, dimana tugas akhir ini juga menggunakan Lucene untuk pemrosesan data.

2.1.5 Sindice at semsearch 2010

Penelitian ini berjudul "*Sindice at semsearch 2010*" oleh Renaud Delbru, Nur Aini Rakhmawati, Giovanni Tummarello [2]. Dalam penelitian ini, peneliti membandingkan dua metode ranking pada *SIREn (Semantic Information Retrieval Engine)* yaitu *query-dependent* dan *query-independent* serta menggabungkan kedua metode tersebut di dalamnya.

Metode yang penulis soroti pada penelitian ini adalah *query-independent* yaitu metode ranking entitas menggunakan algoritma analisis link hirarki [3] yang didasarkan pada dua lapisan model dari Web Data. Lapisan atas (dataset layer) terdiri dari kumpulan antar dataset yang terhubung sedangkan yang lapisan lebih rendah (entitas layer) berisi grafik independen entitas.

Keterkaitan dengan tugas akhir ini adalah penerapan IR pada RDF dimana di dalamnya digunakan metode *query-independent* dan ranking *Weighted Entity Ranking* yang keduanya akan digunakan dalam tugas akhir.

2.2 Dasar Teori

Pada subbab ini akan dijelaskan tentang teori-teori dan istilah yang digunakan dalam penelitian tugas akhir.

2.2.1 Produk Halal

Kata halal berasal dari bahasa Arab yang berarti disahkan, diizinkan, dan diperbolehkan. Pada prinsipnya semua makanan dan minuman yang ada di dunia ini halal untuk dimakan dan diminum kecuali ada larangan dari Allah SWT yaitu yang terdapat dalam Al Qur'an dan yang terdapat dalam hadist Nabi Muhammad SAW.

Dalam sebuah hadits Rasulullah SAW pernah ditanya para sahabat tentang hukum minyak sapi (samin), keju, kulit binatang beserta bulunya untuk perhiasan maupun untuk tempat duduk. Dalam Al-Qur'an anjuran untuk memakan makanan di jelaskan sebagai berikut:

”Dan (Allah) menghalalkan bagi mereka segala yang baik dan mengharamkan bagi mereka segala yang buruk.” (QS al-A'raf [7]: 157).
 ”Hai sekalian manusia, makanlah yang halal lagi baik dari apa yang terbaik di bumi.” (QS al-Baqarah [2]: 168).

Kriteria makanan halal ada 3, yaitu :

- Halal karena dzatnya. Artinya, enda itu memang tidak dilarang oleh hukum syara', seperti nasi, susu, telur, dan lainlain.
- Halal cara mendapatkannya. Artinya sesuatu yang halal itu harus diperoleh dengan cara yang halal pula. Sesuatu yang halal tetapi cara medapatkannya tidak sesuatu dengan hukum syara' maka menjadi haramlah ia. Sebagaimana, mencuri, menipu, dan lain-lain.
- Halal karena proses/cara pengolahannya. Artinya selain sesuatu yang halal itu harus diperoleh dengan cara yang halal pula. Cara atau proses pengolahannya juga harus benar. Hewan, seperti kambing, ayam, sapi, jika disembelih dengan cara yang tidak sesuai dengan hukum Islam maka dagingnya menjadi haram.

2.2.2 OKAPI BM25F

OKAPI (*Online Keyword Access to Public Information*) BM25 adalah model pembobotan hasil pencarian yang dikembangkan oleh City University London berdasarkan pada model probabilistik dasar yang mengurutkan dokumen dalam urutan menurun terhadap nilai relevansi sebuah dokumen terhadap informasi yang dibutuhkan. Okapi BM25 meranking dokumen berdasarkan probabilitas dan menggunakan *term frequency* untuk meranking *similarity*.

Pengerjaan Tugas Akhir ini menggunakan BM25F, yaitu algoritma ekstensi dari BM25. BM25F memiliki kelebihan yaitu fungsi peringkat yang disesuaikan untuk dokumen terstruktur (seperti instance pada RDF). Penggunaan BM25F juga dikarenakan dari hasil penelitian yang sudah ada, probabilitas relevansi dokumen meningkat sebagaimana frekuensi istilah dalam kueri yang semakin banyak. Untuk itu diperlukan fungsi peringkat paling terkini yang menggunakan peningkatan saturasi yang berfungsi untuk pembobotan istilah dalam dokumen yang muncul dalam kueri [13].

Rumus peringkat pada BM25F berdasarkan Pérez-Iglesias [12] adalah sebagai berikut:

$$R(q, d) = \sum_{t \text{ in } q} idf(t) \cdot \frac{weight(t, d)}{k_1 + weight(t, d)} \quad (2.1)$$

Penjelasan:

1. $R(q, d)$ adalah Ranking query q dalam dokumen d (nilai BM25F).
2. t adalah term yang didapatkan dari query.
3. d adalah dokumen, dalam RDF dokumen berarti instance.

4. $idf(t)$ adalah nilai idf yang diperoleh dari perhitungan:

$$idf(t) = \log \frac{N - df(t) + 0.5}{df(t) + 0.5} \quad (2.2)$$

Penjelasan:

- N adalah jumlah seluruh dokumen.
 - $df(t)$ adalah jumlah dokumen yang memiliki term t .
5. $weight(t, d)$ adalah bobot term t dalam dokumen d yang diperoleh dari perhitungan:

$$weight(t, d) = \sum_{c \text{ in } d} \frac{occurs_{t,c}^d \cdot boost_c}{((1 - b_c) + b_c \cdot \frac{l_c}{avl_c})} \quad (2.3)$$

Penjelasan:

- c adalah field pada dokumen yang terdiri dari *subyek-predikat-obyek*.
 - d adalah dokumen, dalam RDF dokumen berarti instance.
 - $occurs_{t,c}^d$ adalah kemunculan term t pada field c dalam dokumen d .
 - $boost_c$ adalah faktor boost yang diberikan ke field c .
 - b_c adalah konstanta berdasarkan panjang field c .
 - l_c adalah panjang dari field c .
 - avl_c adalah rata-rata panjang dari field c .
6. $k1$ adalah konstanta yang dapat mengatur pertumbuhan frekuensi term non-linier.

2.2.3 Semantic Web

Secara etimologi Semantic Web terdiri dari dua kata yaitu *Semantic* yang menurut Kamus Besar Bahasa Indonesia (KBBI) artinya adalah ilmu tentang makna kata dan kalimat; pengetahuan mengenai

seluk-beluk dan pergeseran arti kata. Juga kata *Web* yang berarti World Wide Web (WWW).

Semantic web adalah sebuah fungsi tambahan dari sebuah web dimana memberikan cara yang lebih mudah untuk menemukan, berbagi, menggunakan kembali, dan menggabungkan informasi. Kemampuan ini dibentuk dengan menggabungkan kemampuan teknologi XML untuk membentuk tagging schemes dan RDF's (Resource Description Framework) sebagai pendekatan fleksible yang mewakili data. Semantic web menyediakan format umum untuk pertukaran data. Selain itu Semantic web juga menyediakan bahasa umum untuk merekam bagaimana data berelasi dengan obyek-obyek dunia nyata, memungkinkan orang atau sebuah mesin memulai pada satu database kemudian berhubungan dengan database lain dan terkonseksi satu sama lain [17].

2.2.4 Linked Data

Linked Data adalah langkah yang baik untuk menampilkan dan menghubungkan data terstruktur di dalam Web menggunakan standar yang berlaku menurut *World Wide Web (WWW) Consortium*. Linked Data memperkenalkan cara untuk mengambil, menggunakan kembali dan menampilkan data di Web. Sementara itu pengertian Linked Data menurut W3C, "Untuk membuat Web Data dapat diakses secara mudah, penting untuk memiliki banyak data yang tersedia di Web dalam format standar, terjangkau dan dikelola oleh Semantic Web. Selain itu, tidak hanya akses data Semantic Web, tetapi hubungan antara data yang harus tersedia untuk membuat Web Data. Koleksi dataset yang saling terkait di Web juga dapat disebut sebagai Linked Data" [16]. Linked Data secara umum adalah perbitan data secara terstruktur menggunakan RDF dengan menggunakan URI.

Ada 4 (empat) prinsip linked data [6] :

- Semua item harus diidentifikasi menggunakan URI
- Semua URI harus *dereference-able* artinya dengan menggunakan HTTP, URI membolehkan siapapun (manusia atau mesin) untuk mencari suatu item yang telah diidentifikasi melalui URI
- Mencari suatu URI lebih mengarah kepada data lebih juga dikenal sebagai prinsip *follow-your-nose*
- Link ke URI dalam sekumpulan data lainnya harus dimasukkan untuk mengaktifkan penemuan data (*data discovery*) lebih lanjut

2.2.5 RDF

Resource Description Framework (RDF) adalah kerangka untuk mengambil informasi dari sumber-sumber data. Sumber-sumber tersebut dapat berupa apapun, termasuk dokumen, orang, benda fisik, dan konsep-konsep abstrak. RDF ini muncul saat ini dimana Web perlu di proses oleh aplikasi, bukan hanya untuk ditampilkan. RDF menyediakan framework umum untuk menginformasikan data sehingga dapat dilakukan pertukaran data antar aplikasi tanpa kehilangan makna sebenarnya [5].

RDF data model mirip dengan model konseptual sederhana seperti entity relationship model atau class diagram, namun pada RDF didasarkan pada pembuatan model berdasarkan pernyataan tentang sumber daya / resources (pada web) ke dalam bentuk subyek-predikat-obyek. Bentuk ini dikenal dengan nama *triples* pada terminologi RDF. Subyek menunjukkan sumber daya / resources, predikat menunjukkan ciri-ciri atau aspek sumber daya dan menghubungkan antara subyek dan obyek [10]. Untuk lebih jelasnya dapat dilihat

ilustrasi di bawah ini:

Adnan mengerjakan tugas akhir
(subyek) (predikat) (obyek)

Subyek merupakan suatu hal yang dideskripsikan. Sedangkan obyek merupakan data berupa angka, string, tanggal, ataupun URI dari suatu hal atau benda lain yang memiliki hubungan dengan subjek. Predikat merupakan merupakan suatu URI yang digunakan untuk mendeskripsikan hubungan antara subjek dengan objek. URI dari predikat diambil dari vocabularies, suatu kumpulan URI yang dapat digunakan untuk merepresentasikan informasi terkait bidang tertentu [7].

2.2.6 Semantic Search

Semantic Search adalah teknik data pencarian di mana permintaan pencarian bertujuan untuk tidak hanya menemukan kata kunci, tetapi untuk menentukan maksud dan makna kontekstual dari kata-kata seseorang menggunakan untuk pencarian [15].

Hasil pencarian yang diberikan Semantic Search akan lebih bermakna dengan memahami maksud frase pencarian dan menemukan hasil yang paling relevan dalam repositori data yang dituju. Semantic Search akan memperhatikan sinonim dari istilah, variasi kata dan unsur bahasa alami lainnya sebagai bagian dari pencarian.

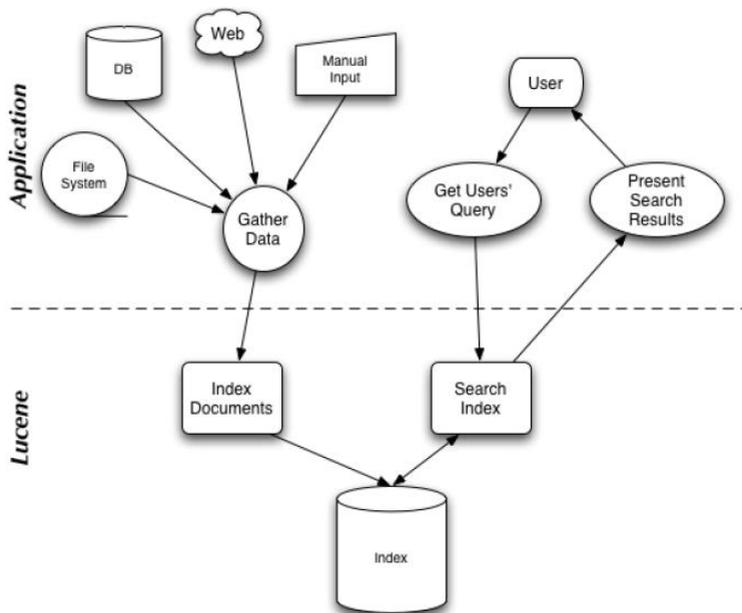
2.2.7 Apache Lucene dan Apache Solr

Apache Lucene adalah *library* pencarian teks di Java yang mempermudah fungsi pencarian dalam aplikasi atau situs web [9]. Lucene

melakukannya dengan menambahkan konten ke indeks pada teks. Lalu dari indeks tersebut memungkinkan untuk melakukan query, kemudian hasil yang didapatkan akan diurutkan berdasarkan peringkat relevansi untuk query atau diurutkan menurut bidang acak seperti tanggal terakhir perubahan pada dokumen.

Lucene mampu melakukan pencarian dengan cepat karena sistem kerjanya bukan mencari teks secara langsung, tetapi mencari indeks dari tiap dokumen. Seperti halnya mencari halaman dalam sebuah buku yang berhubungan dengan kata kunci dengan mencari indeks di bagian belakang buku.

Skema Lucene dengan aplikasi dijelaskan pada Gambar 2.1



Gambar 2.1: Layer aplikasi dengan Lucene

Apache Solr adalah salah satu enterprise search platform yang bersifat open source dari Apache. Fitur utamanya adalah pencarian berbasis teks, *highlight*, *faceted search*, *dynamic clustering*, *database integration*, dan penanganan terhadap *rich document* (word, pdf dsb).

Pencarian menggunakan solr bisa menggunakan URL dan hasil pencariannya bisa berupa xml maupun JSON. Sehingga memudahkan programmer untuk menggunakannya dalam berbagai bahasa pemrograman.

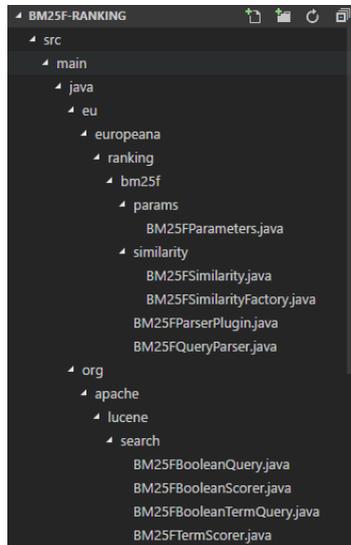
Apache Solr sendiri merupakan peningkatan fungsi dari Apache Lucene yang artinya Apache Solr membutuhkan Apache Lucene untuk melakukan proses di dalamnya. Apache Solr dibuat menggunakan bahasa pemrograman java dan menggunakan lucene untuk fitur indexing dan pencarian.

2.2.8 Plugin BM25F

Plugin BM25F yang digunakan dalam aplikasi Halal Nutrition Food menggunakan kode yang dibuat oleh Diego Cecarelli [1]. Pada project plugin BM25F tersebut terdapat 9 class yang memiliki fungsi masing-masing. Struktur dari project Plugin BM25F yang digunakan dalam aplikasi dapat dilihat pada gambar 2.2

Penjelasan untuk masing-masing class adalah sebagai berikut:

- **BM25FParserPlugin.java**
Class ini berfungsi sebagai class utama yang menghubungkan plugin dengan collection data halal dalam Solr.
- **BM25FQueryParser.java**
Class ini berfungsi untuk melakukan parsing dari kueri yang dimasukkan pengguna, untuk kemudian diproses ke class-



Gambar 2.2: Struktur Project Plugin BM25F

class lain.

- **BM25FParameters.java**
Class ini berfungsi untuk menampung parameter pada plugin yaitu nilai K1, nilai boost, nilai bobot, isi tiap field, dan field utama yang didefinisikan pada collection.
- **BM25FSimilarity.java**
Class ini berfungsi untuk melakukan perhitungan BM25F, yaitu perhitungan TF/IDF, perhitungan weight, perhitungan rata-rata panjang field, dan perhitungan lain yang menunjang algoritma BM25F. Selanjutnya dari hasil perhitungan tersebut akan dihasilkan score relevansi pencarian.
- **BM25FSimilarityFactory.java**
Class ini berisi method aksesor dari class BM25FSimilarity.
- **BM25FBooleanQuery.java**
Class ini berfungsi untuk memroses query yang sesuai de-

ngan dokumen dan cocok dengan kombinasi boolean dari query lain.

- `BM25FBooleanScorer.java`
Class ini berfungsi untuk melakukan Scoring pada Boolean Query dan menghitungnya menggunakan persamaan BM25F.
- `BM25FBooleanTermQuery.java`
Class ini berfungsi untuk memroses query yang hanya mengandung satu term.
- `BM25FBooleanTermScorer.java`
Class ini berfungsi untuk melakukan Scoring pada query yang hanya mengandung satu term dan menghitungnya menggunakan persamaan BM25F.

2.2.9 Laravel

Laravel adalah salah satu framework pemrograman PHP yang dibuat oleh Taylor Otwell. Seperti kebanyakan framework PHP lainnya, Laravel dikembangkan dengan menggunakan struktur Model-View-Controller (MVC). Menurut Sitepoint.com merupakan framework terpopuler pada survey yang dilakukannya pada tahun 2015. Pada penelitian ini menggunakan framework ini dikarenakan beberapa alasan berikut:

1. RESTful routing
RESTful adalah cara baru dalam mengelola *request*. Dengan REST dapat dengan mudah mengelola metode-metode *request* seperti *GET*, *POST*, *PUT*, *PATCH*, *DELETE*, *STORE*, dll.
2. Composer ready
Composer sendiri adalah *Dependency Management* PHP yang membantu kita untuk mencari library yang akan dipakai dan menginstalnya. Dalam melakukan instalasi laravel pun juga perlu menggunakan Composer.

3. **Template engine**
Templating engine adalah program yang mengubah sintaks *template engine* tersebut ke HTML. Laravel sendiri memiliki *templating engine* bernama Blade.
4. Selain itu laravel memiliki beberapa fitur lain seperti SSH, authentication, pagination, session, schema suilder, validator, dan session.

2.2.10 Solarium Package

Solarium adalah package PHP untuk Laravel yang digunakan untuk menghubungkan aplikasi web yang dibuat menggunakan PHP dengan Apache Solr.

Beberapa kelebihan penggunaan Solarium antara lain:

1. Meringankan developer untuk melakukan koneksi dengan Solr, misalnya untuk pengaturan parameter, membangun string, mengonfigurasi API yang mudah digunakan.
2. Model kueri dapat diperluas penggunaannya dan bebas untuk memodifikasinya.
3. Fleksibel, maksudnya model kueri dan hasil tidak terkait dengan penerapan klien Solr tertentu.
4. Dapat digunakan di berbagai aplikasi PHP tanpa bergantung pada suatu framework.
5. Memungkinkan developer untuk berfokus pada logika bisnis.

2.2.11 Analytical Hierarchial Process

Analytical Hierarchial Process (AHP) merupakan suatu proses yang melakukan proses justifikasi terhadap beberapa kriteria. AHP di-

kembangkan oleh Thomas Saaty pada tahun 1980 dengan pendekatan yang memecah masalah kompleks menjadi masalah yang tersusun secara hierarki.

AHP dapat digunakan untuk melakukan penilaian terhadap beberapa criterion yang dapat membantu dalam pengambilan keputusan nantinya. Pada penelitian ini, penggunaan AHP digunakan untuk menentukan nilai boost untuk setiap field yang ada dalam Solr collection, mana yang memiliki *boost* tertinggi ke *boost* terendah. Langkah-langkah yang dilakukan untuk menghitung AHP adalah sebagai berikut:

- Identifikasi kriteria yang akan digunakan dalam penelitian. Pada penelitian ini, terdapat 4 kriteria yang akan dilakukan perhitungan. Empat kriteria tersebut adalah nama field yang ada, yaitu Kode Produk, Nama Produk, Nama Produsen, dan Nama Bahan Makanan.
- Membuat matriks sebesar $n \times n$. Pada tahap ini, dibuat sebuah matriks dengan dimensi $n \times n$ dimana n merupakan kriteria yang akan dilakukan pembobotan nantinya. Contoh matriks perhitungan dapat dilihat pada tabel 2.1.

Tabel 2.1: Matriks AHP awal

Criteria	food_code	food_name	food_man	food_ing
food_code	1			
food_name		1		
food_man			1	
food_ing				1

- Melakukan proses *pairwise-comparison*. Selanjutnya, menggunakan hasil survey, dilakukan perhitungan *pairwise-comparison* dimana kolom dibandingkan dengan

baris. Baris dan kolom memiliki hubungan "Kriteria mana yang lebih penting, antara baris dengan kolom dari nilai 1 hingga 9 dimana 1 merupakan sangat tidak penting dan 9 sangat penting". Misalkan pada baris 1 dan kolom 2 memiliki nilai 5.00, maka kriteria 1 lebih penting daripada kriteria 2. Apabila dilihat dari baris 2 dan kolom satu maka nilainya akan terbalik menjadi 0.20 dimana artinya adalah kriteria 2 tidak terlalu penting dibanding kriteria 1. Contoh perhitungan *pairwise-comparison* terlihat pada tabel 2.2.

Tabel 2.2: Contoh hasil perhitungan *pairwise-comparison*

Criteria	food_code	food_name	food_man	food_ing
food_code	1	0.14	0.14	0.14
food_name	7.00	1	7.00	7.00
food_man	7.00	0.14	1	0.14
food_ing	7.00	0.14	7.00	1

- Jumlahkan setiap kolomnya.
Penjumlahan nilai dilakukan pada semua kolom tiap kriteria. Hasil penjumlahan ini akan dijadikan acuan untuk dilakukan normalisasi di tahap berikutnya. Contoh hasil penjumlahan terlihat pada tabel 2.3.

Tabel 2.3: Contoh hasil penjumlahan kriteria

	food_code	food_name	food_man	food_ing
SUM	22	1.42	15.14	8.28

- Melakukan normalisasi matriks yang telah dibuat.
Pada tahap selanjutnya, dilakukan proses normalisasi matriks sesuai tabel 2.4 dengan cara membagi nilai tiap kolom kriteria dengan jumlah tiap kolom yang telah dihitung ditahap sebelumnya.

Tabel 2.4: Contoh hasil normalisasi *pairwise-comparison*

Criteria	food_code	food_name	food_man	food_ing
food_code	0.04	0.09	0.01	0.02
food_name	0.32	0.70	0.46	0.84
food_man	0.32	0.09	0.07	0.02
food_ing	0.32	0.09	0.46	0.12

- Tentukan strategi dari hasil kriteria yang didapatkan.
Untuk mengetahui pembobotan akhir menggunakan metode AHP, semua baris dirata-rata dan akan didapatkan hasil seperti pada tabel 2.5 . Pada tabel tersebut, angka yang paling tinggi merupakan kriteria yang dianggap paling penting dan angka yang paling rendah merupakan kriteria yang dianggap tidak terlalu penting.

Tabel 2.5: AHP hasil prioritas *field* awal

Kriteria	Nama	Rata-rata/Bobot Kriteria
1	food_code	0.4
2	food_name	0.7
3	food_man	0.12
4	food_ing	0.9

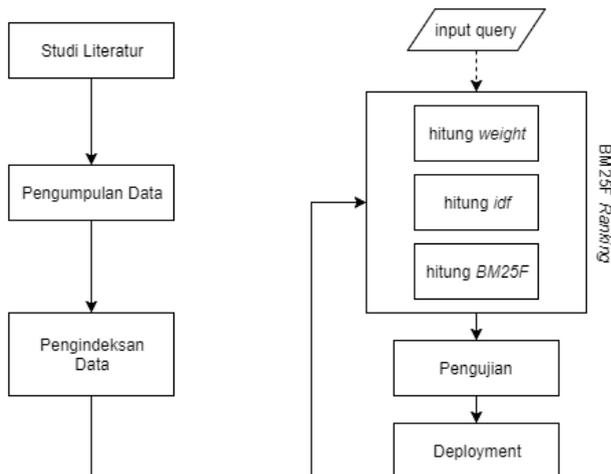
BAB 3

METODOLOGI

Bab ini menjelaskan tentang tahapan – tahapan yang dilakukan dalam pengerjaan tugas akhir beserta penjelasan tiap tahapan tersebut. Lalu disertakan jadwal pengerjaan tiap tahapnya.

3.1 Tahapan Pengerjaan Tugas Akhir

Pada sub bab ini akan dijelaskan mengenai metodologi dalam pelaksanaan tugas akhir. Metodologi tersebut dapat dilihat pada Gambar 3.1



Gambar 3.1: Metodologi Tugas Akhir

3.1.1 Studi literatur

Studi literatur dilakukan untuk memahami konsep, teori dan teknologi yang digunakan dalam penelitian. Pada tahap ini dilakukan pengumpulan literatur terkait yang mendukung dalam menyelesaikan tugas akhir. Literatur yang dipelajari adalah hasil penelitian yang membahas tentang Semantic Web khususnya Semantic Search dengan metode pencarian menggunakan algoritma BM25.

3.1.2 Pengumpulan data

Pengumpulan data adalah proses mengumpulkan data-data produk halal. Data sebelumnya sudah ada dalam aplikasi Halal Nutrition Food yang berjumlah 100 entri. Dalam penelitian ini akan ditambahkan 50 entri produk halal sehingga berjumlah 150 entri produk halal.

3.1.3 Pengindeksan data

Data produk halal yang tersimpan dalam database masih berupa data mentah yang harus diolah lagi supaya mempermudah dalam melakukan pencarian. Data produk halal yang akan dilakukan pencarian harus berada dalam Apache Solr yang sebelumnya telah dilakukan pengindeksan dari basis data. Oleh karena itu proses pengindeksan ini penting dalam aktifitas pencarian data produk halal.

3.1.4 BM25F Ranking

Proses Ranking sebelumnya dijelaskan pada subbab 2.2.2. BM25F adalah ekstensi dari fungsi ranking BM25 yang digunakan untuk

menilai dokumen yang terstruktur. Contoh implementasi metode ranking pada dokumen RDF adalah sebagai berikut:

Instance RDF

Pengujian dilakukan pada 2 buah instance yang ada pada Tabel 3.1.

Tabel 3.1: Dua instances RDF data produk halal

Subyek	Predikat	Obyek
halalf:1	halalv:FoodProduct halalv:certificate halalv:manufacture	Serena Snack 00100039300306 PT Nissin Biscuit Indonesia
halalf:2	halalv:FoodProduct halalv:certificate halalv:manufacture	Nissin Wafers Co- klat 00290047180208 PT Monde Mahko- ta Biskuit

Query

Query yang dimasukkan adalah: Nissin Serena

Query ini terdiri dari 2 kata, sehingga bisa dilakukan tokenisasi menjadi Nissin (t1) dan Serena (t2)

Perhitungan $weight(t, d)$

Dokumen disini terdapat 2 instance yaitu halalf:1 dan halalf:2. Nilai $weight$ dihitung berdasarkan token query yang ada di dalam se-

tiap dokumen. Nilai *boost* untuk setiap field adalah FoodProduct = 3, certificate = 1, dan manufacture = 2. Nilai *boost* ini dibuat berdasarkan perkiraan kemunculan term dalam fieldnya. Perhitungan menggunakan rumus 2.3.

$$weight(t1,halalf:1_{FoodProduct}) = \frac{0 \cdot 3}{((1-1)+1 \cdot \frac{3}{3})} = \frac{0}{1} = 0$$

$$weight(t1,halalf:1_{certificate}) = \frac{0 \cdot 1}{((1-1)+1 \cdot \frac{3}{3})} = \frac{0}{1} = 0$$

$$weight(t1,halalf:1_{manufacture}) = \frac{1 \cdot 2}{((1-1)+1 \cdot \frac{3}{3})} = \frac{2}{1} = 2$$

$$weight(t2,halalf:1_{FoodProduct}) = \frac{1 \cdot 3}{((1-1)+1 \cdot \frac{3}{3})} = \frac{3}{1} = 3$$

$$weight(t2,halalf:1_{certificate}) = \frac{0 \cdot 1}{((1-1)+1 \cdot \frac{3}{3})} = \frac{0}{1} = 0$$

$$weight(t2,halalf:1_{manufacture}) = \frac{0 \cdot 2}{((1-1)+1 \cdot \frac{3}{3})} = \frac{0}{1} = 0$$

$$weight(t,halalf:1) = weight(t1,halalf:1) + weight(t2,halalf:1)$$

$$weight(t,halalf:1) = \mathbf{5}$$

$$weight(t1,halalf:2_{FoodProduct}) = \frac{1 \cdot 3}{((1-1)+1 \cdot \frac{3}{3})} = \frac{3}{1} = 3$$

$$weight(t1,halalf:2_{certificate}) = \frac{0 \cdot 1}{((1-1)+1 \cdot \frac{3}{3})} = \frac{0}{1} = 0$$

$$weight(t1,halalf:2_{manufacture}) = \frac{0 \cdot 2}{((1-1)+1 \cdot \frac{3}{3})} = \frac{0}{1} = 0$$

$$weight(t2,halalf:2_{FoodProduct}) = \frac{0 \cdot 3}{((1-1)+1 \cdot \frac{3}{3})} = \frac{0}{1} = 0$$

$$weight(t2,halalf:2_{certificate}) = \frac{0 \cdot 1}{((1-1)+1 \cdot \frac{3}{3})} = \frac{0}{1} = 0$$

$$weight(t2,halalf:2_{manufacture}) = \frac{0 \cdot 2}{((1-1)+1 \cdot \frac{3}{3})} = \frac{0}{1} = 0$$

$$\begin{aligned} \text{weight}(t, \text{halalf}:2) &= \text{weight}(t1, \text{halalf}:2) + \text{weight}(t2, \text{halalf}:2) \\ \text{weight}(t, \text{halalf}:2) &= \mathbf{3} \end{aligned}$$

Perhitungan $idf(t)$

Nilai idf ini menghitung token query dalam keseluruhan dokumen. Pada perhitungan ini diasumsikan nilai N yaitu jumlah seluruh instance adalah 100 entri. Perhitungan menggunakan rumus 2.2.

$$idf(t1) = \log \frac{100-2+0.5}{2+0.5} = \log \frac{98.5}{2.5} = 1.56$$

$$idf(t2) = \log \frac{100-1+0.5}{1+0.5} = \log \frac{99.5}{1.5} = 1.82$$

$$\begin{aligned} idf(t) &= idf(t1) + idf(t2) \\ idf(t) &= 1.56 + 1.82 = \mathbf{3.38} \end{aligned}$$

Perhitungan $BM25F$

Nilai $BM25F$ adalah nilai akhir yang dimiliki setiap dokumen. Perhitungan menggunakan rumus 2.1.

$$BM25F \text{ halalf}:1 = 3.38 \cdot \frac{5}{1+5} = \mathbf{2.82}$$

$$BM25F \text{ halalf}:2 = 3.38 \cdot \frac{3}{1+3} = \mathbf{2.53}$$

Hasil Ranking

Hasil ranking diurutkan berdasarkan nilai $BM25F$ yang paling besar, oleh karena itu hasil ranking dari 2 instance diatas yang ditampilkan pada hasil pencarian adalah :

1. halalf:1
2. halalf:2

3.1.5 Pengujian

Pada tahap ini dilakukan uji coba secara keseluruhan untuk mengetahui sistem kerja dalam pencarian data produk halalnya apakah terdapat kesalahan atau error pada perangkat lunak. Aplikasi diuji menggunakan metode sebagai berikut:

- Pengujian Fungsional, merupakan pengujian terhadap fungsionalitas dari perangkat lunak yang telah dibangun untuk memastikan fungsionalitas dari perangkat lunak berjalan semestinya. Fungsionalitas perangkat lunak diuji melalui percobaan menggunakan skenario yang telah disiapkan. Setiap skenario penggunaan perangkat lunak akan menguji setiap fungsionalitas berbeda dari perangkat lunak. Sisi fungsional yang akan diuji salah satunya mencoba beberapa query pencarian produk.

Contoh beberapa skenario yang akan dilakukan adalah pada Tabel 3.2.

Tabel 3.2: Skenario Pengujian Fungsional

No.	Faktor	Detail	Ukuran
1.	Ketepatan	Dimasukkan query dengan 2 suku kata	Persentase
2.	Ketepatan	Dimasukkan query dengan n suku kata yang memfokuskan pada nama produk	Persentase
3.	Ketepatan	Dimasukkan query dengan n suku kata yang memfokuskan pada nomor sertifikat	Persentase
4.	Ketepatan	Dimasukkan query dengan n suku kata yang memfokuskan pada nama produsen	Persentase

- Pengujian Non-fungsional, merupakan pengujian terhadap sisi non-fungsional dari perangkat lunak. Sisi non fungsional yang akan diuji adalah uji coba adalah pada Tabel 3.3.

Tabel 3.3: Skenario Pengujian Non-Fungsional

No.	Faktor	Detail	Ukuran
1.	Waktu	Hasil pencarian dengan 2 hingga n suku kata	Detik
2.	Waktu	Hasil pencarian dengan jumlah data yang berbeda	Detik
3.	Kapasitas	Kemampuan server untuk melakukan pencarian ketika banyak user yang mengakses	Bandwith

3.1.6 *Deployment*

Setelah pengujian secara fungsional maupun non fungsional selesai dilakukan dan tidak terjadi *error*, perangkat lunak akan diluncurkan ke masyarakat luas.

3.1.7 **Pembuatan buku Tugas Akhir**

Pembuatan buku tugas akhir akan dilakukan secara bersamaan dengan pembuatan sistem. Buku ini berisi tentang semua aktivitas yang ada dan proses yang berhubungan dalam pengerjaan tugas akhir.

3.2 Penjelasan Sistem

Sistem pencarian yang dibuat dalam aplikasi Halal Nutrition Food ini adalah sistem pencarian *entity* yang tersimpan dalam Apache Solr. Kueri yang dimasukkan oleh pengguna selanjutnya akan dilakukan pencarian dengan menghitung kemunculan term dalam *entity* pada data menggunakan algoritma BM25F. Algoritma BM25F sendiri dipilih sebagai algoritma pencarian dalam aplikasi karena algoritma ini berfokus pada pencarian *entity* berbasis field seperti data yang ada pada aplikasi.

Sesuai dengan perumusan masalah yang dijelaskan, rumusan masalah yang pertama adalah tentang bagaimana cara pengguna bisa memperoleh informasi suatu produk secara relevan dapat terjawab dengan metode pencarian aplikasi yang menggunakan algoritma BM25F dan pencarian dengan hanya menggunakan satu halaman. Pencarian menggunakan algoritma BM25F memungkinkan pengguna memperoleh hasil pencarian sesuai kueri dengan memperhitungkan kemunculan term kueri dalam *entity*. Sementara pencarian yang hanya menggunakan satu halaman memungkinkan pengguna untuk memasukkan kueri pencarian berdasarkan beberapa field yang ada seperti nama produk, kode produk, nama produsen, dan nama bahan makanan secara bersamaan dalam sebuah kueri.

Rumusan masalah yang kedua adalah mengenai penerapan algoritma BM25F dalam sistem pencarian dapat terjawab dengan penggunaan plugin BM25F yang dipasangkan di dalam Apache Solr. Dalam implementasinya, plugin ini membutuhkan beberapa konfigurasi seperti yang dijelaskan pada Bab 5. Kemudian dalam perhitungan BM25F memerlukan variabel nilai *boost* untuk setiap field yang ada. Oleh karena itulah dilakukan survei kepada masyarakat umum untuk menghitung AHP dari 4 kategori field yang ada pada data guna melihat prioritas field yang dicari agar menyesuaikan de-

ngan kebutuhan pengguna. Dengan cara itu, field dengan prioritas tertinggi akan memiliki nilai yang lebih besar dan lebih diperhitungkan dalam proses pencarian produk dalam sistem pencarian.

Halaman ini sengaja dikosongkan

BAB 4

PERANCANGAN

Bab ini membahas terkait alur perancangan dalam beberapa hal yang diperlukan dalam proses pembuatan aplikasi sesuai dengan metode yang dijelaskan pada bab 3. Di dalamnya akan dijelaskan tentang proses penggalan kebutuhan dan desain sistem.

4.1 Penyusunan Kuisisioner

Penyusunan kuisisioner ini digunakan untuk menghitung AHP (*Analytical Hierarchical Process*) guna mencari nilai *boost* yang tepat untuk masing-masing *field*. Kuisisioner diberikan kepada masyarakat umum untuk mengetahui kecenderungan masyarakat dalam melakukan pencarian berdasarkan 4 kategori *field*, yaitu:

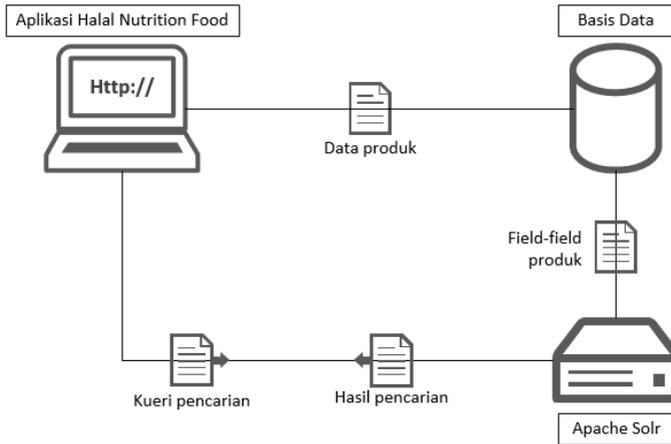
- Nama Produk
- Kode Produk
- Nama Produsen
- Nama Bahan Makanan

Kuisisioner dibuat menggunakan *Google Form* (<http://intip.in/halalin>) lalu disebarakan secara online. Konten kuisisioner dapat dilihat pada Lampiran A.

4.2 Desain Sistem

Sistem pencarian produk halal yang dibuat dalam aplikasi Halal Nutrition Food ini melalui dua layer server yang saling terkoneksi.

si, penjelasannya digambarkan pada Gambar 4.1



Gambar 4.1: Desain sistem dalam aplikasi

Penjelasan desain sistem:

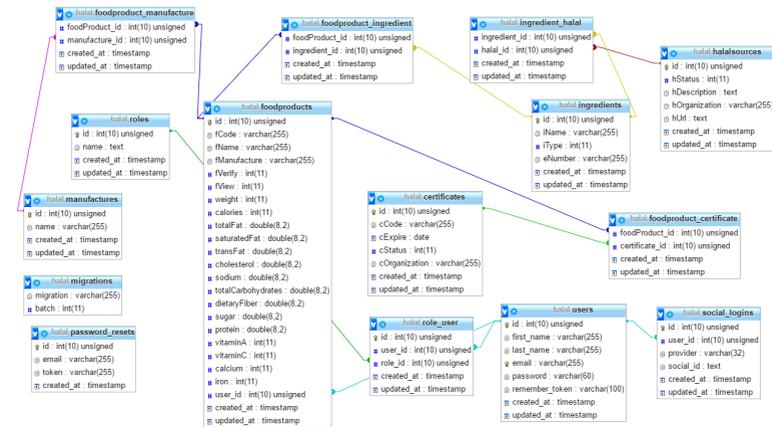
1. Apache Web Server, di dalamnya terdapat dua sistem yang berjalan yakni basis data dan aplikasi web Halal Nutrition Food.
 - Sistem basis data yang digunakan adalah MySQL, skema basis datanya akan dijelaskan pada bagian desain basis data. Sistem kerjanya adalah ketika pengguna memasukkan data produk halal maka aplikasi web akan menyimpan data-data produk tersebut ke dalam basis data MySQL.
 - Aplikasi web Halal Nutrition Food yang berjalan pada server ini dibangun menggunakan framework PHP Laravel yang di dalamnya terdapat fungsi untuk menghubungkan website dengan server Apache Solr, pro-

ses penghubungan keduanya digunakan ketika pengguna mengirimkan kueri pencarian produk melalui halaman aplikasi web kemudian aplikasi web akan mengirimkan kueri tersebut untuk diproses menggunakan algoritma BM25F di dalam server Apache Solr, selanjutnya server Apache Solr akan mengembalikan daftar hasil pencarian ke aplikasi web.

2. Apache Solr, digunakan untuk memproses pencarian yang dilakukan pengguna dan menyimpan indeks dari dokumen. Dalam Apache Solr akan dilakukan dua proses:
 - Proses *indexing* field-field dari tabel dalam basis data agar pengguna bisa melakukan pencarian berdasarkan data dalam field tersebut.
 - Proses *scoring* menggunakan algoritma BM25F sehingga hasil pencarian akan diurutkan berdasarkan penilaian algoritma tersebut.

4.3 Desain Basis Data

Skema basis data yang digunakan untuk membangun aplikasi web ini digambarkan pada Gambar 4.2



Gambar 4.2: Skema basis data yang digunakan dalam aplikasi

Penjelasan tabel dalam skema basis data:

- foodproducts berisi data-data utama dari produk makanan dan minuman seperti kode produk, nama produk, produsen, kandungan gizi, dan nama pengguna yang memasukkan produk tersebut. Tabel ini menjadi referensi untuk beberapa tabel berikutnya.
- foodproducts.ingredient adalah tabel relasi antara tabel foodproducts dan tabel ingredient. Di dalamnya berisi kolom id produk dan id ingredient yang ada di dalam setiap produk.
- foodproducts.manufacture adalah tabel relasi antara foodproducts dan tabel manufacture. Di dalamnya berisi kolom id produk dan id manufacture.
- foodproducts.certificate adalah tabel relasi antara tabel foodproducts dan tabel certificate. Di dalamnya berisi kolom id produk dan id certificate yang dimiliki oleh setiap produk.
- ingredients berisi data mengenai bahan-bahan atau zat aditif yang terkandung dalam setiap produk. Di dalamnya berisi

kolom nama zat, tipe zat, eNumber zat, dsb.

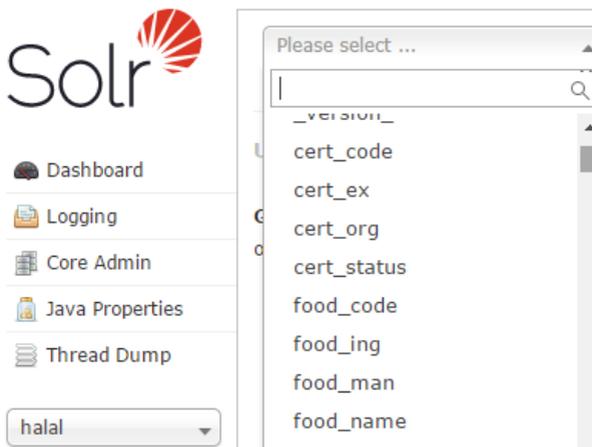
- `ingredients_halal` adalah tabel relasi antara tabel `ingredients` dengan tabel `halalsources`. Di dalamnya berisi kolom `id ingredients` dan `id dari sumber halal`. Fungsinya untuk melihat apakah bahan tersebut termasuk bahan yang halal atau tidak.
- `halalsources` berisi data mengenai penjelasan sumber halal.
- `manufactures` berisi data mengenai produsen makanan dan minuman tersebut.
- `certificates` berisi data mengenai sertifikat halal yang dimiliki oleh sebuah produk dari berbagai sumber.
- `users` berisi data pengguna yang telah terdaftar dalam aplikasi yang merupakan tabel referensi untuk menentukan role dari setiap pengguna.
- `roles` berisi data mengenai role pengguna yang ada di dalam aplikasi.
- `role_user` adalah tabel relasi antara `users` dan tabel `roles`. Di dalamnya berisi kolom `id user` dan `id role`. Fungsinya untuk menjelaskan seorang pengguna mempunyai role sebagai apa.
- `social_login` adalah tabel yang berisi data mengenai pengguna yang login ke aplikasi menggunakan sosial media yang dimilikinya.
- `migrations` adalah tabel bawaan dari Laravel yang berfungsi untuk menyimpan *migration* yang dimiliki oleh aplikasi.

4.4 Desain Skema *Solr collection*

Solr collection adalah sebuah kumpulan indeks dalam dokumen yang berada dalam server Apache Solr. Proses indeks berlangsung karena dalam Apache Solr terdapat fungsi yang bernama *Data Import Handler* yaitu sebuah fungsi yang berguna untuk melakukan

koneksi dari Apache Solr ke DBMS. Setelah berhasil terkoneksi kemudian skema dalam *solr collection* ini diperlukan untuk menyimpan indeks dari field-field yang akan dilakukan pencarian oleh pengguna.

Skema dalam *solr collection* dalam aplikasi ini digambarkan pada Gambar 4.3



Gambar 4.3: Desain skema *solr collection*

Penjelasan skema dalam *solr collection*:

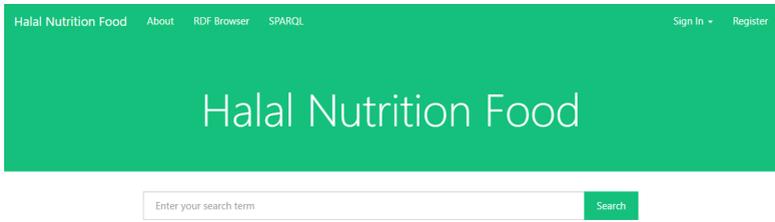
1. id berfungsi sebagai field unik utama yang ada dalam skema.
2. food_name berfungsi untuk mengindeks nama-nama dari produk makanan dan minuman yang ada di basis data.
3. food_man berfungsi untuk mengindeks nama produsen dari makanan dan minuman yang ada di basis data.
4. food_ing berfungsi untuk mengindeks nama-nama bahan yang terkandung dalam produk makanan dan minuman yang ada di

- basis data.
5. `food_code` berfungsi untuk mengindeks kode produk dalam produk makanan dan minuman yang ada di basis data.
 6. `cert_code` berfungsi untuk mengindeks kode dari sertifikat yang dimiliki produk makanan dan minuman yang ada di basis data.
 7. `cert_org` berfungsi untuk mengindeks nama organisasi dari pemberi sertifikat produk makanan dan minuman yang ada di basis data.
 8. `cert_ex` berfungsi untuk mengindeks masa aktif dari sertifikat yang dimiliki produk makanan dan minuman yang ada di basis data.
 9. `cert_status` berfungsi untuk mengindeks status dari sertifikat yang dimiliki produk makanan dan minuman yang ada di basis data.

4.5 Desain Antarmuka Pengguna

Desain antarmuka aplikasi terdapat beberapa bagian, namun dalam tugas akhir ini memfokuskan pada proses pencarian yang dilakukan oleh pengguna. Oleh karena itu berikut dilampirkan tiga desain halaman aplikasi yaitu halaman utama, halaman hasil pencarian dan halaman produk.

Halaman utama dalam aplikasi Halal Nutrition Food yang telah dilakukan perubahan dari versi sebelumnya bisa dilihat pada Gambar 4.4



Gambar 4.4: Halaman utama antarmuka pengguna aplikasi

Pada halaman utama ini terdapat *Search field* yang ditampilkan sebagai fungsi utama pada aplikasi. Kemudian di halaman ini terdapat beberapa menu, yaitu menu *Sign In* untuk autentikasi pengguna yang ingin masuk ke dalam aplikasi, *Register* untuk pengguna yang ingin mendaftar ke dalam aplikasi, *About* berisi penjelasan mengenai aplikasi Halal Nutrition Food, *RDF Browser* berisi fungsi untuk melakukan pencarian terhadap file RDF, *SPARQL* berisi SPARQL endpoint untuk melakukan kueri terhadap SPARQL dalam aplikasi. *RDF Browser* dan *SPARQL* bukan merupakan fokus utama dalam tugas akhir ini karena merupakan fokus dari penelitian yang dilakukan sebelumnya.

Halaman hasil pencarian yang digambarkan pada Gambar 4.5, sebenarnya adalah halaman yang sama dengan halaman utama, hanya saja di bagian bawah dari *search field* terdapat analisis hasil pencarian yaitu jumlah dokumen yang ditemukan dan waktu untuk melakukan pencarian, kemudian di bawahnya ada daftar produk hasil pencarian dari kueri yang dimasukkan oleh pengguna. Daftar produk yang ditampilkan adalah nama produk, kode produk, nama produsen, bahan-bahan yang terkandung dalam produk, dan nilai relevansi produk terhadap kueri yang dimasukkan oleh pengguna.

Halal Nutrition Food About RDF Browser SPARQL Admin Submit Browse

Halal Nutrition Food

day time Search

Your search took **0.189 seconds** and yielded **7 result(s)** raw output - solr json

Good Time Mini (894755030400) relevance score : **4.251935**
 PT. Amott's Indonesia
 Wheat Flour, Choco Chips, Sugar, Margarine, Cocoa Powder, Sodium carbonate, Eggs, Salt, Emulsifier, Sesame Seeds
 Peanuts, Treenuts

Good Day Original (8991002103535) relevance score : **2.7940774**
 PT. Santos Jaya Abadi
 Sugar, Non Dairy Creamer, Instant Coffee, Artificial Sweetener Sucralose, Acesulfame-K

Gambar 4.5: Halaman hasil pencarian dalam aplikasi

Halal Nutrition Food About RDF Browser SPARQL Admin Submit Browse

Food: Good Time Mini

Nutrition Facts	
Good Time Mini	
Amount Per Serving	
Calories 90	
	% Daily Value*
Total Fat 4g	6%
Saturated Fat 2.5g	13%
Trans Fat 0g	
Cholesterol 0mg	0%
Sodium 90mg	4%
Total Carbohydrates 0g	0%
Dietary Fiber 2g	8%
Sugars 5g	
Protein 1g	
Vitamin A	0%
Vitamin C	0%
Calcium	0%
Iron	0%

* Percent Daily Values are based on a 2000 calorie diet.

INGREDIENTS: Wheat Flour, Choco Chips, Sugar, Margarine, Cocoa Powder, Eggs, Salt, Emulsifier, Sesame Seeds Peanuts, Treenuts.

Food ID
894755030400

Food Name
Good Time Mini

Food Manufacture
PT. Amott's Indonesia

Food Ingredient
Wheat Flour, Choco Chips, Sugar, Margarine, Cocoa Powder, Eggs, Salt, Emulsifier, Sesame Seeds Peanuts, Treenuts.

Food Additive

- Sodium carbonate

Food Info

- This food is less filling

Food Certificate		
No Certificate	Expire Date	Certificate Status
00100027970104	03-09-2018	Development

Gambar 4.6: Halaman produk dalam aplikasi

Halaman produk yang digambarkan pada Gambar 4.6 ini tampil ketika pengguna memilih produk dari daftar hasil pencarian dari halaman sebelumnya. Di dalam halaman ini berisi informasi lengkap mengenai sebuah produk makanan dan minuman, seperti nama produk, kode produk, produsen, bahan zat aditif, sertifikat halal, dan kandungan gizi yang terkandung dalam produk tersebut.

BAB 5

IMPLEMENTASI

Pada bab ini akan dijelaskan terkait proses implementasi aplikasi Halal Nutrition Food yang dirancang. Di dalamnya akan dijelaskan mulai dari lingkungan implementasi yang digunakan dan proses implementasi dari aplikasi.

5.1 Lingkungan Implementasi

Pada bagian ini dibahas terkait lingkungan pengujian yang digunakan dalam implementasi tugas akhir terkait perangkat yang digunakan baik perangkat keras maupun perangkat lunak. Tabel 5.1 berisikan spesifikasi perangkat keras dan perangkat lunak untuk implementasi pada tugas akhir ini.

Tabel 5.1: Spesifikasi perangkat keras

Perangkat	Spesifikasi
Jenis	HP 1000 Notebook PC
OS	Windows 10 Education 64-bit
Processor	Intel Core i3
RAM	8GB
Hard Disk Drive	500GB

Kemudian untuk perangkat lunak yang digunakan dalam implementasi model ditunjukkan dalam tabel 5.2.

Tabel 5.2: Spesifikasi perangkat lunak

Nama Perangkat Lunak	Kegunaan dalam Implementasi
Xampp 5.6.15 dengan PHP 5.6.15	Webserver
MySQL 5.0.11	Database
Apache Solr 4.0	Solr Server
apache-solr-dataimporthandler-4.0.0.jar	Plugin Data Import Handler
mysql-connector-java-5.1.41-bin.jar	Konektor MySQL ODBC
bm25-ranking-0.0.1-SNAPSHOT-jar-with-dependencies.jar	Plugin BM25F

5.2 Implementasi

Pada proses implementasi akan dilakukan realisasi dari perancangan yang sudah dibuat sebelumnya. Berikut adalah proses implementasi yang dilakukan:

1. Konfigurasi server
2. Pengindeksan data MySQL
3. Perhitungan AHP
4. Pemrosesan Algoritma BM25F
5. Koneksi Apache Solr Dengan Aplikasi

5.2.1 Konfigurasi Server

Konfigurasi server yang dimaksud adalah server Apache untuk halaman web aplikasi, basis data MySQL, serta server Solr. Dengan penjelasan sebagai berikut:

- Apache berjalan pada port 80 yang berfungsi untuk mempro-

ses halaman web aplikasi Halal Nutrition Food.

- MySQL berjalan pada port 3306 yang berfungsi untuk menyimpan data-data mengenai produk halal yang telah dimasukkan pengguna melalui halaman web.
- Apache Solr berjalan pada port 8983 yang berfungsi untuk melakukan perhitungan algoritma BM25F yang datanya diambil dari basis data MySQL. Untuk selanjutnya hasil perhitungan akan ditampilkan kembali ke halaman web aplikasi.

5.2.2 Pengindeksan Data MySQL

Data produk halal tersimpan dalam basis data MySQL yang sebelumnya telah dimasukkan oleh pengguna melalui halaman web aplikasi. Kemudian data tersebut akan dilakukan pengindeksan oleh Apache Solr menggunakan fungsi *Data Import Handler*.

Untuk melakukan koneksi dan pengindeksan data dari MySQL ke Apache Solr dibutuhkan konfigurasi sebagai berikut:

1. Membuat collection data halal

Pada tahap ini dilakukan pembuatan collection baru dalam server Solr, kemudian dalam *schema.xml* dimasukkan *field* data yang nantinya diperlukan untuk pengindeksan. *field* data yang dimaksud dapat dilihat pada kode 5.1

Kode 5.1: *field* pada *schema.xml*

```
<field name="id" type="string" multiValued="false" indexed="true"
  required="true" stored="true"/>
<field name="cert_code" type="text_general" indexed="true" stored="true"/
>
<field name="cert_ex" type="text_general" indexed="true" stored="true"/>
<field name="cert_org" type="text_general" indexed="true" stored="true"/>
<field name="cert_status" type="text_general" indexed="true" stored="true"
"/>
<field name="food_code" type="text_general" indexed="true" required="true"
stored="true"/>
<field name="food_ing" type="text_general" multiValued="true" indexed="
true" stored="true"/>
```

```
<field name="food_man" type="text_general" indexed="true" required="true"
  stored="true"/>
<field name="food_name" type="text_general" indexed="true" required="true"
  stored="true"/>
```

Dalam skema tersebut dibuat 9 *field* yang nantinya akan dijadikan wadah untuk menampung data hasil pengindeksan dari basis data MySQL. Lalu pada *field-field* ini lah perhitungan algoritma BM25F akan dilakukan.

2. Koneksi Apache Solr dengan MySQL

Sebelum melakukan pengindeksan diperlukan pengaturan koneksi antara Apache Solr dengan basis data MySQL. Untuk itu pada file `solrconfig.xml` disisipkan potongan kode yang tertera pada kode 5.2

Kode 5.2: Data Import Handler pada collection data halal

```
<!-- Konfigurasi Data Import Handler -->
<lib dir="lib/" regex="mysql-connector-java -.*\.jar" />
<lib dir="lib/" regex="solr-dataimporthandler -.*\.jar" />
<requestHandler name="/dataimport" class="solr.DataImportHandler">
  <lst name="defaults">
    <str name="config">data-config.xml</str>
  </lst>
</requestHandler>
```

3. File data-config.xml

File `data-config.xml` dibuat di dalam direktori `conf` pada collection data halal. Fungsinya adalah menjelaskan nama basis data yang dituju beserta autentikasinya, dan kolom apa saja yang akan dilakukan pengindeksan oleh Apache Solr. Potongan kodenya dapat dilihat pada kode 5.3

Kode 5.3: File data-config.xml

```
<dataConfig>
  <dataSource type="JdbcDataSource" driver="com.mysql.jdbc.Driver" url=
    "jdbc:mysql://localhost/halal" user="root" password="" />
  <document>
    <entity name="foodproducts" query="SELECT id, fCode, fName,
      fManufacture FROM foodproducts">
      <field column="id" name="id" />
      <field column="fCode" name="food_code" />
      <field column="fName" name="food_name" />
      <field column="fManufacture" name="food_man" />
    </entity>
  </document>
</dataConfig>
```

```

<entity name="ingredients" query="SELECT DISTINCT i.iName
FROM ingredients i JOIN foodproduct.ingredient fi ON i.
id = fi.ingredient_id WHERE i.id IN (SELECT
foodproduct.ingredient.ingredient_id FROM
foodproduct.ingredient INNER JOIN foodproducts ON
foodproducts.id = foodproduct.ingredient.foodProduct.id
WHERE foodproducts.id = '{ foodproducts.id }')">
  <field column="iName" name="food.ing"/>
</entity>
</document>
</dataConfig>

```

Pada potongan kode dijelaskan bahwa koneksi Apache Solr dilakukan ke basis data halal dengan autentikasi yang tertera di dalam kode tersebut.

Isi dari kolom dalam basis data yang didefinisikan sebagai sebuah *entity* kemudian dimasukkan ke dalam setiap *field* di skema Solr yang telah dibuat sebelumnya. Kemudian untuk mengambil *field ingredients* dilakukan kueri join sehingga daftar bahan-bahan yang menyusun suatu produk dapat terdaftar dengan jelas.

5.2.3 Perhitungan AHP

Setiap *field* dalam Solr *collection* perlu diberikan nilai *boost* agar hasil pencarian relevan dengan pencarian. Perhitungan AHP ini akan menjawab nilai *boost* ideal sesuai dengan permintaan pengguna yang sebelumnya dilakukan telah survei kepada responden untuk mengetahui kecenderungan pengguna dalam melakukan pencarian produk halal.

Dari hasil survei yang disebar kepada responden, sejumlah 136 responden sudah mengisi kuisioner. Selanjutnya dilakukan perhitungan *Consistency Ratio* (CR) pada pembobotan yang didapat untuk keempat kategori *field*. Perhitungan menggunakan calcu-

lator AHP (http://bpmmsg.com/academic/ahp_calc.php) untuk membantu proses penghitungan konsistensi AHP. Dari hasilnya didapatkan bahwa nilai CR terbilang tinggi, yaitu sebesar 37.5% dengan tabel prioritas sesuai pada tabel 5.3.

Tabel 5.3: AHP hasil prioritas *field* awal

No	Category	Priority	Rank
1	Kode produk	3.4%	4
2	Nama produk	63.5%	1
3	Nama produsen	9.1%	3
4	Nama bahan makanan	24%	2

Namun dikarenakan berdasarkan teori konsistensi AHP, apabila nilai rasio konsistensi dibawah 10 persen, maka dapat dikatakan hasil AHP tersebut konsisten, namun apabila diatas 10 persen, perlu dilakukan penyesuaian terhadap penilaian AHP. Selanjutnya penilaian AHP tersebut dilakukan penyesuaian sehingga menghasilkan nilai CR sebesar 4.6% dan menghasilkan tabel prioritas yang terlihat pada tabel 5.4.

Tabel 5.4: AHP hasil prioritas *field* setelah normalisasi

No	Category	Priority	Rank
1	Kode produk	4.7%	4
2	Nama produk	63.5%	1
3	Nama produsen	9.8%	3
4	Nama bahan makanan	22%	2

Hasil tabel 5.4 yang akan dijadikan acuan untuk mengisi nilai *boost* pada plugin BM25F nantinya.

5.2.4 Pemrosesan Algoritma BM25F

Perhitungan algoritma BM25F dilakukan dengan membuat Plugin Query Parser pada Solr. Sehingga dibutuhkan 2 aktifitas yaitu membuat Plugin BM25F dan memasukkan Plugin tersebut sebagai metode utama dalam melakukan pencarian data dalam indeks collection data halal. Kedua aktifitas tersebut adalah sebagai berikut.

1. Pembuatan Plugin BM25F

Pembuatan Plugin untuk memroses BM25F menggunakan bahasa kode plugin yang dijelaskan pada Bab 2 sebelumnya. Dari kode tersebut kemudian dibuat file jar nya untuk kemudian dimasukkan ke dalam collection data halal dan perhitungan persamaan pencarian akan menggunakan perhitungan dari plugin tersebut.

2. Konfigurasi Plugin pada Collection Data Halal

Konfigurasi dilakukan pada file `solrconfig.xml` yang di dalamnya ditambahkan *Query Parser*, *Request Handler*, dan *Similarity*. Fungsi dari *Query Parser* adalah melakukan parsing kueri tekstual dan mengubahnya menjadi objek kueri Lucene. Kodenya dapat dilihat pada potongan kode 5.4

Kode 5.4: Query Parser untuk BM25F

```
<queryParser name="bm25f" class="eu.europeana ranking .bm25f.
  BM25FParserPlugin">
  <str name="mainfield">food_name</ str>
  <float name="k1">10.0</ float>
  <lst name=" fieldsBoost">
    <float name=" food_code">4.7</ float>
    <float name=" food_name">63.5</ float>
    <float name=" food_man">9.8</ float>
    <float name=" food_ing">22</ float>
  </lst>
  <lst name=" fieldsB">
    <float name=" food_code">0.06</ float>
    <float name=" food_name">0.22</ float>
    <float name=" food_man">0.23</ float>
    <float name=" food_ing">0.48</ float>
  </lst>
</queryParser>
```

Pada potongan kode tersebut dijelaskan mengenai variabel nilai untuk setiap *field* yang ada dalam *collection* data halal. Nilai yang didefinisikan antara lain nilai *k1*, nilai *boost* tiap *field*, dan nilai bobot *length* tiap *field*, dan Penjelasan untuk tiap variabel tersebut adalah sebagai berikut:

- *mainField* ditentukan dengan prioritas *field* dengan *rank* tertinggi yang didapatkan dari Tabel 5.4, yaitu Nama produk atau di skema disebut sebagai *food_name*.
- *k1* adalah nilai saturasi yang bernilai 10.
- *fieldsBoost* diperoleh dari Tabel 5.4 hasil perhitungan AHP. Nilai tiap *priority* inilah yang dijadikan nilai *boost* untuk setiap *field* yang ada.
- *fieldsB* diperoleh dari penghitungan rata-rata *length term* untuk setiap *field* dalam basis data. Caranya adalah menggunakan kueri di dalam database MySQL. Pada potongan kode 5.5 dijelaskan kueri yang digunakan untuk mencari *length term* dari setiap *field*.

Kode 5.5: Kueri SQL untuk mencari *length field*

```
# Length food_code
SELECT AVG(LENGTH(fCode) - LENGTH(REPLACE(fCode, ' ', '')) + 1) AS
length_fCode FROM foodproducts
# Hasil : length_fCode = 1

# Length food_name
SELECT AVG(LENGTH(fName) - LENGTH(REPLACE(fName, ' ', '')) + 1) AS
length_fName FROM foodproducts
# Hasil : length_fName = 4.0093

# Length food_man
SELECT AVG(LENGTH(fManufacture) - LENGTH(REPLACE(fManufacture, ' ', ''))
+ 1) AS length_fManufacture FROM foodproducts
# Hasil : length_fManufacture = 4.1121

# Length food_ing
SELECT AVG(length) AS length_fIngredients FROM (SELECT COUNT(
ingredient_id) AS length FROM foodproduct.ingredient GROUP BY
foodproduct.id) nested
# Hasil : length_fIngredients = 8.49
```

Setelah diketahui rata-rata *length* untuk setiap *field*, akan dihitung nilai *fieldsB* tiap *field* dengan menghitung proporsi nilai dari keempat *field* sehingga berjumlah 100%. Cara yang

dilakukan adalah sebagai berikut:

$$\begin{aligned} \text{Total length} &= \text{length_fCode} + \text{length_fName} + \\ &\text{length_fManufacture} + \text{length_fIngredients} \\ \text{Total length} &= 1 + 4.0093 + 4.1121 + 8.49 = 17.6114 \end{aligned}$$

$$\text{Nilai fieldsB food_code} = \frac{1}{17.6114} = 0.06$$

$$\text{Nilai fieldsB food_name} = \frac{4.0093}{17.6114} = 0.22$$

$$\text{Nilai fieldsB food_man} = \frac{4.1121}{17.6114} = 0.23$$

$$\text{Nilai fieldsB food_ing} = \frac{8.49}{17.6114} = 0.48$$

Sementara itu *Request Handler* adalah plugin dari Solr yang memroses logika dari setiap permintaan yang dilakukan pengguna. Kodenya dapat dilihat di potongan kode 5.6

Kode 5.6: Request Handler untuk BM25F

```
<requestHandler name="/bm25f" class="solr.SearchHandler">
  <lst name="defaults">
    <str name="echoParams">explicit</str>
    <str name="defType">bm25f</str>
    <str name="fl">id, food_code, food_name, food_man, food_ing, score</str>
    <str name="wt">json</str>
    <str name="indent">true</str>
  </lst>
</requestHandler>
```

Pada potongan kode tersebut dijelaskan bahwa nama dari *request handler*nya adalah *bm25f* dalam format json dan yang diambil adalah data dari beberapa *field* yang tercantum, termasuk di dalamnya ada *score* hasil pencarian untuk setiap *instance* produk halal.

Sementara untuk konfigurasi *Similarity* yang dilakukan adalah mengatur persamaan default untuk collection data halal agar menggunakan persamaan dari Plugin yang telah dibuat.

Kodenya dapat dilihat pada potongan kode 5.7

Kode 5.7: BM25F Similarity

```
<!-- BM25FSimilarity overriding the default similarity -->
<similarity class="eu.europeana.ranking.bm25f.similarity.
    BM25FSimilarityFactory" />
```

Pada potongan kode tersebut dijelaskan bahwa persamaan yang digunakan adalah persamaan yang berada dalam package Plugin yang telah dibuat. Kode tersebut dibutuhkan untuk melakukan *override* terhadap persamaan yang secara standar dimiliki oleh Apache Solr.

5.2.5 Koneksi Apache Solr Dengan Aplikasi

Data yang telah terindeks di dalam Apache Solr akan dilakukan pencarian oleh pengguna dan selanjutnya hasil pencarian akan ditampilkan kembali ke halaman web, oleh karena itu dibutuhkan koneksi antara Apache Solr dengan aplikasi web.

Aplikasi web yang dibuat menggunakan Laravel, menggunakan package Solarium untuk mengoptimalkan koneksi dengan Apache Solr.

Potongan kode Solarium dalam aplikasi Halal Nutrision Food dapat terlihat pada kode 5.8 dan 5.9.

Kode 5.8: Koneksi Solarium dengan aplikasi

```
return [
    'endpoint' => [
        'localhost' => [
            'host' => env('SOLR_HOST', '127.0.0.1'),
            'port' => env('SOLR_PORT', '8983'),
            'path' => env('SOLR_PATH', '/solr/'),
            'core' => env('SOLR_CORE', 'halal')
        ]
    ]
];
```

Kode 5.9: Konfigurasi Solarium dalam aplikasi

```

/**
 * @var The SOLR client.
 */
protected $client;

public function __construct(\Solarium\Client $client)
{
    $this->client = $client;
}

public function index()
{
    if (Input::has('q')) {
        // Execute main search
        $select = array(
            'query'      => Input::get('q'),
            'handler'    => 'bm25f',
            'start'      => 0,
            'rows'       => 10,
        );
        $query = $this->client->createSelect($select);

        // Query based on input user
        $query->setQuery(Input::get('q'));

        // add debug settings
        $debug = $query->getDebug();
        $debug->setExplainOther('id:MA*');

        // Execute the query and return the result
        $resultset = $this->client->select($query);

        // Debug result
        $debugResult = $resultset->getDebug();

        // Get query time in seconds
        $ms = $debugResult->getTiming()->getTime();
        $s = $ms/1000;

        // Pass the resultset to the view and return.
        return view('pages.v2.home', array(
            'resultset' => $resultset,
            'debugResult' => $debugResult,
        ));
    }
    return view('pages.v2.home');
}

```

Halaman ini sengaja dikosongkan

BAB 6

HASIL DAN PEMBAHASAN

Pada bab ini akan dijelaskan hasil dan pembahasan dari proses pengujian aplikasi.

6.1 Hasil Pengujian

Pada bagian ini akan dijelaskan hasil pengujian aplikasi, baik pengujian algoritma BM25F maupun pengujian kinerja.

6.1.1 Pengujian Algoritma BM25F

Pengujian algoritma dilakukan menggunakan berbagai skenario penggunaan aplikasi dimana setiap skenario berfokus dalam menguji algoritma BM25F apakah menghasilkan hasil pencarian yang relevan dengan kueri yang dimasukkan pengguna serta hasil yang lebih baik dari fungsi pencarian yang dimiliki oleh aplikasi Halal Nutrition Food versi sebelumnya. Untuk lebih jelasnya, berikut adalah skenario pengujian yang dilakukan:

1. Pengujian algoritma

Pada skenario pengujian ini dilakukan pencarian dalam aplikasi Halal Nutrition Food dengan nilai *boost* dan *length* seperti yang didefinisikan sebelumnya. Pencarian dilakukan menggunakan kueri "kacang hijau".

Hasil pencarian dapat dilihat seperti pada Gambar 6.1.

Halal Nutrition Food

Search

Your search took **0.087 seconds** and yielded **3** result(s) raw output - solr json

Energen rasa kacang hijau (8996001440087)
 PT. Torabika Eka Semesta relevance score : **5.706025**
Sugar, creamer, Wheat Flour, milk powder, Oat Corn, Calcium carbonate, Salt, Vanili, Eggs, Malt Extract

ABC Minuman Sari Kacang Hijau (711844162419)
 PT. Jeinz ABC Indonesia relevance score : **5.653508**
Mungbean Juice, Water, Sugar, Palm Sugar, Pandan Leaves, Ginger, Natural Colourant Caramel Class I-E150a

Gambar 6.1: Hasil pencarian uji coba algoritma BM25F

Penjelasan hasil pencarian teratas dari gambar tersebut adalah sebagai berikut:

Tabel 6.1: Penjelasan nilai untuk *entity* hasil pencarian teratas

Field	Kemunculan	Nilai boost	Nilai length
food_code	0	4.7	0.06
food_name	2	63.5	0.22
food_man	0	9.8	0.23
food_ing	0	22.0	0.48
Nilai BM25F	5.706025		

Nilai kemunculan pada *field* food_code dalam tabel diatas adalah 2 karena terdapat kata "kacang" dan "hijau" pada nama produk dan nilai pada *field* lain bernilai 0 karena tidak

muncul term dalam *entity* tersebut. Nilai *boost* ditentukan berdasarkan nilai AHP yang telah dihitung pada Bab 5 bagian Perhitungan AHP. Dan nilai *length* diperoleh dari perhitungan rata-rata *length* untuk field tersebut sesuai Bab 5 bagian Pemrosesan Algoritma BM25F. Dengan hasil debug dari kueri tersebut seperti terlihat pada kode 6.1

Kode 6.1: Hasil debug kueri satu term satu field

```

"debug":{
  "rawquerystring ":"kacang hijau",
  "querystring ":"kacang hijau",
  "parsedquery ":"BM25FBooleanQuery(+food_name :kacang +food_name :hijau)
  ",
  "parsedquery_toString ":"+food_name :kacang +food_name :hijau ",
  "explain":{
    "54":{
      \n5.706025 = sum of:
      \n 2.7444978 = idf(t) * [field scores / (k1) + field scores]
      \n 63.301994 = field scores, sum of:
      \n 63.301994 = score(doc=45,field=food_name, freq=1.0 = tf
      in food_name), division of:
      \n 63.5 = numerator, product of:
      \n 1.0 = tf in food_name
      \n 63.5 = boost[food_name]
      \n 1.0031279 = denominator: ((1 - bField) + bField *
      length / avgFieldLength) :
      \n 4.0 = length(food_name)
      \n 3.9439251 = avgFieldLength(food_name)
      \n 0.22 = lengthBoost(food_name)
      \n 3.1780539 = idf
      \n 10.0 = k1
      \n 2.961527 = idf(t) * [field scores / (k1) + field scores]
      \n 63.301994 = field scores, sum of:
      \n 63.301994 = score(doc=45,field=food_name, freq=1.0 = tf
      in food_name), division of:
      \n 63.5 = numerator, product of:
      \n 1.0 = tf in food_name
      \n 63.5 = boost[food_name]
      \n 1.0031279 = denominator: ((1 - bField) + bField *
      length / avgFieldLength) :
      \n 4.0 = length(food_name)
      \n 3.9439251 = avgFieldLength(food_name)
      \n 0.22 = lengthBoost(food_name)
      \n 3.4293683 = idf
      \n 10.0 = k1",
      ....
    }
  },
  "QParser ":"BM25FQueryParser",
  "timing":{
    "time":16.0,
    ....
  }
}

```

Pada hasil diatas terlihat untuk hasil pencarian paling tinggi memiliki nilai 5.706025. Penjelasan nilai tersebut mengikuti

rumus yang telah dijelaskan dalam rumus 2.1, rumus 2.2, dan rumus 2.3 adalah sebagai berikut:

$$k1 = 10$$

$$\text{boost}(\text{food_name}) = 63.5$$

$$bField(\text{food_name}) = 0.22$$

$$\text{length}(\text{food_name}) = 4$$

$$\text{avgFieldLength}(\text{food_name}) = 3.9439251$$

$$\text{occurs}(\text{kacang}) = 1$$

$$\text{idf}(\text{kacang}) = 3.178053$$

$$\text{weight}(\text{kacang}, d) = \frac{1 \cdot 63.5}{((1-0.22)+0.22 \cdot \frac{4}{3.9439251})} = \frac{63.5}{1.0049763} = 63.301994$$

$$BM25F(\text{kacang}) = 3.178053 \cdot \frac{24.87621}{10+24.87621} = 2.744497$$

$$\text{occurs}(\text{hijau}) = 1$$

$$\text{idf}(\text{hijau}) = 3.4293683$$

$$\text{weight}(\text{hijau}, d) = \frac{1 \cdot 63.5}{((1-0.22)+0.22 \cdot \frac{4}{3.9439251})} = \frac{63.5}{1.0049763} = 63.301994$$

$$BM25F(\text{hijau}) = 3.4293683 \cdot \frac{24.87621}{10+24.87621} = 2.961527$$

$$BM25F = 2.744497 + 2.961527 = \mathbf{5.706025}$$

2. Relevansi hasil pencarian dengan kueri

Pada skenario pengujian ini dilakukan pengecekan relevansi hasil pencarian dengan kueri yang dimasukkan oleh pengguna. Kueri yang dimasukkan lebih dari satu term untuk mengetahui apakah hasil yang didapatkan benar-benar sesuai dengan kueri yang dimasukkan.

Kueri yang dimasukkan ”susu sugar milk ultra” memperoleh hasil sesuai pada gambar 6.2

The screenshot shows a search interface for 'Halal Nutrition Food'. The search bar contains the query 'susu sugar milk ultra' and a 'Search' button. Below the search bar, it indicates that the search took 0.013 seconds and yielded 2 results. The results are listed as follows:

Product Name	Relevance Score
Ultra Milk - Minuman Susu UHT Rasa Coklat (8998009010231) PT. Ultrajaya Milk Industry <i>Fresh Milk, Sugar, Skimmed Milk Powder, Cocoa Powder, Vegetable Stabilizer, Chocolate Artificial Flavor, Salt</i>	12.066771
Tango Susu Vanilla (8991102300544) PT. Ultra Prima Abadi <i>Wheat Flour, Sugar, Vegetable Fat, milk powder, Dextrose, Emulsifier, Salt, Egg Powder, Artificial Vanilla Flavour</i>	10.134773

Gambar 6.2: Pengujian relevansi hasil pencarian dengan kueri

Pada hasil tersebut terlihat bahwa term kueri yang ada bukan hanya ada satu field, term ”susu” berada pada field nama produk, term ”sugar” berada pada field bahan, term ”milk” berada pada field nama produk dan nama produsen, dan term ”ultra” berada pada field nama produk dan nama produsen. Data produk yang memiliki term kueri paling banyak di dalamnya akan ditampilkan paling atas dan memperoleh nilai skor BM25F yang lebih besar.

3. Nilai BM25F dengan varian kueri

Pada skenario pengujian ini dilakukan pencarian dalam aplikasi Halal Nutrition Food menggunakan beberapa kueri yang

kemudian dilihat hasil pencariannya apakah kemunculan term dalam tiap field mempengaruhi pemrosesan nilai BM25F.

Hasil pencarian menggunakan beberapa kueri terlihat pada tabel 6.2

Tabel 6.2: Hasil pencarian dengan varian kueri

Kueri	Hasil ke-	Kemunculan term				Nilai BM25F
		food_code	food_name	food_man	food_ing	
good	1	0	1	1	0	2.2181323
	2	0	1	0	0	2.1952717
	3	0	1	0	0	2.1952717
milk coklat	1	0	2	0	1	5.8168297
	2	0	1	0	2	5.6486883
	3	0	1	0	1	5.6486883
roma mayora biskuit	1	0	2	1	0	8.589926
	2	0	2	1	0	8.589926
susu sugar milk ultra	1	0	3	1	3	12.066771
	2	0	1	1	2	10.134773

Dari tabel diatas terlihat bahwa kemunculan term dalam field yang ada pada dokumen produk mempengaruhi besaran nilai BM25F untuk suatu dokumen produk.

6.1.2 Pengujian Kinerja

Pengujian Kinerja yang dilakukan kepada aplikasi Halal Nutrition Food dilakukan dengan menguji performa aplikasi menggunakan aplikasi JMeter. Uji coba performa bertujuan untuk mengetahui kemampuan aplikasi dalam menangani permintaan dari banyak pengguna dalam waktu yang bersamaan dengan skenario jumlah pengguna 10 orang, 50 orang, dan 100 orang. Detail skenario pengujian performa dapat dilihat pada tabel 6.3.

Skenario dalam yang telah dilakukan dalam tabel diatas menggunakan 4 variasi kueri dengan jumlah term yang berbeda. Dari hasil

Tabel 6.3: Hasil pengujian performa aplikasi berdasarkan kueri

Kueri	Jumlah Pengguna			Hasil	AVG Bytes
	10	50	100		
susu	0.175 detik	0,183 detik	0.194 detik	6	55.63 Bytes
susu sugar	0.177 detik	0.178 detik	0.185 detik	5	53.65 Bytes
susu sugar milk	0.180 detik	0.185 detik	0.189 detik	5	53.77 Bytes
susu sugar milk ultra	0.187 detik	0.192 detik	0.199 detik	2	47.12 Bytes

pengujian akan terlihat hasil pencarian yang diperoleh, waktu respon server, dan rata-rata Bytes dari bandwidth yang dihasilkan oleh aplikasi dalam melakukan pencarian dari kueri yang dimasukkan oleh pengguna.

6.2 Pembahasan

Pada subbab ini akan dibahas dan disimpulkan hasil dari pengujian fungsional dan kinerja dari aplikasi Halal Nutrition Food.

6.2.1 Pembahasan Uji Algoritma BM25F

Pada pengujian algoritma aplikasi telah dilakukan beberapa skenario, yakni skenario menggunakan varian kueri term dalam field seperti pada pengujian pencarian kueri satu term satu field, pencarian kueri satu term multi field, pencarian kueri multi term satu field, dan pencarian kueri multi term multi field didapatkan hasil bahwa skor hasil pencarian dalam aplikasi sudah sesuai dengan kaidah algoritma BM25F dan menghasilkan hasil pencarian yang cocok dengan kueri yang diminta oleh pengguna.

Pada skenario relevansi hasil pencarian dengan kueri didapatkan

hasil bahwa pencarian bisa dilakukan pada beragam field seperti field nama produk, kode produk, nama produsen, maupun nama bahan-bahan penyusun produk menyesuaikan dengan term yang ada dalam kueri yang dimasukkan oleh pengguna.

Sementara pengujian terakhir yaitu pencarian kueri dalam field dengan variasi nilai *boost* didapatkan hasil bahwa nilai *boost* untuk setiap field sangat berpengaruh pada skor hasil pencarian, sehingga aplikasi perlu mencocokkan field mana yang memiliki nilai tertinggi agar hasil yang didapatkan lebih optimal.

6.2.2 Pembahasan Uji Kinerja

Pengujian kinerja dilakukan untuk mengamati performa aplikasi dalam menampilkan atau merespon permintaan pengguna dengan jumlah data dan pengguna yang tidak sama. Dari skenario yang dilakukan, terlihat bahwa jumlah term dalam kueri dan jumlah pengguna yang mengakses dalam waktu bersamaan mempengaruhi waktu respon server terhadap permintaan pengguna. Misalkan saja ketika term hanya berjumlah 1 kata waktu respon server lebih cepat dibandingkan dengan kueri dengan term lebih dari satu, dan jumlah 10 pengguna yang mengakses server secara bersamaan waktu respon server lebih cepat dibandingkan dengan ketika jumlah pengguna mencapai 50 dan 100 pengguna. Kemudian besaran rata-rata bandwidth yang dihasilkan oleh server bergantung pada hasil pencarian yang ditemukan. Terlihat ketika hasil pencarian berjumlah 6 baris maka rata-rata bandwidth lebih besar daripada rata-rata bandwidth ketika hasil pencarian berjumlah 2 baris.

BAB 7

KESIMPULAN DAN SARAN

Pada bab ini akan dijelaskan mengenai kesimpulan dan saran dari hasil pengerjaan tugas akhir.

7.1 Kesimpulan

Berdasarkan aktifitas yang telah dilakukan dalam pengerjaan tugas akhir dengan judul "Rancang Bangun Sistem Pencarian Dalam Aplikasi Halal Nutrition Food Menggunakan Algoritma OKAPI BM25F" yang telah dilakukan, didapatkan kesimpulan sebagai berikut:

1. Pencarian produk halal yang metode skornya menggunakan algoritma Okapi BM25F berhasil dibuat dalam aplikasi Halal Nutrition Food.
2. Dalam melakukan pencarian, aplikasi Halal Nutrition Food memiliki 2 entitas utama yaitu basis data MySQL dan Apache Solr. Data produk halal yang disimpan dalam basis data MySQL selanjutnya dilakukan pengindeksan dalam Apache Solr.
3. Alur pencarian data adalah ketika pengguna memasukkan kueri pencarian, kueri tersebut akan dilakukan perhitungan oleh plugin BM25F berdasarkan data indeks yang tersimpan di dalam Apache Solr dan hasil pencariannya akan ditampilkan kembali kepada pengguna pada halaman aplikasi.
4. Sistem pencarian produk halal pada aplikasi Halal Nutrition Food yang dilakukan dalam tugas akhir ini menggunakan algoritma BM25F dalam kueri dimana setiap field dalam data

memiliki nilai sendiri yang akan menentukan tingkat prioritas setiap field tersebut dalam pencarian data.

5. Dalam melakukan pencarian produk halal dalam aplikasi pada tugas akhir ini pengguna dapat mencari data produk didasarkan pada nama produk, kode produk, nama produsen, dan nama bahan-bahan penyusun produk sementara pada aplikasi versi sebelumnya hanya bisa mencari berdasarkan nama produk dan bahan aditif saja.

7.2 Saran

Saran penulis untuk penelitian yang akan dibuat selanjutnya adalah sebagai berikut:

1. Untuk penelitian selanjutnya dapat melihat log kueri pencarian kueri yang dilakukan pengguna pada aplikasi yang sudah berjalan agar bisa ditentukan field mana yang seharusnya diberikan nilai boost paling tinggi supaya hasil pencarian lebih relevan.
2. Penelitian ini berfokus pada aplikasi yang berbasis website. Selanjutnya peneliti berharap aplikasi dapat dikembangkan dalam platform mobile sehingga pengguna lebih mudah untuk melakukan pencarian atau memasukkan data produk halal ke dalam aplikasi.

DAFTAR PUSTAKA

- [1] Diego Cecarelli. Solr bm25f plugin, 2017. Dapat diakses di <https://github.com/europeana/contrib/tree/master/bm25f-ranking>.
- [2] Renaud Delbru, Nur Aini Rakhmawati, and Giovanni Tummarello. Sindice at semsearch 2010. In *Proceedings of the 19th International World Wide Web Conference, Raleigh, North Carolina, USA*. Citeseer, 2010.
- [3] Renaud Delbru, Nickolai Toupikov, Michele Catasta, Giovanni Tummarello, and Stefan Decker. Hierarchical link analysis for ranking web data. In *Extended Semantic Web Conference*, pages 225–239. Springer, 2010.
- [4] Jauhar Fatawi. Rancang bangun perangkat lunak linked open data halal dan gizi pada produk makanan dan minuman. 2016.
- [5] BBC Guus Schreiber, VU University Amsterdam Yves Raimond. Rdf 1.1 primer, 2017. Dapat diakses di <https://www.techopedia.com/definition/23731/semantic-search>.
- [6] Michael Hausenblas. Interlinking multimedia: how to apply linked data principles to multimedia fragments. 2009.
- [7] Tom Heath, Michael Hausenblas, Chris Bizer, Richard Cyganiak, and Olaf Hartig. How to publish linked data on the web. In *Tutorial in the 7th International Semantic Web Conference, Karlsruhe, Germany*, 2008.
- [8] Angga Indrawan. Inilah 10 negara dengan populasi muslim terbesar di dunia, 2017. Dapat diakses di <http://www.republika.co.id/berita/dunia-islam/islam-nusantara/15/05/27/noywh5-inilah-10-negara-dengan-populasi-muslim-terbesar-di-dunia>.

- [9] Kelvin. Basic concepts lucene, 2017. Dapat diakses di <http://www.lucenetutorial.com/basic-concepts.html>.
- [10] Ralph R. Swick Ora Lassila. Resource description framework (rdf) model and syntax specification, 2017. Dapat diakses di <https://www.w3.org/TR/1999/REC-rdf-syntax-19990222/>.
- [11] José R Pérez-Agüera, Javier Arroyo, Jane Greenberg, Joaquín Pérez Iglesias, and Víctor Fresno. Using bm25f for semantic search. In *Proceedings of the 3rd international semantic search workshop*, page 2. ACM, 2010.
- [12] Joaquín Pérez-Iglesias, José R Pérez-Agüera, Víctor Fresno, and Yuval Z Feinstein. Integrating the probabilistic models bm25/bm25f into lucene. *arXiv preprint arXiv:0911.5046*, 2009.
- [13] Stephen Robertson, Hugo Zaragoza, and Michael Taylor. Simple bm25 extension to multiple weighted fields. In *Proceedings of the thirteenth ACM international conference on Information and knowledge management*, pages 42–49. ACM, 2004.
- [14] Tedy Saputra. Tuning parameter dalam fungsi okapi bm25 pada mesin pencari teks bahasa indonesia. 2013.
- [15] Techopedia. Semantic search, 2017. Dapat diakses di <https://www.w3.org/TR/rdf11-primer/>.
- [16] W3C. Linked data - w3c, 2016. Dapat diakses di <https://www.w3.org/standards/semanticweb/data>.
- [17] Webopedia. Semantic web application, 2017. Dapat diakses di http://www.webopedia.com/TERM/S/Semantic_Web_application.html.

LAMPIRAN A

KUISIONER

Kuisioner Pencarian Produk Halal

Perkenalkan saya Adnan Mauludin Fajriyadi NRP 5213100103, mahasiswa departemen Sistem Informasi ITS, Laboratorium Akuisisi Data dan Diseminasi Informasi yang sedang melakukan penelitian mengenai pencarian produk halal.

Saya meminta kesediaan saudara untuk mengisi kuisioner dalam rangka mengetahui kecenderungan konsumen dalam melakukan pencarian produk halal. Partisipasi saudara sangat diperlukan dalam rangka mengembangkan aplikasi ini nantinya. Sebelumnya saya ucapkan terima kasih banyak atas kesediaan waktunya.

Nama : _____

Aplikasi Halal Nutrition Food

Halal Nutrition Food adalah aplikasi yang sedang dikembangkan dalam penelitian. Aplikasi ini berisi database produk halal yang salah satu fiturnya adalah pencarian produk halal. Dalam fitur pencarian terdapat 4 kategori kata kunci yaitu Kode Produk, Nama Produk, Nama Produsen, dan Nama Bahan Makanan.

Oleh karena itu saya meminta pendapat saudara agar aplikasi ini nantinya bisa berjalan optimal sesuai dengan kebutuhan pengguna.

Lebih penting mana Kode Produk dibanding Nama Produk?

a. Kode produk

b. Nama produk

Seberapa penting kategori yang Anda pilih terhadap kategori yang tidak terpilih?

Isikan angka dengan skala 1 (Sama pentingnya) hingga 9 (Sangat penting): -----

Lebih penting mana Kode Produk dibanding Nama Produsen?

- a. Kode produk
- b. Nama produsen

Seberapa penting kategori yang Anda pilih terhadap kategori yang tidak terpilih?

Isikan angka dengan skala 1 (Sama pentingnya) hingga 9 (Sangat penting): -----

Lebih penting mana Kode Produk dibanding Nama Bahan Makanan?

- a. Kode produk
- b. Nama bahan makanan

Seberapa penting kategori yang Anda pilih terhadap kategori yang tidak terpilih?

Isikan angka dengan skala 1 (Sama pentingnya) hingga 9 (Sangat penting): -----

Lebih penting mana Nama Produk dibanding Nama Produsen?

- a. Nama produk
- b. Nama produsen

Seberapa penting kategori yang Anda pilih terhadap kategori yang tidak terpilih?

Isikan angka dengan skala 1 (Sama pentingnya) hingga 9 (Sangat penting): -----

Lebih penting mana Nama Produk dibanding Nama Bahan Makanan?

- a. Nama produk
- b. Nama bahan makanan

Seberapa penting kategori yang Anda pilih terhadap kategori yang tidak terpilih?

Isikan angka dengan skala 1 (Sama pentingnya) hingga 9 (Sangat penting): -----

Lebih penting mana Nama Produsen dibanding Nama Bahan Makanan?

- a. Nama produsen
- b. Nama bahan makanan

Seberapa penting kategori yang Anda pilih terhadap kategori yang tidak terpilih?

Isikan angka dengan skala 1 (Sama pentingnya) hingga 9 (Sangat penting): -----

Halaman ini sengaja dikosongkan

BIODATA PENULIS



Penulis dilahirkan di Jakarta pada tanggal 7 September 1995. Merupakan anak pertama dari tiga bersaudara yang telah menempuh pendidikan formal yaitu SD Negeri 01 Cipinang Melayu, MTs Negeri 14 Jakarta, dan MA Negeri 9 Jakarta.

Pada tahun 2013 penulis melanjutkan pendidikan di Departemen Sistem Informasi FTIF - Institut Teknologi Sepuluh Nopember (ITS) Surabaya dan terdaftar sebagai mahasiswa

dengan NRP 5213100103.

Selama menjadi mahasiswa penulis aktif mengikuti berbagai kegiatan kemahasiswaan baik berupa kepanitiaan dan organisasi di tingkat Departemen, Institut, dan Nasional. Organisasi yang aktif diikuti oleh penulis adalah Badan Eksekutif Mahasiswa (BEM) ITS dimana penulis telah menjadi pengurus selama 3 periode kepemimpinan mulai periode 2014/2015, 2015/2016, dan 2016/2017 dalam Kementerian Kominfo BEM ITS. Pada periode kepemimpinan BEM ITS yang ketiga, penulis diamanahi sebagai Menteri Komunikasi dan Informatika (Kominfo) BEM ITS.

Pada tahun keempat penulis tertarik dengan bidang *linked data* dan *semantic web*, sehingga penulis mengambil bidang minat Laboratorium Akuisisi Data dan Diseminasi Informasi (ADDI). Lebih lanjut penulis dapat dihubungi melalui email me@adnanfajri.com.