



ITS
Institut
Teknologi
Sepuluh Nopember

TUGAS AKHIR - KS141501

INTEGRASI DATA DAN VISUALISASI GRAF PADA APLIKASI HALAL NUTRITION FOOD

DATA INTEGRATION AND GRAPH VISUALIZATION IN HALAL NUTRITION FOOD APPLICATION

**BERLIAN FAJAR YUSUF
NRP 05211 4400 00144**

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

**DEPARTEMEN SISTEM INFORMASI
Fakultas Teknologi Informasi dan Komunikasi
Institut Teknologi Sepuluh Nopember
Surabaya 2018**



ITS
Institut
Teknologi
Sepuluh Nopember

TUGAS AKHIR - KS141501

INTEGRASI DATA DAN VISUALISASI GRAF PADA APLIKASI HALAL NUTRITION FOOD

BERLIAN FAJAR YUSUF
NRP 05211 4400 00144

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

DÉPARTEMEN SISTEM INFORMASI
Fakultas Teknologi Informasi dan Komunikasi
Institut Teknologi Sepuluh Nopember
Surabaya 2018



ITS
Institut
Teknologi
Sepuluh Nopember

FINAL PROJECT - KS141501

***DATA INTEGRATION AND GRAPH VISUALIZATION
IN HALAL NUTRITION FOOD APPLICATION***

**BERLIAN FAJAR YUSUF
NRP 05211 4400 00144**

Supervisor :

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

Irmasari Hafidz, S.Kom., M.Sc.

**Information Systems Department
Faculty of Information and Communication Technology
Institut Teknologi Sepuluh Nopember
Surabaya 2018**

LEMBAR PENGESAHAN

**INTEGRASI DATA DAN VISUALISASI GRAF PADA
APLIKASI HALAL NUTRITION FOOD**

TUGAS AKHIR

**Disusun Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada**

**Departemen Sistem Informasi
Fakultas Teknologi Informasi dan Komunikasi
Institut Teknologi Sepuluh Nopember**

Oleh:

BERLIAN FAJAR YUSUF

NRP. 05211 4400 00144

Surabaya, 18 Juli 2018

**KEPALA
DEPARTEMEN SISTEM INFORMASI**

Dr. Ir. Aris Tjahyanto, M.Kom.
NIP. 19650310 199102 1 001

LEMBAR PERSETUJUAN
INTEGRASI DATA DAN VISUALISASI GRAF PADA
APLIKASI HALAL NUTRITION FOOD

TUGAS AKHIR

Disusun Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada
Departemen Sistem Informasi
Fakultas Teknologi Informasi dan Komunikasi
Institut Teknologi Sepuluh Nopember

Oleh:

BERLIAN FAJAR YUSUF
NRP. 05211 4400 00144

Disetujui Tim Penguji: Tanggal Ujian : 9 Juli 2018
Periode Wisuda : September 2018

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

Irmasari Hafidz, S.Kom., M.Sc. (Pembimbing II)

Faizal Johan Atletiko, S.Kom., M.T. (Penguji I)

Radityo Prasetyanto Wibowo, S.Kom., (Penguji II)
M.Kom.

INTEGRASI DATA DAN VISUALISASI GRAF PADA APLIKASI HALAL NUTRITION FOOD

Nama Mahasiswa : Berlian Fajar Yusuf
NRP : 05211 4400 00144
Jurusan : Sistem Informasi FTIK-ITS
Pembimbing 1 : Nur Aini Rakhmawati, S.Kom.,
M.Sc.Eng., Ph.D.
Pembimbing 2 : Irmasari Hafidz, S.Kom., M.Sc.

ABSTRAK

Halal Nutrition Food merupakan sebuah aplikasi pencarian produk halal berbasis web yang menampilkan berbagai informasi produk halal secara rinci yang didapatkan dari gabungan dataset yang diintegrasikan dalam bentuk Linked Open Data. Namun, saat ini, Halal Nutrition Food masih memiliki sedikit jumlah produk beserta informasinya dan juga masih belum memiliki visualisasi sehingga pengguna tidak dapat memiliki wawasan mengenai data dengan mudah. Open Food Fact merupakan open data produk makanan beserta informasinya yang berasal dari seluruh dunia. Untuk itu, perlu adanya integrasi data antara Open Food Fact dan Halal Nutrition Food. Kami hanya mengambil data produk yang memiliki bahan makanan dalam bahasa inggris dan perancis. Data tersebut dibersihkan melalui data cleansing yang di dalamnya terdapat pengukuran kesamaan bahan makanan menggunakan Levenshtein dan Jaccard distance untuk memperbaiki kesalahan penulisan. Hasil data cleansing diintegrasikan ke Halal Nutrition Food. Untuk mengetahui kelompok makanan apa saja yang ada, data yang sudah dalam bentuk graf produk dan bahan makanan, dikelompokkan menggunakan METIS graph partitioning. Hasil data cleansing menunjukkan bahwa untuk memperbaiki kesalahan penulisan, Levenshtein distance lebih baik daripada Jaccard distance. Pengukuran kesamaan menggunakan Jaccard distance

menunjukkan bahwa ada terlalu banyak bahan makanan yang mirip. Padahal, sebenarnya tidak ada kesalahan penulisan dan bahan makanan tersebut memiliki perbedaan arti. Produk dan bahan makanan dikelompokkan menjadi 20 kelompok, sesuai pengelompokan produk yang dilakukan oleh MUI. Hasil graph partitioning divisualisikan bersama dengan visualisasi graf hubungan produk dan bahan makanan, visualisasi graf produk makanan yang berstatus haram dan visualisasi graf produk makanan yang mengandung MSG.

Kata kunci: halal, levenshtein, jaccard, integrasi data, graph partitioning, visualisasi graf

DATA INTEGRATION AND GRAPH VISUALIZATION IN HALAL NUTRITION FOOD APPLICATION

Student Name : Berlian Fajar Yusuf
NRP : 05211 4400 00144
Department : Sistem Informasi FTIK-ITS
Supervisor 1 : Nur Aini Rakhmawati, S.Kom.,
M.Sc.Eng., Ph.D.
Supervisor 2 : Irmasari Hafidz, S.Kom., M.Sc.

ABSTRACT

Halal Nutrition Food is a web-based halal product search application featuring a variety of halal product information in detail obtained from a combined dataset that is integrated into the form of Linked Open Data. However, currently, Halal Nutrition Food still has a small number of products along with its information and also still has no visualization so users can not have insight into data easily. Open Food Fact is an open data of food products along with information from all over the world. So, there is a need for data integration between Open Food Fact and Halal Nutrition Food. We only collect food products in English and French from Open Food Fact. Those data are cleaned through data cleansing with similarity measurement using Levenshtein and Jaccard distance to correct typos. The data cleansing results are integrated into Halal Nutrition Food. To reveal what food product/ingredient clusters are, the data that already in the product and food graphs, are clustered using METIS graph partitioning. The data cleansing results show that to correct typos, the Levenshtein distance is better than the Jaccard distance. Jaccard distance similarity measurement shows that there are too many similar ingredients. In fact, there are no typos and those have the different meaning. Food products and ingredients are clustered into 20 clusters, according to the MUI product grouping. The results of graph partitioning are visualized along with graph

*visualization of food products and ingredients relationships,
haram food products and food products that contain MSG.*

***Keyword: halal, levenshtein, jaccard, data integration,
graph partitioning, graph visualization***

KATA PENGANTAR

Alhamdulillahirobbil'aalamin. Puji syukur kami panjatkan kehadiran Allah, Tuhan Yang Maha Esa, karena dengan limpahan rahmat-Nya penulis dapat menyelesaikan laporan tugas akhir dengan judul:

INTEGRASI DATA DAN VISUALISASI GRAF PADA APLIKASI HALAL NUTRITION FOOD

yang merupakan salah satu syarat kelulusan pada Departemen Sistem Informasi, Fakultas Teknologi Informasi dan Komunikasi, Institut Teknologi Sepuluh Nopember Surabaya.

Terima kasih penulis sampaikan kepada pihak-pihak yang telah mendukung, memberikan saran, motivasi, semangat, dan bantuan baik berupa materiil maupun moril demi tercapainya tujuan pembuatan tugas akhir ini. Tugas akhir ini tidak akan pernah terwujud tanpa bantuan dan dukungan dari berbagai pihak yang sudah melauangkan waktu, tenaga dan pikirannya. Secara khusus penulis akan menyampaikan ucapan terima kasih yang sebanyak-banyaknya kepada:

1. Allah SWT yang senantiasa melimpahkan berkah dan rahmat-Nya selama penulis mengerjakan Tugas Akhir ini.
2. Kedua orang tua, saudara, serta keluarga tercinta yang selalu mendukung serta mendoakan tiada henti untuk menyelesaikan Tugas Akhir ini.
3. Ibu Nur Aini Rakhmawati, S.Kom., M.Sc.Eng., Ph.D. dan Ibu Irmasari Hafidz, S.Kom., M.Sc. selaku dosen pembimbing yang telah meluangkan waktu, memberikan ilmu, petunjuk, dan motivasi selama pengerjaan Tugas Akhir.
4. Bapak Nisfu Asrul Sani, S.Kom., M.Sc. selaku dosen wali yang selalu memberikan saran dan motivasi kepada kepada penulis.
5. Seluruh dosen Departemen Sistem Informasi FTIK-ITS yang telah memberikan ilmu yang bermanfaat bagi penulis.

6. Teman-teman Laboratorium RDIB, ADDI, dan MSI yang selalu menanyakan progress Tugas Akhir penulis.
7. Keluarga OSIRIS yang telah memberikan banyak pembelajaran, kenangan manis dan pahit semasa kuliah.
8. Dan seluruh pihak yang telah membantu penulis secara langsung maupun tidak langsung sehingga tugas akhir ini dapat terselesaikan dengan baik.

Penulis menyadari penyusunan laporan ini masih jauh dari kata sempurna sehingga penulis menerima adanya kritik maupun saran yang membangun untuk perbaikan di masa yang akan datang. Semoga buku tugas akhir ini dapat memberikan manfaat bagi pembaca.

Surabaya, 28 Juni 2018

Berlian Fajar Yusuf

DAFTAR ISI

| | |
|---|-------|
| ABSTRAK | v |
| ABSTRACT | vii |
| KATA PENGANTAR | ix |
| DAFTAR ISI | xi |
| DAFTAR GAMBAR | xvii |
| DAFTAR TABEL | xxi |
| DAFTAR KODE | xxiii |
| BAB I PENDAHULUAN | 1 |
| 1.1. Latar Belakang | 1 |
| 1.2. Rumusan Masalah | 2 |
| 1.3. Batasan Masalah | 3 |
| 1.4. Tujuan Penelitian | 3 |
| 1.5. Manfaat Penelitian | 3 |
| 1.6. Relevansi | 4 |
| BAB II TINJAUAN PUSTAKA | 5 |
| 2.1. Penelitian Sebelumnya | 5 |
| 2.2. Dasar Teori | 7 |
| 2.2.1. Halal Nutrition Food | 7 |
| 2.2.2. Open Food Fact | 8 |
| 2.2.3. RDF | 9 |
| 2.2.4. Levenshtein Distance | 10 |
| 2.2.5. Jaccard Distance | 11 |
| 2.2.6. Apache Lucene | 11 |
| 2.2.7. Graph Partitioning | 12 |
| 2.2.8. D3js | 14 |
| BAB III METODOLOGI PENELITIAN | 15 |
| 3.1. Studi Literatur | 16 |
| 3.2. Akuisisi Data | 16 |
| 3.2.1. Pengamatan Skema RDF pada Open Food Fact dan Halal Nutrition Food | 16 |
| 3.3. Data Cleansing | 19 |
| 3.3.1. Casefolding | 20 |
| 3.3.2. Penghilangan/Penggantian Karakter yang Tidak Diperlukan | 20 |

| | | |
|---------------------------------|---|-----------|
| 3.3.3. | Penerjemahan Bahan Makanan yang Berbahasa Perancis ke Bahasa Inggris..... | 22 |
| 3.3.4. | Pengubahan Huruf Aksen ke Non-Aksen..... | 22 |
| 3.3.5. | Penghilangan Bahan Makanan Berbahasa Non Inggris yang Masih Tersisa | 23 |
| 3.3.6. | Stopword Removal..... | 23 |
| 3.3.7. | Tokenizing | 24 |
| 3.3.8. | Penghilangan Data Kotor yang Mirip Stopword | 24 |
| 3.3.9. | Pengukuran Kesamaan pada Bahan Makanan | 25 |
| 3.3.10. | Supervisi Manual terhadap Bahan Makanan | 27 |
| 3.4. | Integrasi Data..... | 27 |
| 3.5. | Graph Partitioning | 28 |
| 3.5.1. | Mengubah Data Graph ke Format Masukan METIS | 28 |
| 3.5.2. | Melakukan Graph Partitioning | 29 |
| 3.6. | Visualisasi Graf | 29 |
| 3.7. | Pengujian | 31 |
| 3.8. | Penyusunan Buku Tugas Akhir | 32 |
| 3.9. | Arsitektur Sistem | 33 |
| BAB IV PERANCANGAN | | 35 |
| 4.1. | Akuisisi Data | 35 |
| 4.2. | Data Cleansing..... | 36 |
| 4.2.1. | Casefolding | 37 |
| 4.2.2. | Penghilangan/Penggantian Karakter yang Tidak Diperlukan | 37 |
| 4.2.3. | Penerjemahan Bahan Makanan yang Berbahasa Perancis ke Bahasa Inggris..... | 38 |
| 4.2.4. | Pengubahan Huruf Aksen menjadi Non Aksen | 38 |
| 4.2.5. | Penghilangan Bahan Makanan Berbahasa Non Inggris yang Masih Tersisa | 38 |
| 4.2.6. | Stopword Removal..... | 39 |
| 4.2.7. | Tokenizing | 42 |
| 4.2.8. | Penghilangan Data Kotor yang Mirip Stopword | 42 |

| | | |
|--------------------------|---|----|
| 4.2.9. | Pengukuran Kesamaan pada Bahan Makanan | 43 |
| 4.2.10. | Supervisi Manual terhadap Bahan Makanan | 44 |
| 4.3. | Integrasi Data | 44 |
| 4.3.1. | Impor Data ke Basis Data | 44 |
| 4.3.2. | Melakukan Independent Ranking..... | 45 |
| 4.3.3. | Menghasilkan File Turtle | 46 |
| 4.3.4. | Mengindeks Produk dan Bahan Makanan | 46 |
| 4.3.5. | Menghasilkan Produk Terkait..... | 46 |
| 4.4. | Graph Partitioning | 46 |
| 4.4.1. | Membuat File Masukan METIS..... | 46 |
| 4.4.2. | Melakukan Graph Partitioning di METIS | 47 |
| 4.5. | Visualisasi Graf | 48 |
| 4.5.1. | Desain Antarmuka Visualisasi Graf | 48 |
| 4.6. | Pengujian..... | 50 |
| 4.6.1. | Pengujian Jumlah Data yang Dihasilkan setelah Pengukuran Kesamaan Bahan Makanan | 50 |
| 4.6.2. | Pengujian Fungsional Visualisasi Graf | 51 |
| BAB V IMPLEMENTASI | | 53 |
| 5.1. | Lingkungan Penelitian | 53 |
| 5.2. | Akuisisi Data | 53 |
| 5.3. | Data Cleansing | 55 |
| 5.3.1. | Casefolding..... | 56 |
| 5.3.2. | Penghilangan/Penggantian Karakter yang Tidak Diperlukan | 56 |
| 5.3.3. | Penerjemahan Bahan Makanan yang Berbahasa Perancis ke Bahasa Inggris | 56 |
| 5.3.4. | Pengubahan Huruf Aksen menjadi Non Aksen | 56 |
| 5.3.5. | Penghilangan Data Berbahasa Non Inggris yang Masih Tersisa | 57 |
| 5.3.6. | Stopword Removal | 60 |
| 5.3.7. | Tokenizing | 60 |
| 5.3.8. | Penghilangan Data Kotor yang Mirip Stopword | 60 |
| 5.3.9. | Pengukuran Kesamaan pada Bahan Makanan | 62 |

| | |
|---|-----------|
| 5.3.10. Supervisi Manual terhadap Bahan Makanan | 64 |
| 5.4. Integrasi Data..... | 65 |
| 5.4.1. Impor Data ke Basis Data | 65 |
| 5.4.2. Melakukan Independent Ranking | 67 |
| 5.4.3. Menghasilkan File Turtle | 67 |
| 5.4.4. Mengindeks Produk dan Bahan Makanan | 67 |
| 5.4.5. Menghasilkan Produk Terkait | 68 |
| 5.5. Graph Partitioning | 68 |
| 5.5.1. Mengubah File Masukan METIS | 68 |
| 5.5.2. Melakukan Graph Partitioning | 68 |
| 5.6. Visualisasi Graf | 70 |
| BAB VI HASIL DAN PEMBAHASAN | 75 |
| 6.1. Hasil | 75 |
| 6.1.1. Hasil Pengujian Jumlah Data yang Dihasilkan setelah Pengukuran Kesamaan Bahan Makanan | 75 |
| 6.1.2. Hasil Graph Partitioning | 75 |
| 6.1.3. Hasil Visualisasi Graf | 75 |
| 6.1.4. Hasil Pengujian Fungsionalitas Visualisasi Graf..... | 80 |
| 6.2. Pembahasan | 81 |
| 6.2.1. Pembahasan Hasil Pengujian Jumlah Data yang Dihasilkan setelah Pengukuran Kesamaan Bahan Makanan | 81 |
| 6.2.2. Pembahasan Hasil Graph Partitioning | 83 |
| 6.2.3. Pembahasan Hasil Pengujian Fungsionalitas Visualisasi Graf..... | 87 |
| BAB VII KESIMPULAN DAN SARAN..... | 89 |
| 7.1. Kesimpulan..... | 89 |
| 7.2. Saran..... | 89 |
| DAFTAR PUSTAKA..... | 91 |
| A. LAMPIRAN A : KODE PROGRAM..... | 1 |
| A.1 Pembuatan File Masukan Metis | 1 |
| B. LAMPIRAN B : DATA..... | 1 |
| B.1 Contoh file rdf open food fact..... | 1 |
| B.2 Contoh file turtle pada halal nutrition food | 1 |
| B.3 Data sebelum data cleansing | 1 |

| | | |
|------|---|-----|
| B.4 | Data masukan deteksi bahasa menggunakan langdetect | 1 |
| B.5 | Data hasil deteksi bahasa menggunakan langdetect.... | 1 |
| B.6 | Data masukan deteksi bahasa menggunakan pengecekan dictionary | 1 |
| B.7 | Data dictionary bahasa inggris dan perancis | 1 |
| B.8 | Data hasil deteksi bahasa menggunakan pengecekan dictionary..... | 1 |
| B.9 | Data daftar stopword | 1 |
| B.10 | Data daftar bahan makanan sebagai masukan pengukuran kesamaan bahan makanan | 1 |
| B.11 | Data hasil pengukuran kesamaan menggunakan levenshtein distance | 1 |
| B.12 | Data hasil pengukuran kesamaan menggunakan jaccard distance..... | 1 |
| B.13 | Data daftar bahan makanan setelah pengukuran kesamaan | 1 |
| B.14 | Data daftar bahan makanan yang terintegrasi | 1 |
| B.15 | Data impor csv ke basis data halal nutrition food | 1 |
| B.16 | Data file turtle yang telah diindeks | 2 |
| B.17 | Data file masukan METIS | 2 |
| B.18 | Data file keluaran METIS..... | 2 |
| B.19 | Data file JSON sebagai masukan visualisasi graf | 2 |
| C. | LAMPIRAN C : HASIL GRAPH PARTITIONING.. | 1 |
| | BIODATA PENULIS | 101 |

Halaman ini sengaja dikosongkan

DAFTAR GAMBAR

| | |
|---|----|
| Gambar 2.1 Contoh Levenshtein Distance | 10 |
| Gambar 2.2 Proses pada Apache Lucene | 12 |
| Gambar 2.3 Ilustrasi Graph Partitioning..... | 13 |
| Gambar 2.4 Multilevel Graph Partitioning | 14 |
| Gambar 3.1 Tahapan Pengerjaan Tugas Akhir | 15 |
| Gambar 3.2 Ontology yang Digunakan pada Open Food Fact | 16 |
| Gambar 3.3 Vocabulary Halal Nutrition Food | 19 |
| Gambar 3.4 Ontology/Vocabulary pada Halal Nutrition Food | 19 |
| Gambar 3.5 Contoh Pengukuran Kesamaan Menggunakan Levenshtein Distance | 24 |
| Gambar 3.6 Contoh Pengukuran Kesamaan Menggunakan Jaccard Distance..... | 26 |
| Gambar 3.7 Contoh File Input METIS | 28 |
| Gambar 3.8 Contoh File Output METIS | 29 |
| Gambar 3.9 Contoh Visualisasi Graf Hubungan Antara Nama Produk, Bahan Makanan | 30 |
| Gambar 3.10 Contoh Visualisasi Hasil Graph Partitioning ... | 30 |
| Gambar 3.11 Contoh Visualisasi Graf Produk Berstatus Haram | 31 |
| Gambar 3.12 Contoh Visualisasi Graf Produk Ber-MSG..... | 31 |
| Gambar 3.13 Arsitektur Sistem Integrasi Data dan Visualisasi Graf pada Halal Nutrition Food..... | 33 |
| Gambar 4.1 Alur Akuisisi Data | 35 |
| Gambar 4.2 Alur Data Cleansing | 36 |
| Gambar 4.3 Contoh Produk Makanan beserta Bahannya | 36 |
| Gambar 4.4 Alur Casefolding | 37 |
| Gambar 4.5 Alur Penghilangan/Penggantian Karakter yang Tidak Diperlukan | 37 |
| Gambar 4.6 Alur Penerjemahan Bahan Makanan yang Berbahasa Perancis ke Bahasa Inggris | 38 |
| Gambar 4.7 Alur Pengubahan Huruf Aksen menjadi Non Aksen | 38 |

| | |
|---|----|
| Gambar 4.8 Penghilangan Data Berbahasa Non Inggris yang Masih Tersisa | 39 |
| Gambar 4.9 Alur Stopword Removal | 40 |
| Gambar 4.10 Alur Tokenizing | 42 |
| Gambar 4.11 Alur Penghilangan Data Kotor yang Mirip Stopword | 43 |
| Gambar 4.12 Alur Pengukuran Kesamaan antar Bahan Makanan..... | 44 |
| Gambar 4.13 Struktur Basis Data Halal Nutrition Food yang Terkait dengan Data Open Food Fact | 45 |
| Gambar 4.14 Struktur Tabel foodproduct_ingredient | 47 |
| Gambar 4.15 Struktur Tabel ingredient_foodproduct..... | 47 |
| Gambar 4.16 Desain Antarmuka Halaman Utama Graph Visualization | 48 |
| Gambar 4.17 Desain Antarmuka Halaman Visualisasi Graf Hubungan Produk dan Bahan Makanan | 49 |
| Gambar 4.18 Desain Antarmuka Halaman Visualisasi Hasil Graph Partitioning | 49 |
| Gambar 4.19 Desain Antarmuka Halaman Visualisasi Graf Produk Berstatus Haram..... | 50 |
| Gambar 4.20 Desain Antarmuka Halaman Visualisasi Graf Produk Ber-MSG | 50 |
| Gambar 5.1 Contoh File Impor Batch Replace EmEditor | 55 |
| Gambar 5.2 Batch Replace pada EmEditor | 56 |
| Gambar 5.3 Cuplikan Daftar Bahan Makanan..... | 57 |
| Gambar 5.4 Cuplikan Hasil Deteksi Bahasa menggunakan Langdetect..... | 58 |
| Gambar 5.5 Cuplikan Hasil Deteksi Bahasa menggunakan Pengecekan Dictionary..... | 59 |
| Gambar 5.6 Cuplikan Hasil Pengukuran Kesamaan Stopword dan Bahan Makanan | 61 |
| Gambar 5.7 Cuplikan Hasil Pengukuran Kesamaan pada Bahan Makanan Menggunakan Levenshtein Distance | 64 |
| Gambar 5.8 Cuplikan Hasil Pengukuran Kesamaan pada Bahan Makanan Menggunakan Jaccard Distance | 64 |
| Gambar 5.9 Eksekusi Graph Partitioning pada METIS | 69 |

| | |
|--|----|
| Gambar 6.1 Visualisasi Graf Hubungan Produk dan Bahan Makanan..... | 76 |
| Gambar 6.2 Visualisasi Hasil Graph Partitioning..... | 76 |
| Gambar 6.3 Hasil Graph Partitioning - Produk yang Bersertifikat Halal | 77 |
| Gambar 6.4 Visualisasi Graf Produk dan Bahan Makanan Berstatus Haram – 1 | 77 |
| Gambar 6.5 Visualisasi Graf Produk dan Bahan Makanan Berstatus Haram - 2..... | 78 |
| Gambar 6.6 Visualisasi Graf Produk dan Bahan Makanan Berstatus Haram - 3..... | 78 |
| Gambar 6.7 Visualisasi Graf Produk dan Bahan Makanan Ber-MSG – 1..... | 79 |
| Gambar 6.8 Visualisasi Graf Produk dan Bahan Makanan Ber-MSG – 2..... | 79 |
| Gambar 6.9 Visualisasi Graf Produk dan Bahan Makanan Ber-MSG – 3..... | 80 |

Halaman ini sengaja dikosongkan

DAFTAR TABEL

| | |
|--|----|
| Tabel 2.1 Studi Literatur | 5 |
| Tabel 3.1 Class dan Properties Ontology Food LIRMM pada Open Food Fact..... | 17 |
| Tabel 3.2 Penggunaan Ontology pada Halal Nutrition Food . | 17 |
| Tabel 3.3 Contoh Casefolding..... | 20 |
| Tabel 3.4 Karakter yang Diganti Koma..... | 20 |
| Tabel 3.5 Karakter yang Dihilangkan..... | 21 |
| Tabel 3.6 Contoh Penerjemahan dari Bahasa Perancis ke Inggris | 22 |
| Tabel 3.7 Contoh Perubahan Huruf Aksentuasi menjadi Non Aksentuasi | 23 |
| Tabel 3.8 Contoh Stopword Removal | 23 |
| Tabel 3.9 Contoh Tokenizing | 24 |
| Tabel 3.10 Contoh Pengukuran Kesamaan antar Bahan Makanan..... | 25 |
| Tabel 3.11 Contoh Ranking Kemiripan Bahan Makanan | 26 |
| Tabel 4.1 Penggalan Stopword..... | 40 |
| Tabel 4.2 Penggalan CSV Hasil Data Cleansing | 44 |
| Tabel 4.3 Kelompok Produk pada Pencarian Produk Halal MUI | 47 |
| Tabel 4.4 Rancangan Pengujian Jumlah Data Bahan Makanan yang Dihasilkan setelah Pengukuran Kesamaan Bahan Makanan..... | 51 |
| Tabel 4.5 Rancangan Pengujian Fungsionalitas Visualisasi Graf | 51 |
| Tabel 5.1 Spesifikasi Perangkat Keras dalam Pengembangan Aplikasi..... | 53 |
| Tabel 5.2 Spesifikasi Perangkat Lunak dalam Pengembangan Aplikasi..... | 53 |
| Tabel 5.3 Contoh Perbaikan Jumlah Kolom dan Baris sebelum Data Cleansing | 54 |
| Tabel 5.4 Transformasi ke File Impor Tabel Foodproducts... .. | 65 |
| Tabel 5.5 Transformasi ke File Impor Tabel Ingredients | 66 |
| Tabel 5.6 Transformasi ke File Impor Tabel Foodproduct_ingredient | 67 |

| | |
|--|----|
| Tabel 6.1 Hasil Pengujian Jumlah Data yang Dihasilkan setelah Pengukuran Kesamaan Bahan Makanan..... | 75 |
| Tabel 6.2 Hasil Pengujian Fungsionalitas Visualisasi Graf ... | 80 |
| Tabel 6.3 Hasil Graph Partitioning | 83 |
| Tabel 6.4 Hasil Graph Partitioning - Produk yang Tersertifikasi Halal | 86 |

DAFTAR KODE

| | |
|--|----|
| Kode Program 5.1 Pengubahan Huruf Aksentu menjadi Non Aksentu..... | 57 |
| Kode Program 5.2 Deteksi Bahasa Menggunakan Langdetect | 58 |
| Kode Program 5.3 Deteksi Bahasa Menggunakan Pengecekan Dictionary | 59 |
| Kode Program 5.4 Pengukuran Kesamaan Data Bahan Makanan dengan Stopword..... | 61 |
| Kode Program 5.5 Pengukuran Kesamaan Bahan Makanan Menggunakan Levenshtein Distance..... | 63 |
| Kode Program 5.6 Pengukuran Kesamaan Bahan Makanan Menggunakan Jaccard Distance | 64 |
| Kode Program 5.7 Eksekusi Program Pengindeksan..... | 68 |
| Kode Program 5.8 Eksekusi Program untuk Menghasilkan Produk Terkait yang Mirip..... | 68 |
| Kode Program 5.9 Visualisasi Graf..... | 73 |

Halaman ini sengaja dikosongkan

BAB I

PENDAHULUAN

Pada bab pendahuluan akan dijelaskan proses identifikasi masalah penelitian yang meliputi latar belakang masalah, perumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, dan relevansi terhadap pengerjaan tugas akhir. Berdasarkan uraian pada bab ini, harapannya gambaran umum permasalahan dan pemecahan masalah pada tugas akhir dapat dipahami.

1.1. Latar Belakang

Menurut data sensus pada tahun 2010, Indonesia merupakan negara yang memiliki penduduk mayoritas muslim, yaitu sebesar 87, 18%[1]. Sebagai negara dengan penduduk mayoritas muslim, masyarakat Indonesia, mereka, khususnya yang muslim, perlu mengetahui makanan mana yang halal dan yang bukan. Di Indonesia, Lembaga Pengkajian Pangan Obat-obatan dan Kosmetika (LPPOM) MUI adalah lembaga resmi yang mengeluarkan sertifikat halal untuk suatu produk makanan dan juga menyediakan informasi produk makanan apa saja yang bersertifikat halal. Tercatat ada 259.984 produk (pangan, obat dan kosmetika) yang telah tersertifikasi halal pada periode 2012-2017[2]. Namun, informasi tersebut masih sulit diakses oleh pengguna khususnya masyarakat awam.

Halal Nutrition Food adalah situs web yang menyediakan informasi tentang makanan halal beserta kandungan nutrisinya. Halal Nutrition Food memungkinkan pengguna untuk mencari produk makanan halal melalui mesin pencarian dan mengetahui nutrisi berupa fakta nutrisi, komposisi, produsen, sertifikat, dsb. Halal Nutrition Food menyediakan nutrisi produk halal dimana tidak ditampilkan di situs LPPOM MUI. Namun, volume data pada Halal Nutrition Food masih sedikit. Sampai saat ini, ada 315 produk yang tercatat di basis data. Untuk itu, perlu adanya penambahan data agar Halal Nutrition Food lebih lengkap.

Open Food Fact (openfoodfact.org) merupakan open data produk makanan dan nutrisinya dimana memiliki data produk makanan yang beredar di 177 negara[3]. Open Food Fact menggunakan mekanisme crowdsourcing untuk menghimpun data. Hal itu memungkinkan setiap orang dapat memasukkan data makanan sesuai dengan keinginannya, tanpa adanya ketentuan penulisan/format. Oleh karena itu, data Open Food Fact bersifat multi-language, dan dirty. Untuk itu, perlu adanya praproses data yang bertujuan untuk membersihkan data bahan makanan yang masih bersifat kotor.

Halal Nutrition Food dan Open Food Fact menggunakan RDF. Untuk menggabungkan data, perlu dilakukan pengamatan, sejauh manakah perbedaan skema sehingga pengamatan perlu dilakukan sebelum mengintegrasikan keduanya.

Oleh karena Open Food Fact memiliki banyak data, data yang diambil adalah data produk makanan yang memiliki bahan makanan berbahasa perancis dan inggris. Setelah data pada Open Food Fact digabungkan, Halal Nutririon Food memiliki banyak produk beserta bahannya. Untuk mengetahui kelompok produk apa saja yang ada di dalam Halal Nutrition Food, graph partitioning dilakukan dengan mengelompokkan produk dan bahan makanan ke beberapa kluster menggunakan METIS. Melihat di situs Halal Nutrition Food belum memiliki visualisasi, graf akan divisualisasikan secara interaktif untuk memudahkan pengguna dalam memahami insight data bersamaan dengan hasil graph partitioning. Pada akhirnya, situs web Halal Nutrition Food akan memiliki data yang lebih banyak dan dilengkapi visualisasi, bukan hanya mesin pencarian saja.

1.2. Rumusan Masalah

Rumusan masalah yang akan diselesaikan dalam tugas akhir ini adalah:

1. Bagaimana mengintegrasikan data World Open Food Fact ke Halal Nutrition Food?

2. Bagaimana melakukan Graph Partitioning untuk mengelompokkan produk makanan pada Halal Nutrition Food?
3. Bagaimana memvisualisasikan data graf Halal Nutrition Food?

1.3. Batasan Masalah

Batasan masalah terkait pengerjaan tugas akhir ini adalah:

1. Hasil akhir tugas akhir ini berupa visualisasi graf pada Halal Nutrition Food.
2. Data Open Food Fact yang diambil adalah kolom id, nama produk makanan, brand dan bahannya, sedangkan produk yang tidak memiliki bahan makanan tidak diambil.
3. Data produk Open Food Fact yang diambil adalah produk yang memiliki data bahan makanan yang berbahasa perancis dan inggris.
4. *Graph partitioning* dilakukan pada 2 data, yaitu data graf 311.659 produk dan 37.864 bahan makanan, dan data produk yang memiliki sertifikasi halal yang memiliki bahan makanan.
5. Visualisasi graf hubungan produk dan bahan makanan dilakukan pada 50 produk yang memiliki bahan makanan terbanyak.

1.4. Tujuan Penelitian

Tujuan dari penelitian tugas akhir ini adalah:

1. Mengintegrasikan data World Open Food Fact ke Halal Nutrition Food.
2. Melakukan graph partitioning untuk mengelompokkan data graf Halal Nutrition Food.
3. Membuat visualisasi data graf Halal Nutrition Food.

1.5. Manfaat Penelitian

Manfaat yang diharapkan dapat dihasilkan dari pengerjaan tugas akhir ini adalah:

Bagi Peneliti

Peneliti dapat memahami dan menghasilkan penelitian mengenai pengembangan perangkat lunak linked data mengenai produk halal yang mana dapat dikembangkan lagi ke depannya.

Bagi Pengguna

Pengguna dapat memiliki hasil pencarian yang lebih banyak mengenai informasi produk makanan dan lebih mudah memahami data melalui visualisasi graf.

1.6. Relevansi

Tugas akhir ini memiliki relevansi pada laboratorium ADDI (Akuisisi Data dan Diseminasi Informasi) karena memiliki kecocokan dengan bidang keilmuan laboratorium yaitu akuisisi data, paradigma pengolahan dan diseminasi informasi. Selain itu, tugas akhir ini juga menerapkan ilmu yang terdapat pada mata kuliah sistem informasi seperti Penggalan Data dan Analisa Bisnis dan Pemrograman Integratif.

Tugas akhir ini memecahkan masalah, yaitu menambah volume data pada Halal Nutrition Food dengan mengintegrasikannya dengan Open Food Fact dan mengetahui kelompok produk makanan pada Halal Nutrition Food dengan menerapkan graph partitioning dan memberikan visualisasi graf terhadap pengguna. Oleh karena itu, tugas akhir ini layak dijadikan tugas akhir pada tingkat S1.

BAB II TINJAUAN PUSTAKA

Pada bab ini akan dijelaskan mengenai penelitian sebelumnya dan dasar teori yang dijadikan acuan atau landasan dalam pengerjaan tugas akhir ini. Landasan teori akan memberikan gambaran secara umum dari landasan penjabaran tugas akhir ini.

2.1. Penelitian Sebelumnya

Tabel 2.1 berisi penelitian-penelitian sebelumnya yang dijadikan acuan dalam pengerjaan tugas akhir.

Tabel 2.1 Studi Literatur

| Nama Peneliti | Judul Penelitian (Tahun) | Kesimpulan | Relevansi |
|---------------|---|---|---|
| Hao Chen | String Metrics and Word Similarity Applied to Information Retrieval (2012)[4] | Penelitian ini membahas string metric (Edit Distance, Jaccard Distance, Q-gram, Cosine Similarity dan Dice Coefficient) dan word similarity menggunakan WordNet (Path-length based dan Information-content based similarity). Beberapa metode tersebut diterapkan untuk membandingkan beberapa kata sehingga dapat dilakukan perbandingan performa antar metode. Hasilnya, levensthein adalah string metric yang lebih baik dibanding yang lainnya. | Sebagai acuan untuk mengukur kesamaan kata menggunakan edit distance/levensthein distance dan jaccard distance guna melakukan preprocessing data. |

| Nama Peneliti | Judul Penelitian (Tahun) | Kesimpulan | Relevansi |
|----------------------|--|--|---|
| Ahmad Choirun Najib | Rancang Bangun Aplikasi Android Halal Nutrition Food Menggunakan Kombinasi Query Independent Dan Query Dependent Ranking (2018)[5] | Penelitian ini membahas mengenai indexing dan searching menggunakan Apache Lucene pada aplikasi Halal Nutrition Food. | Sebagai acuan untuk integrasi data dengan aplikasi Halal Nutrition Food |
| Azmi Adi Firman syah | Pengembangan Pencarian Produk Terkait Menggunakan Euclidean Distance dan Cosine Similarity pada Aplikasi Halal Nutrition Food(2018)[6] | Penelitian ini membahas mengenai pencarian produk terkait berdasarkan bahan makanan menggunakan Euclidean Distance dan Cosine Similarity pada aplikasi Halal Nutrition Food. | Sebagai acuan untuk integrasi data dengan aplikasi Halal Nutrition Food |
| Youp Uylings | Partitioning of Big Graphs (2015)[7] | Penelitian ini membahas mengenai graph partitioning pada graf yang berukuran besar menggunakan algoritma METIS (recursive and k-way), node dan edge partitioning. Hasilnya | Sebagai acuan untuk melakukan graph partitioning menggunakan METIS |

| Nama Peneliti | Judul Penelitian (Tahun) | Kesimpulan | Relevansi |
|---|--|---|--|
| | | menunjukkan bahwa METIS merupakan algoritma yang paling efisien. | graph partitioning |
| Raga'ad M. Tarawneh, Patricia Keller, dan Achim Ebert | A General Introduction to Graph Visualization Techniques (2011)[8] | Penelitian ini membahas macam graph layout dan penerapannya. Selain itu, penelitian ini juga membahas teknik interaksi pada visualisasi graf yang baik. | Sebagai acuan layout dan teknik visualisasi interaktif graf. |

2.2. Dasar Teori

Berikut ini dijabarkan dasar-dasar teori yang digunakan dalam pengerjaan tugas akhir ini.

2.2.1. Halal Nutrition Food

Pada tahun 2015, Irfan Rizki Ananda mengembangkan aplikasi web linked open data produk makanan halal. Selanjutnya, pada tahun 2016, Jauhar Fatawi mengembangkan perangkat lunak linked open data halal dan gizi pada makanan dan minuman yang selanjutnya disebut Halal Nutrition Food. Halal Nutrition Food adalah situs web yang menyediakan informasi tentang makanan halal beserta kandungan nutrisinya[9]. Pada tahun 2017, Adnan Mauludin Fajriyadi mengembangkan sistem pencarian dalam aplikasi tersebut menggunakan algoritma algoritma OKAPI BM25F. Halal Nutrition Food memungkinkan pengguna untuk mencari produk makanan halal melalui mesin pencarian dan mengetahui nutrisi berupa fakta nutrisi, komposisi, produsen, sertifikat, dsb. Halal Nutrition Food menyediakan nutrisi produk halal dimana tidak ditampilkan di situs LPPOM MUI. Pada tahun 2018, Azmi Adi Firmansyah mengembangkan sistem pencarian produk terkait menggunakan Euclidean Distance dan Cosine Similarity dan

Ahmad Choirun Najib mengembangkan aplikasi android menggunakan Query Independent dan Query Dependent Ranking.

Data Halal Nutrition Food didapat dari crowdsourcing, linked data dan data webscrapping dari situs LPPOM MUI. Saat ini, ada 324 produk makanan yang tercatat di basis data Halal Nutrition Food.

2.2.2. Open Food Fact

Open Food Fact (openfoodfact.org) adalah sebuah basis data yang kolaboratif, gratis, dan terbuka dimana berisi produk makanan di seluruh dunia dengan bahan, alergen, fakta nutrisi dan semua informasi yang dapat kita temukan di label produk[10]. Data tersebut diperoleh dari mekanisme crowdsourcing, dimana pengguna memasukkan data produk makanan baik melalui situs web maupun aplikasi mobile. Sampai saat ini, terdapat 320.000++ produk yang sudah ada di dalam database dan 6.000++ pengguna yang sukarela memasukkan data produk makanan[3]. Karena menggunakan crowdsourcing, maka setiap orang bebas memasukkan data tanpa mempedulikan ketentuan mengenai data yang dimasukkan. Oleh karena itu, data yang ada masih bersifat kotor dengan kata lain perlu dilakukan preprocessing data untuk membersihkan dan layak diproses.

Data Open Food Fact didapatkan dari MongoDB dump yang diunduh dari situs openfoodfact.org. Data tersebut memiliki format bson, sebuah representasi dari file backup MongoDB. Di dalamnya terdapat 1 collection, yaitu products yang di dalamnya terdapat 442.467 document atau disebut juga memiliki jumlah produk sebesar tersebut. Setiap document memiliki jumlah field (kolom) yang berbeda-beda, yang terbesar, yaitu 166 field. Selain MongoDB dump, data Open Food Fact juga tersedia dalam format RDF dan CSV yang diambil dari MongoDB Dump tersebut.

2.2.3. RDF

Resource Description Framework (RDF) adalah kerangka untuk menerangkan 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 ditampilkan kepada orang. RDF menyediakan framework umum untuk menginformasikan data sehingga dapat dilakukan pertukaran data antar aplikasi tanpa kehilangan makna[11].

RDF didasarkan pada pembuatan model berdasarkan pernyataan tentang sumber daya / resources (pada web) ke dalam bentuk subject-predicate-obyek. Bentuk ini dikenal dengan nama triples. Subyek dan obyek menunjukkan sumber daya, predikat menunjukkan ciri-ciri atau aspek sumber daya dan menghubungkan antara subyek dan obyek.

| | | |
|----------|------------|---------|
| Beri | Makan | Nasi |
| (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[11]. RDF triples memiliki dua tipe, sebagai berikut:

- Literal Triples, merupakan triples dengan RDF literal berupa string, angka, atau tanggal sebagai objek. Literal triples digunakan untuk mendeskripsikan sifat / properti dari suatu hal / data.
- RDF Links, merepresentasikan hubungan antara dua sumber data. RDF links terdiri dari tiga referensi URI. URI yang digunakan pada subjek dan objek untuk mengidentifikasi sumber data yang saling terkait, serta

URI pada predikat untuk mendefinisikan keterkaitan antar data.

Pada linked data dikenal istilah ontology/vocabulary, yaitu sebuah model yang menjelaskan hubungan antar sumber daya (subyek dan obyek) yang ada. Perbedaannya, vocabulary memiliki aturan yang tidak terlalu ketat. Pada Halal Nutrition Food, RDF akan direpresentasikan dalam bentuk file Turtle (.ttl), sebuah file yang memiliki aturan penulisan lebih sederhana dan pendek daripada N-Triples (.nt) atau RDF file (.rdf).

2.2.4. Levenshtein Distance

Levenshtein distance (Edit distance) adalah jumlah minimum operasi edit yang diperlukan untuk mengubah S menjadi T dengan operasi penambahan, penghapusan, atau penggantian satu karakter tunggal[12]. Semakin kecil nilai Levenshtein Distance berarti semakin mirip kedua string yang dibandingkan. Sebagai contoh, kita dapat menghitung jarak antara dua string, "apple" dan "orange" dengan jarak Levenshtein. Nilai jarak adalah 5. Kata "apple" dapat diubah menjadi "orange" dengan:

1. penyisipan "O",
2. penggantian "A" menjadi "R",
3. penggantian "P" menjadi "A",
4. penggantian "P" menjadi "N", dan
5. penggantian "L" menjadi "G".

| | | A | P | P | L | E |
|---|---|---|---|---|---|---|
| | 0 | 1 | 2 | 3 | 4 | 5 |
| O | 1 | 1 | 2 | 3 | 4 | 5 |
| R | 2 | 2 | 2 | 3 | 4 | 5 |
| A | 3 | 2 | 3 | 3 | 4 | 5 |
| N | 4 | 3 | 3 | 4 | 4 | 5 |
| G | 5 | 4 | 4 | 4 | 5 | 5 |
| E | 6 | 5 | 5 | 5 | 5 | 5 |

Gambar 2.1 Contoh Levenshtein Distance

2.2.5. Jaccard Distance

Jaccard similarity adalah alat statistik yang digunakan untuk membandingkan kesamaan dan keragaman sampel. Koefisien Jaccard mengukur kesamaan antara kumpulan sampel yang terbatas, dan didefinisikan sebagai ukuran irisan dibagi dengan ukuran gabungan kumpulan sampel[13]. Untuk mengukur kesamaan dua kata, maka persamaan Jaccard similarity dapat didefinisikan sebagai jumlah karakter/huruf yang merupakan irisan dibagi dengan jumlah karakter/huruf yang merupakan gabungan dari kedua kata tersebut. Sedangkan Jaccard Distance didefinisikan sebagai 1 dikurangi Jaccard Similarity. Semakin kecil nilai Jaccard Distance berarti semakin mirip kedua string yang dibandingkan.

$$JD (W_1, W_2) = 1 - \frac{|W_1 \cap W_2|}{|W_1 \cup W_2|} \quad 2.1$$

Keterangan :

$JD (W_1, W_2) = Jaccard\ Distance\ Kata\ 1\ dan\ Kata\ 2$

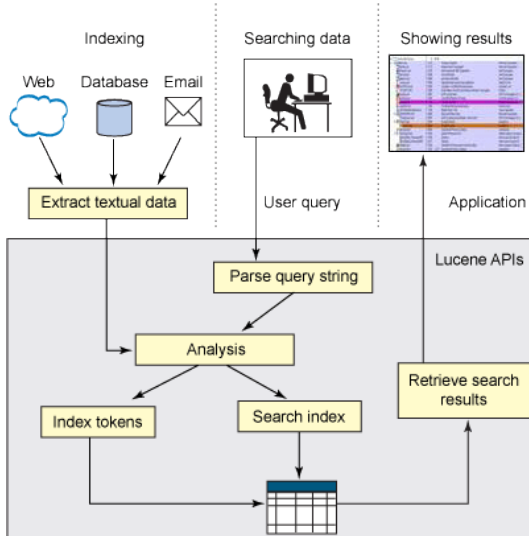
$W_1 = Kata\ 1$

$W_2 = Kata\ 2$

2.2.6. Apache Lucene

Apache Lucene merupakan library mesin pencari teks dengan fitur lengkap dan performa tinggi yang dituliskan menggunakan bahasa Java. Teknologi ini cocok untuk semua aplikasi yang membutuhkan fitur full-text search, terutama untuk aplikasi cross platform. Hal tersebut dilakukan dengan menambahkan konten dari dokumen ke dalam indeks full-text. Hal Ini untuk dilakukan kueri pada indeks, dan mengembalikan hasil berdasarkan relevansi dengan kueri atau diurutkan menurut atribut lainnya seperti tanggal modifikasi terakhir pada dokumen. Konten yang ditambahkan ke Lucene bisa berasal dari berbagai sumber, seperti database SQL/NoSQL, file sistem, atau bahkan dari situs web. Lucene mampu mendapatkan hasil pencarian yang cepat karena melakukan pencarian terhadap indeks daripada mencari teks secara langsung. Indeks ini disebut indeks terbalik (inverted index),

karena membalikkan struktur data yang berpusat pada halaman menjadi struktur data yang berpusat pada kata kunci[14]. Proses yang terjadi pada Apache Lucene akan dijelaskan pada gambar 2.2[15].



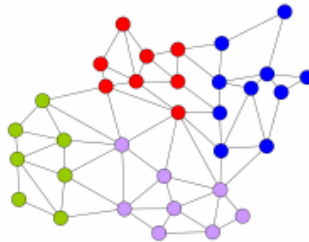
Gambar 2.2 Proses pada Apache Lucene

2.2.7. Graph Partitioning

Graf adalah struktur yang terdiri dari sekumpulan objek (node atau vertice) di mana beberapa pasang benda terkait oleh garis (edge atau line). Untuk menyimpan informasi, pada graph database, setiap node merepresentasikan entitas (seperti benda atau orang) dan edge merepresentasikan hubungan antara dua node tersebut. Setiap node memiliki properties atau informasi yang berkaitan erat dengan node[16].

Graph partitioning adalah istilah yang ditujukan pada pembagian node pada graf dengan mengurangi edge sehingga graf yang besar menjadi sejumlah blok graf yang lebih kecil untuk memudahkan penanganan. Graph partitioning dikatakan bagus apabila setiap blok graf memiliki ukuran yang sama dan

jumlah edge yang menghubungkan antar komponen berjumlah sedikit[17].

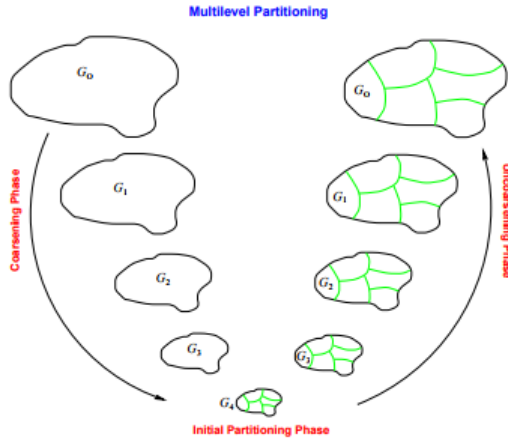


Gambar 2.3 Ilustrasi Graph Partitioning

Graph partitioning juga bisa digunakan untuk analisis kluster dengan mengelompokkan entitas yang memiliki hubungan paling erat ke dalam satu blok graf sehingga satu graf besar akan menjadi beberapa graf kecil yang juga menggambarkan beberapa kluster.

Salah satu alat yang digunakan untuk melakukan graph partitioning adalah METIS. METIS adalah seperangkat program untuk graph partitioning dan menghasilkan urutan pengisian-pengurangan untuk sparse matrices, yang dikembangkan oleh Karypis Lab, University of Minnesota. Algoritma yang diimplementasikan dalam METIS didasarkan pada skema multilevel recursive-bisection, multilevel k-way, dan multi-constraint partitioning. Keunggulan METIS, yaitu menyediakan graph partitioning yang bagus dan lebih cepat[18].

Ada 3 tahap yang dilakukan oleh METIS, seperti yang terlihat pada Gambar 2.4. Pertama, coarsening, graf awal dipisah menjadi ratusan graf yang lebih kecil dengan memotong edge. Kedua, partitioning, menghitung bisection graph yang lebih kecil tersebut dengan algoritma Kernighan-Lin. Ketiga, uncoarsening, graf tersebut diproyeksikan ke graf yang lebih besar[18].



Gambar 2.4 Multilevel Graph Partitioning

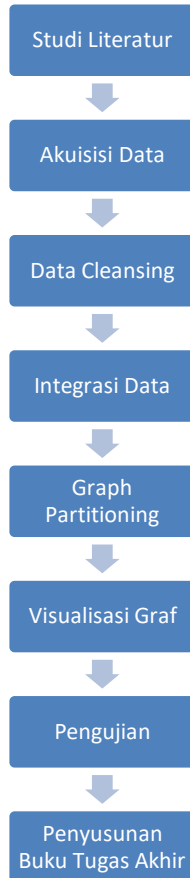
2.2.8. D3js

D3.js adalah library JavaScript untuk memanipulasi dokumen berdasarkan data. D3 digunakan untuk menampilkan data menggunakan HTML, SVG, dan CSS. Penekanan D3 pada standar web memberi kemampuan penuh browser modern, menggabungkan komponen visualisasi yang kuat dan pendekatan berbasis data untuk manipulasi DOM[19]. Dengan library networkD3-nya, D3js dapat digunakan untuk memvisualisasikan network atau graf dengan indah dan interaktif sehingga lebih menarik.

BAB III

METODOLOGI PENELITIAN

Pada bab metode penelitian akan diurutkan tahapan-tahapan apa saja yang dilakukan dalam menyelesaikan penelitian tugas akhir ini. Masing-masing tahapan akan dijelaskan secara rinci.



Gambar 3.1 Tahapan Pengerjaan Tugas Akhir

Berikut merupakan penjelasan dari setiap tahapan yang ada pada metodologi yang digunakan.

3.1. Studi Literatur

Studi literatur dilakukan dengan cara mengumpulkan berbagai buku dan penelitian sebelumnya yang berhubungan dengan dataset, data cleansing, Levensthein distance, Jaccard distance, integrasi data, graph partitioning dan visualisasi graf.

3.2. Akuisisi Data

Data yang akan melalui cleansing diambil dari MongoDB dump Open Food Fact, collection products, kolom id, code, brands, product_name, ingredients_text_en dan ingredients_text_fr yang berisi id, barcode, brand, nama produk dan bahan makanan.

3.2.1. Pengamatan Skema RDF pada Open Food Fact dan Halal Nutrition Food

Untuk menggabungkan data RDF, pengamatan mengenai ontology/vocabulary yang digunakan oleh kedua data perlu dilakukan. RDF pada Open Food Fact yang menjelaskan produk dan bahannya dapat dilihat di situs <http://world.openfoodfacts.org/data>. Ontology/vocabulary yang digunakan oleh Open Food Fact untuk menggambarkan domain makanan dapat dilihat pada Gambar 3.2. Contoh RDF pada Open Food Fact dapat dilihat pada Lampiran B.

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:food="http://data.lirmm.fr/ontologies/food#"
  xmlns:dcterms="http://purl.org/dc/terms/"
  xmlns:dc="http://purl.org/dc/elements/1.1/"
  xmlns:void="http://rdfs.org/ns/void#"
  xmlns:owl="http://www.w3.org/2002/07/owl#"
  xmlns:foaf="http://xmlns.com/foaf/0.1/">
```

Gambar 3.2 Ontology yang Digunakan pada Open Food Fact

LIRMM food ontology atau <http://data.lirmm.fr/ontologies/food#> merupakan sebuah ontology untuk memodelkan domain makanan yang dikembangkan oleh LIRMM Perancis. Penggunaan ontology

LIRMM food ontology pada Open Food Fact dijelaskan pada Tabel 3.1.

Tabel 3.1 Class dan Properties Ontology Food LIRMM pada Open Food Fact

| Class | Properties | Keterangan Properties |
|-------------------|---|--|
| food:Food | food:ingredientList AsText | Teks daftar bahan makanan |
| | food:containsIngredient | Menentukan bahan yang terkandung dalam objek makanan - dapat menentukan kuantitas, satuan, dan sifat bahan |
| | food:carbohydratesPer100g ; food:sodiumPer100g ; food:proteinsPer100g ; dsb | Menerangkan kandungan nutrisi per 100g makanan |
| | food:name | Nama makanan |
| food:Food Product | food:code | Barcode |
| food:Ingredient | food:food | Makanan yang merupakan bahan dari makanan lain |
| | food:rank | Peringkat bahan dalam suatu daftar bahan menurut kuantitasnya |
| | food:percent | Jumlah bahan dalam persentase dari total jumlah bahan dalam makanan |

Penggunaan ontology/vocabulary yang digunakan pada Halal Nutrition Food dapat dilihat pada Tabel 3.2.

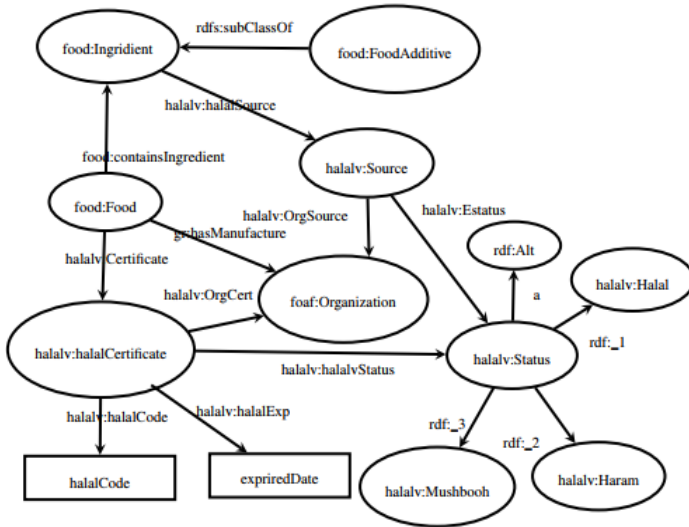
Tabel 3.2 Penggunaan Ontology pada Halal Nutrition Food

| Ontology | Class | Properties | Keterangan Properties |
|----------|-----------|-------------------------|--------------------------------|
| | food:Food | food:containsIngredient | Menerangkan bahan dari makanan |

| Ontology | Class | Properties | Keterangan Properties |
|---|-------------------|--|---|
| http://purl.org/foodontology# | food:foodAdditive | | |
| http://purl.org/goodrelations/v1# | | gr:hasManufacturer | Menerangkan memiliki manufaktur/diproduksi oleh |
| http://data.lirmm.fr/ontologies/food# | food:Food | foodlirmm:fatPer100g ; foodlirmm:saturatedFatPer100g ; foodlirmm:sodiumPer100g ; dsb. | Menerangkan kandungan nutrisi tersebut tiap 100 gram |
| | food:Ingredient | foodlirmm:rank | Peringkat bahan dalam suatu daftar bahan menurut kuantitasnya |
| | food:FoodProduct | foodlirmm:code | Barcode produk makanan |
| http://xmlns.com/foaf/0.1/ | foaf:Organization | | |

Adapun vocabulary yang dibuat pada Halal Nutrition Food. Halalv (<http://halal.addi.is.its.ac.id/halalv.ttl#>) merupakan *vocabulary* yang dibuat sendiri pada Halal Nutrition Food. Pada Halal Nutrition Food, setiap produk, bahan, manufaktur, sumber halal dan sertifikatnya disimpan dalam file Turtle. Halalv vocabulary dijelaskan pada Gambar 3.3. Sedangkan ontology yang digunakan pada Halal Nutrition Food dijelaskan

pada Gambar 3.4. Untuk file turtle, selengkapnya dapat dilihat pada Lampiran B.



Gambar 3.3 Vocabulary Halal Nutrition Food

```

@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .↓
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .↓
@prefix owl: <http://www.w3.org/2002/07/owl#> .↓
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .↓
@prefix foaf: <http://xmlns.com/foaf/0.1/> .↓
@prefix halalv: <http://halal.addi.is.its.ac.id/halalv.ttl#> .↓
@prefix halalv: <http://halal.addi.is.its.ac.id/resources/foodproducts/> .↓
@prefix halalv: <http://halal.addi.is.its.ac.id/resources/ingredients/> .↓
@prefix halalv: <http://halal.addi.is.its.ac.id/resources/halalsources/> .↓
@prefix halalv: <http://halal.addi.is.its.ac.id/resources/certificates/> .↓
@prefix halalv: <http://halal.addi.is.its.ac.id/resources/manufactures/> .↓
@prefix food: <http://purl.org/foodontology#> .↓
@prefix foodlirmm: <http://data.lirmm.fr/ontologies/food#> .↓
@prefix gr: <http://purl.org/goodrelations/v1#> .↓
  
```

Gambar 3.4 Ontology/Vocabulary pada Halal Nutrition Food

3.3. Data Cleansing

Data cleansing bertujuan untuk membersihkan data yang masih bersifat kotor akibat mekanisme *crowdsourcing* yang ada pada

Open Food Fact dan mengubah data bahan makanan yang masih berbahasa perancis ke bahasa inggris.

3.3.1. Casefolding

Casefolding dilakukan untuk mengubah bahan makanan menjadi huruf kecil (*lowercase*). Contoh dapat dilihat pada Tabel 3.3.

Tabel 3.3 Contoh Casefolding

| Contoh | |
|--|--|
| Sebelum | Sesudah |
| Carbonated Water, Sugar, Colour ,Caramel E150d,, Phosphoric Acid, Preservative ,Potassium sorbate,, Flavourings, Caffeine, Sweeteners ,Aspartame, acesulfame K,. | carbonated water, sugar, colour ,caramel e150d,, phosphoric acid, preservative ,potassium sorbate,, flavourings, caffeine, sweeteners ,aspartame, acesulfame k,. |

3.3.2. Penghilangan/Penggantian Karakter yang Tidak Diperlukan

Pada bahan makan terdapat simbol/karakter tertentu yang tidak diperlukan sehingga perlu dihapus. Karakter yang perlu dihapus/diganti serta contohnya dapat dilihat pada Tabel 3.4 dan Tabel 3.5.

Tabel 3.4 Karakter yang Diganti Koma

| Karakter | Contoh | |
|----------|--|--|
| | Sebelum | Sesudah |
| [] | sirop de sucre mélassé [contient des sulfites] | sirop de sucre mélassé ,contient des sulfites, |
| () | sirop de sucre mélassé (contient des sulfites) | sirop de sucre mélassé, contient des sulfites, |
| { } | sirop de sucre mélassé {contient des sulfites} | sirop de sucre mélassé {contient des sulfites} |
| · | rice · wheatflakes | rice ,wheatflakes |
| : | colorants: lutéine | colorants, lutéine |

| Karakter | Contoh | |
|----------|------------------------------------|-----------------------------------|
| | Sebelum | Sesudah |
| & | citric acid & sodium benzoate | citric acid,sodium benzoate |
| • | huile de tournesol • farine de blé | huile de tournesol, farine de blé |
| " | ent appelée "Noitéell | ent appelée,Noitéell |
| < | Steaks haches bio <5% | Steaks haches bio,5% |
| @ | arôme naturel. @mango sorbetijs | arôme naturel,mango sorbetijs |
| ? | sea salt? raising agent | sea salt, raising agent |
| = | vetten= dont acides ras satures | vetten, dont acides ras satures |
| - | eau - farine de blé | eau, farine de blé |
| „ | potassium sorbate,, flavourings | potassium sorbate, flavourings |

Tabel 3.5 Karakter yang Dihilangkan

| Karakter | Contoh | | Regular Expression |
|-----------------------|-----------------------------|--------------------|--------------------|
| | Sebelum | Sesudah | |
| “ | “sirop | sirop | |
| _ | _sulfites_ | sulfites | |
| * | soja* | soja | |
| # | # cacao | cacao | |
| (angka) % ...,% | 8,2% fett | fett | \d[,]\d[%] |
| %(angka)) | %44.5 marshmallow | marshmallow | [%]\d\d[.]\d |
| (angka)g | sel 1,2 g | sel | \d[,]\d g |
| (angka) kcal | valeur énergétique 454 kcal | valeur énergétique | \d\d\d kcal |

| Karakter | Contoh | | Regular Expression |
|------------------------------|--------------------|-------------|---|
| | Sebelum | Sesudah | |
| (angka)kj | pour 100 ml 105 kj | pour 100 ml | $\backslash d \backslash d \backslash d \text{ kj}$ |
| (angka)ml | 100 ml | pour | $\backslash d \backslash d \backslash d \text{ ml}$ |
| (angka)(satuan) | 120mg | | $[\backslash d]+[a-z]^*$ |
| (angka)(kata)(angka) | 12x5 | | $[\backslash d]+[a-z]^*+[\backslash d]+$ |
| (angka)(kata)(angka)(satuan) | 12x12ml | | $[\backslash d]+[a-z]^*+[\backslash d]+[a-z]^+$ |

3.3.3. Penerjemahan Bahan Makanan yang Berbahasa Perancis ke Bahasa Inggris

Halal Nutrition Food memiliki data berbahasa inggris. Untuk itu, data bahan makanan pada Open Food Fact yang masih berbahasa perancis perlu diterjemahkan ke bahasa inggris. Penerjemahan dilakukan dengan SDL Free Translation Tools[20]. Contoh penerjemahan dapat dilihat pada Tabel 3.6.

Tabel 3.6 Contoh Penerjemahan dari Bahasa Perancis ke Inggris

| Contoh | |
|---|---|
| Bahan Makanan (FR) | Bahan Makanan (EN) |
| sirop de glucose,eau,caramel,arôme naturel de vanille bourbon | glucose syrup, water, caramel, natural aroma of bourbon vanilla |

3.3.4. Pengubahan Huruf Aksentu ke Non-Aksentu

Ada produk dan bahan makanan yang memiliki detail pengucapan tertentu yang diungkapkan melalui huruf aksentu. Untuk memudahkan pada saat integrasi, nama produk dan bahan makanan yang memiliki huruf aksentu perlu diubah

menjadi huruf non-aksen. Contoh pengubahan huruf aksen dapat dilihat pada Tabel 3.7.

Tabel 3.7 Contoh Pengubahan Huruf Aksen menjadi Non Aksen

| Contoh | |
|------------------------------|------------------------------------|
| Bahan Makanan Berhuruf Aksen | Bahan Makanan Tidak Berhuruf Aksen |
| glacé cherries | glace cherries |
| banana purée | banana puree |
| pâte of cocoa | pate of cocoa |

3.3.5. Penghilangan Bahan Makanan Berbahasa Non Inggris yang Masih Tersisa

Meskipun data bahan makanan sudah diambil dari kolom bahan makanan berbahasa perancis dan inggris, ternyata masih terdapat bahan makanan yang berbahasa non inggris seperti italia, jerman, spanyol, swedia dan belanda. Untuk itu, perlu adanya pendeteksian bahasa untuk mengetahui data bahan makanan yang berbahasa selain bahasa inggris dan perancis.

Bahan makanan melalui 2 deteksi, yaitu menggunakan library langdetect[21] dan pengecekan pada *dictionary* bahasa inggris[22] maupun perancis[23]. Setelah terdeteksi, data makanan yang berbahasa non inggris dapat dihilangkan dan yang berbahasa perancis dapat diterjemahkan ke bahasa inggris.

3.3.6. Stopword Removal

Stopword berisi kata-kata selain bahan makanan, yang mana tidak diperlukan. Setelah daftar stopwords dibuat, maka stopwords yang masih ada pada bahan makanan tersebut perlu dihapus. Contoh stopwords removal dapat dilihat pada Tabel 3.8.

Tabel 3.8 Contoh Stopword Removal

| Contoh | |
|---|-------------------------------|
| Sebelum | Sesudah |
| ,extra virgin olive oil,salt,ingredient from organic farming, | ,extra virgin olive oil,salt, |

3.3.7. Tokenizing

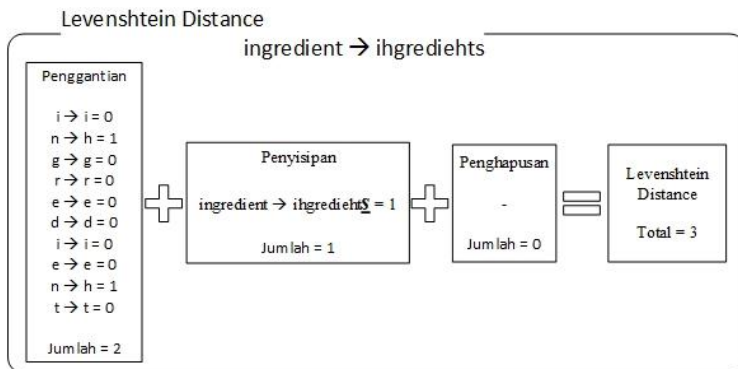
Untuk memisahkan setiap bahan makanan menjadi tiap bahan makanan yang unik, maka perlu dilakukan tokenizing berdasarkan koma (,) karena setiap bahan makanan pada tiap produk makanan terpisahkan oleh koma. Contoh tokenizing dapat dilihat pada Tabel 3.9.

Tabel 3.9 Contoh Tokenizing

| Contoh | |
|---------------------------|--|
| Sebelum | Sesudah |
| petits pois,eau,sucre,sel | - petits pois - eau - sucre - sel |

3.3.8. Penghilangan Data Kotor yang Mirip Stopword

Setelah dilakukan stopwords removal, ada kata-kata selain bahan makanan yang masih belum terhapus. Padahal kata-kata tersebut merupakan kata-kata stopwords yang mengalami salah penulisan. Untuk itu, harus dilakukan pengukuran kesamaan antara stopwords dan data bahan makanan menggunakan Levenshtein Distance. Data yang mirip dengan stopwords maka perlu dihapus. Contoh pengukuran kesamaan menggunakan Levenshtein distance dapat dilihat pada Gambar 3.5.



Gambar 3.5 Contoh Pengukuran Kesamaan Menggunakan Levenshtein Distance

Pada contoh pada Gambar 3.5, untuk merubah kata “ingredient” menjadi “ihgrediechts”, dibutuhkan:

- Penggantian n menjadi h,
- Penggantian n menjadi h,
- Penyisipan s,
- Tidak ada penghapusan,

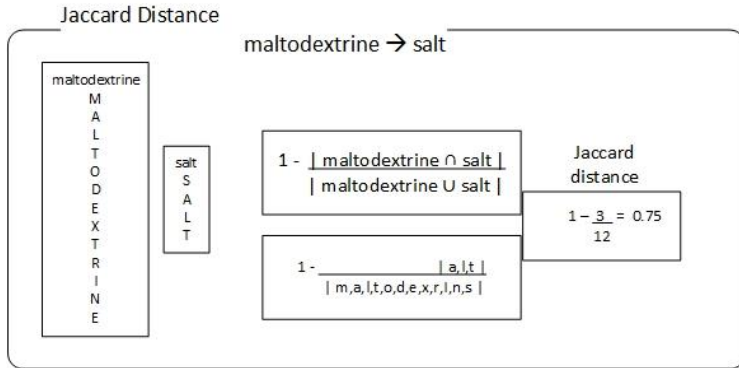
sehingga nilai Levenshtein distance adalah 3 karena memiliki 3 perubahan.

3.3.9. Pengukuran Kesamaan pada Bahan Makanan

Untuk memperbaiki kesalahan penulisan, bahan makanan perlu diukur kesamaannya menggunakan Levenshtein distance dan Jaccard distance. Bahan makanan yang memiliki kesamaan tinggi akan diperbaiki penulisannya. Contoh: Eggs dan egg memiliki levenshtein distance 1, yaitu penambahan huruf, maka kedua kata tersebut terindikasi sama sehingga kata eggs diganti dengan egg. Contoh pengukuran kesamaan dapat dilihat pada Tabel 3.10. Contoh pengukuran kesamaan menggunakan Levenshtein distance dapat dilihat pada Gambar 3.5 sedangkan Jaccard distance dapat dilihat pada Gambar 3.6.

Tabel 3.10 Contoh Pengukuran Kesamaan antar Bahan Makanan

| Bahan makanan yang diukur | Diukur kesamaannya dengan | Levenshtein distance | Jaccard distance |
|---------------------------|---------------------------|----------------------|------------------|
| egg | maltodextrine | 12 | 0.9166 |
| | eggs | 1 | 0.3333 |
| | salt | 4 | 1 |
| | milk | 4 | 1 |
| | sugar | 4 | 0.8333 |
| maltodextrine | maltodextrine | 1 | 0.1666 |
| | eggs | 11 | 0.9230 |
| | salt | 10 | 0.8461 |
| | milk | 10 | 0.8461 |
| | sugar | 11 | 0.8571 |



Gambar 3.6 Contoh Pengukuran Kesamaan Menggunakan Jaccard Distance

Setelah diketahui hasil pengukurannya, bahan makanan akan diurutkan mana yang paling mirip berdasarkan nilai kesamaannya. Untuk menghindari penggantian bahan makanan yang tidak mirip, pemilihan akan dilakukan, apakah memang bahan makanan yang diukur memiliki kesamaan dan mengalami salah penulisan. Bahan makanan yang mengalami salah penulisan perlu diganti dengan bahan makanan terkait. Contoh ranking/urutan kemiripan bahan makanan dapat dilihat pada Tabel 3.11.

Tabel 3.11 Contoh Ranking Kemiripan Bahan Makanan

| Bahan makanan yang diukur | Ranking | Bahan makanan yang paling mirip (levensthein) | Bahan makanan yang paling mirip (jaccard) |
|---------------------------|---------|---|---|
| egg | 1 | eggs | eggs |
| | 2 | salt | maltodextrine |
| | 3 | milk | salt |
| | 4 | sugar | milk |
| | 5 | maltodextrine | sugar |
| maltodextrine | 1 | maltodextrine | maltodextrine |
| | 2 | salt | salt |
| | 3 | Milk | milk |
| | 4 | eggs | sugar |
| | 5 | sugar | eggs |

Untuk contoh, bahan makanan “eggs” memiliki kesamaan yang paling mirip dengan “egg”. Oleh karena itu, penulisan bahan makanan “eggs” diganti menjadi “egg”. Begitu juga bahan makanan “ma\todextrine” dan “maltodextrine”.

3.3.10. Supervisi Manual terhadap Bahan Makanan

Setelah bahan makanan melalui pengukuran kesamaan, data bahan makanan diperiksa secara manual apakah data bahan makanan tersebut memang sebuah bahan makanan dan sudah benar penulisannya. Jika bahan makanan tersebut masih salah penulisannya maka diperbaiki dan jika bahan makanan tersebut tidak benar-benar sebuah bahan makanan maka dihapus. Oleh karena hal ini dilakukan secara manual, tidak semua bahan makanan telah melalui pemeriksaan.

3.4. Integrasi Data

Kedua data memakai ontology LIRMM Food Ontology. Properties yang sama-sama digunakan oleh kedua data, yaitu :rank, :code, :fatPer100g, :saturatedFatPer100g, :sodiumPer100g dan properties lain yang menerangkan kandungan nutrisi per 100 gram. Skema RDF pada Open Food Fact dan Halal Nutrition Food memiliki kesamaan sehingga dalam integrasi, skema RDF Halal Nutrition Food dapat dipakai untuk menuliskan file turtle yang nantinya digunakan sebagai masukan saat pengindeksan.

Setelah mengamati RDF yang ada di Open Food Fact dan Halal Nutrition Food; dan melakukan *data cleansing*, hasil data cleansing akan diimpor ke basis data Halal Nutrition Food. Setelah diimpor ke basis data, data nama produk makanan dan bahannya akan dihasilkan menjadi file turtle dan siap diintegrasikan dengan melakukan indeks menggunakan Apache Lucene. Pengindeksan dilakukan dengan menjalankan kode program yang dihasilkan pada penelitian sebelumnya oleh Ahmad Choirun Najib[5].

Independent ranking juga dilakukan untuk mengetahui apakah bahan makanan dari suatu produk memiliki tautan di DBpedia atau tidak. Independent ranking dihasilkan dari kode program

yang telah dibuat pada penelitian sebelumnya oleh Ahmad Choirun Najib[5].

Daftar produk terkait juga dapat dihasilkan untuk memberikan produk yang mirip dengan produk yang sedang dicari. Produk terkait dihasilkan dari kode program yang dibuat pada penelitian Halal Nutrition Food sebelumnya oleh Azmi Adi Firmansyah[6].

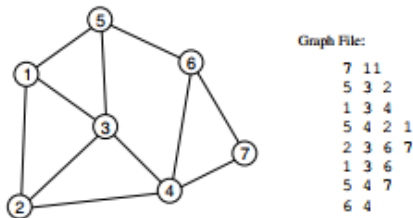
3.5. Graph Partitioning

3.5.1. Mengubah Data Graph ke Format Masukan METIS

Berikut karakteristik file masukan dari METIS.

- Graf dengan N node disimpan dalam file .graph, N+1 baris.
- Baris pertama mencantumkan jumlah node dan jumlah edge.
- Setiap baris berikutnya mencantumkan "tetangga" sebuah node.

Contoh data graf dan file masukan METIS dapat dilihat pada Gambar 3.7.



Gambar 3.7 Contoh File Input METIS

Baris pertama diisi jumlah *node* dan jumlah *edge*, yaitu 7 *node* dan 11 *edge*. Baris kedua dan seterusnya diisi *node* dan tetangganya. *Node* 1 terhubung ke *node* 5,3,2. *Node* 2 terhubung ke *node* 1,3,4. *Node* 3 terhubung ke *node* 5,4,2,1. *Node* 4 terhubung ke *node* 2,3,6,7. *Node* 5 terhubung ke *node* 1,3,6. *Node* 6 terhubung ke *node* 5,4,7. *Node* 7 terhubung ke *node* 6,4.

3.5.2. Melakukan Graph Partitioning

Setelah file masukan METIS sudah ada, yang dilakukan adalah memasukkan file ke program *graph partitioning* METIS yaitu *gpmemis*. Contoh file hasil *graph partitioning* dapat dilihat pada Gambar 3.8.

```
1
1
1
0
0
0
0
```

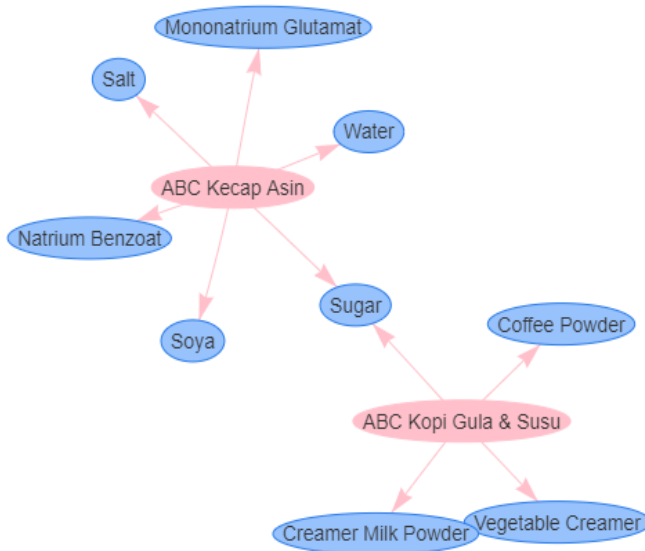
Gambar 3.8 Contoh File Output METIS

Node 1 termasuk ke kluster 1, *node* 2 termasuk kluster 1, *node* 3 termasuk kluster 1, *node* 4 termasuk kluster 0, dan seterusnya.

3.6. Visualisasi Graf

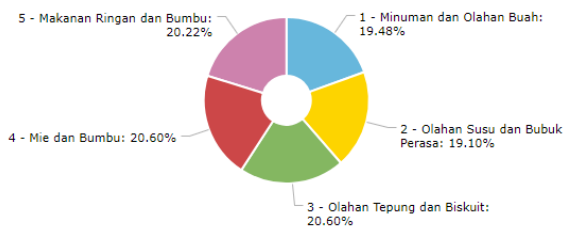
File keluaran dari METIS akan diubah menjadi format CSV untuk memudahkan proses impor ke basis data. Hasil *graph partitioning* divisualisasikan bersama dengan visualisasi graf. Data produk makanan dan bahannya dari basis data diubah menjadi bentuk JSON untuk visualisasi menggunakan D3js. Setelah itu, kode program javascript menggunakan *library* D3js dibuat untuk memvisualisasikan data graf makanan yang sudah dikelompokkan. Visualisasi bersifat interaktif meliputi fitur *selecting*, *zooming* dan *panning*. Berikut visualisasi graf yang akan ditampilkan pada Halal Nutrition Food.

- Visualisasi graf hubungan antar nama produk dan bahannya.



Gambar 3.9 Contoh Visualisasi Graf Hubungan Antara Nama Produk, Bahan Makanan

- Visualisasi hasil graph partitioning



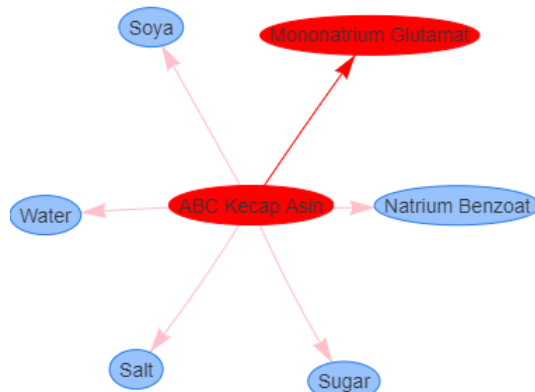
Gambar 3.10 Contoh Visualisasi Hasil Graph Partitioning

- Visualisasi produk makanan berstatus haram.



Gambar 3.11 Contoh Visualisasi Graf Produk Berstatus Haram

- Visualisasi produk makanan yang mengandung msg



Gambar 3.12 Contoh Visualisasi Graf Produk Ber-MSG

3.7. Pengujian

Pengujian yang dilakukan adalah seberapa banyak data yang diintegrasikan berdasarkan metode pengukuran kesamaan bahan makanan. Setelah melalui *data cleansing*, penghitungan jumlah data bahan makanan yang berhasil terintegrasi dilakukan untuk mengetahui lebih banyak mana data yang terintegrasi, apakah metode Levensthein atau Jaccard distance. Pengujian visualisasi dilakukan dengan menguji fungsionalitas dari visualisasi graf pada aplikasi Halal Nutrition Food.

3.8. Penyusunan Buku Tugas Akhir

Pada tahap ini dilakukan penyusunan laporan tugas akhir sebagai bentuk dokumentasi pengerjaan tugas akhir ini. Dalam laporan tersebut meliputi:

Bab I Pendahuluan

Pada bab ini akan dijabarkan identifikasi masalah penelitian meliputi latar belakang masalah, rumusan masalah, atasan masalah, tujuan, dan relevansi keilmuan.

Bab II Tinjauan Pustaka

Pada bab ini akan dijabarkan dasar teori, temuan dan penelitian sebelumnya yang digunakan sebagai acuan dalam penyusunan tugas akhir.

Bab III Metodologi Penelitian

Pada bab ini akan dijabarkan tahapan-tahapan apa saja yang dilakukan.

Bab IV Perancangan

Pada bab ini akan dijelaskan proses perancangan data cleansing, integrasi data, graph partitioning dan visualisasi graf.

Bab V Implementasi

Pada bab ini akan dijelaskan proses implementasi data cleansing, integrasi data, graph partitioning dan visualisasi graf.

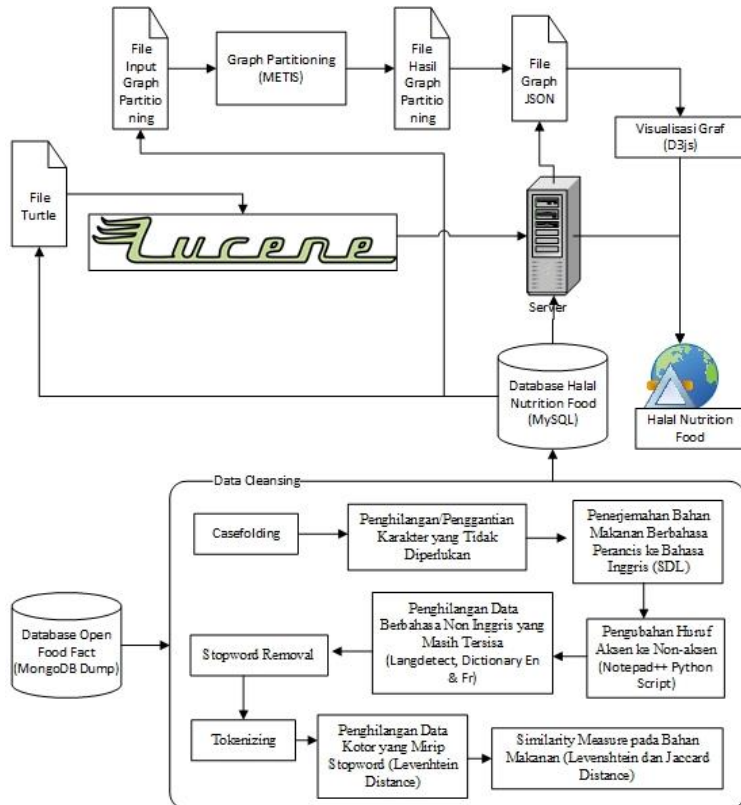
Bab VI Hasil dan Pembahasan

Pada bab ini akan dijelaskan hasil dan pembahasan dari hasil yang didapatkan pada tugas akhir.

Bab VII Kesimpulan dan Saran

Pada bab ini berisi tentang kesimpulan dan saran serta hal-hal yang dapat dikembangkan lebih lanjut.

3.9. Arsitektur Sistem



Gambar 3.13 Arsitektur Sistem Integrasi Data dan Visualisasi Graf pada Halal Nutrition Food

Berikut penjelasan arsitektur sistem pada Gambar 3.13.

1. Data dari Open Food Fact dibersihkan melalui data cleansing.
2. Data hasil cleansing diimpor ke basis data Halal Nutrition Food.
3. File turtle dihasilkan dari basis data sebagai masukan pengindeksan.

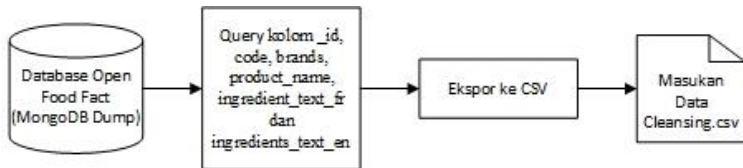
4. Apache Lucene, digunakan untuk memproses pencarian yang dilakukan pengguna dan menyimpan indeks dari dokumen (produk makanan dan bahannya).
5. Server, di dalamnya terdapat dua sistem yang berjalan yakni basis data dan aplikasi web Halal Nutrition Food.
 - Sistem basis data yang digunakan adalah MySQL, 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 Lucene untuk mensin pencarian.
6. Untuk melakukan graph partitioning pada METIS dibutuhkan file masukan yang berasal dari basis data.
7. Masukan visualisasi graf didapat dari basis data yang diubah menjadi bentuk JSON lalu divisualisasikan menggunakan D3js beserta hasil graph partitioning.

BAB IV PERANCANGAN

Pada bab ini akan dibahas mengenai rancangan penelitian untuk tugas akhir ini yang merupakan rancangan bagaimana penelitian akan dilakukan. Bab ini akan membahas proses mulai dari akuisisi data dan cleansing, integrasi data, graph partitioning dan visualisasi graf.

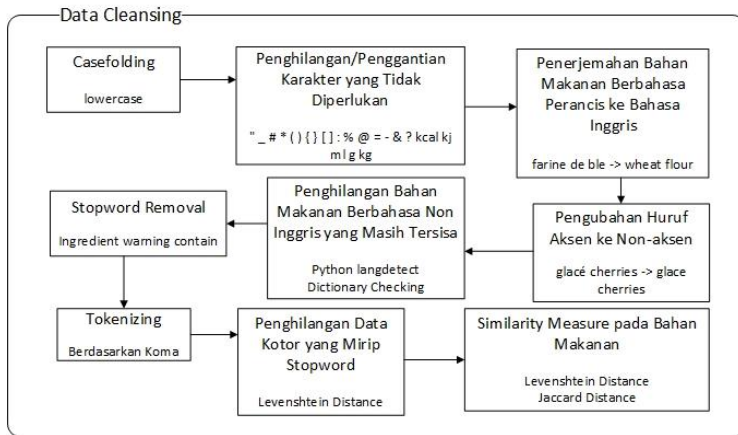
4.1. Akuisisi Data

Data didapat dari mongoddb dump. Data tersebut diquery untuk mendapatkan kolom `_id`, `code`, `brands`, `product_name`, `ingredient_text_fr` dan `ingredients_text_en` lalu diekspor ke dalam bentuk csv sehingga yang awalnya memiliki 442.467 menjadi 379.928 dokumen (produk makanan). Dari 379.928 produk, 347.727 produk memiliki bahan makanan, selebihnya tidak. Hingga akhirnya, ada 344.929 produk yang ditambahkan ke Halal Nutrition Food setelah data cleansing.



Gambar 4.1 Alur Akuisisi Data

4.2. Data Cleansing



Gambar 4.2 Alur Data Cleansing

Cleansing data bahan makanan dilakukan karena data bahan makanan masih bersifat sangat kotor. Berikut adalah contoh data cleansing pada data bahan makanan salah satu produk, yaitu Nestle Princessa Kokosowa.



Gambar 4.3 Contoh Produk Makanan beserta Bahannya

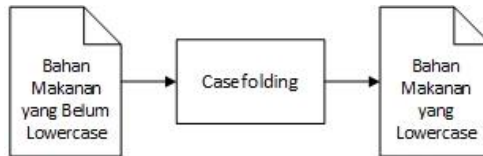
"ingredients_text" : "White chocolat (31%) (sugar, cocoa butter, full_milk_ powder, vegetable oils (palm, shea, illipe), whey powder (from_milk_), emulsifier (lecithins), flavouring), _wheat_ flour, palm oil, whey powder (from_milk_), sugar, skimmed_milk_ powder, desiccated coconut (2%), maltodextrin, flavourings, raising agents (ammonium carbonates, sodium carbonates), emulsifiers (lecithins, E476), salt."

Data seperti di atas akan menjadi seperti berikut.

| | | |
|----------------------|----------------------|-------------------|
| -white chocolate | -sugar | -cocoa butter |
| -milk powder | -vegetable oil | -palm |
| -shea | -illipe | -whey powder |
| -milk | -emulsifier | -lecithin |
| -flavouring | -wheat flour | -palm oil |
| -skimmed milk powder | -desiccated coconut | -maltodextrine |
| -raising agent | -ammonium carbonates | -sodium carbonate |
| -salt | | |

4.2.1. Casefolding

Casefolding dilakukan dengan mengubah data menjadi huruf kecil (lowercase). Contoh dapat dilihat pada Tabel 3.3.



Gambar 4.4 Alur Casefolding

4.2.2. Penghilangan/Penggantian Karakter yang Tidak Diperlukan

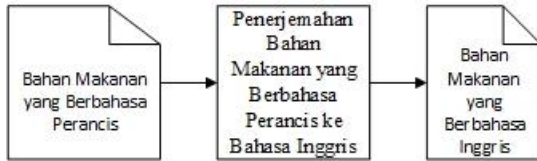
Penghilangan/penggantian karakter yang tidak diperlukan dilakukan dengan mengganti karakter yang sudah ditentukan menjadi karakter tertentu seperti yang dijelaskan pada Tabel 3.4 dan Tabel 3.5.



Gambar 4.5 Alur Penghilangan/Penggantian Karakter yang Tidak Diperlukan

4.2.3. Penerjemahan Bahan Makanan yang Berbahasa Perancis ke Bahasa Inggris

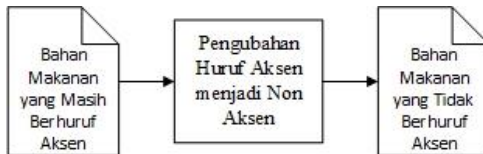
Bahan makanan yang masih berbahasa perancis akan diterjemahkan ke bahasa inggris. Penerjemahan dilakukan dengan tools SDL free translation[20] dengan membuat file teks (.txt) yang berisi bahan makanan dari kolom ingredient_text_fr lalu mengunggahnya untuk penerjemahan. Setelah diterjemahkan, kolom yang berisi bahan makanan akan digabung karena semua bahan makanan sudah diterjemahkan ke bahasa inggris sehingga nama kolom menjadi “ingredient”. Contoh dapat dilihat pada Tabel 3.6.



Gambar 4.6 Alur Penerjemahan Bahan Makanan yang Berbahasa Perancis ke Bahasa Inggris

4.2.4. Pengubahan Huruf Aksen menjadi Non Aksen

Beberapa nama produk dan bahan makanan memiliki huruf yang memiliki aksen (suara pengucapan). Untuk itu huruf aksen harus diabaikan dan diubah menjadi huruf yang tidak memiliki aksen. Contoh dapat dilihat pada Tabel 3.7.



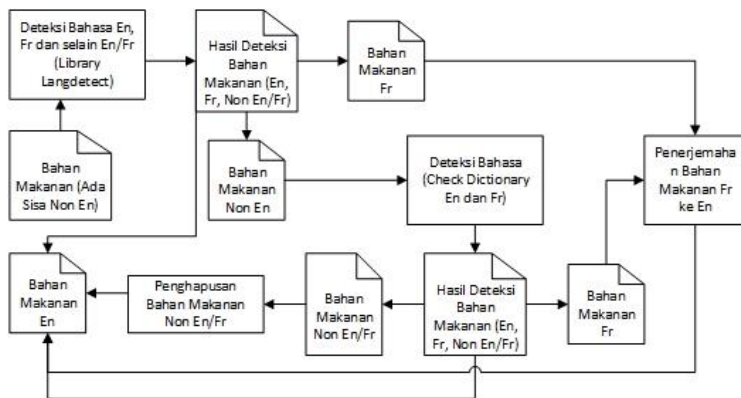
Gambar 4.7 Alur Pengubahan Huruf Aksen menjadi Non Aksen

4.2.5. Penghilangan Bahan Makanan Berbahasa Non Inggris yang Masih Tersisa

Meskipun data bahan makanan sudah diambil dari kolom bahan makanan berbahasa perancis dan inggris, ternyata masih terdapat bahan makanan yang berbahasa non inggris seperti

italia, jerman, spanyol, swedia dan belanda. Untuk itu, perlu adanya pendeteksian bahasa untuk mengetahui data bahan makanan yang berbahasa selain bahasa inggris dan perancis. Setelah terdeteksi, data makanan yang berbahasa non inggris dapat dihilangkan dan yang berbahasa perancis dapat ditransalsikan ke bahasa inggris.

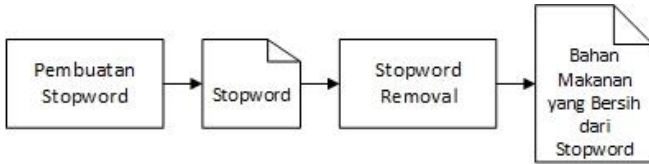
Ada 2 deteksi bahasa yang digunakan karena masih ada bahan makanan berbahasa inggris yang masih terdetekasi sebagai non inggris. Deteksi bahasa pertama, dilakukan dengan menggunakan library langdetect[21]. Deteksi bahasa kedua, dilanjutkan dengan pengecekan pada dictionary inggris[22] dan perancis[23] pada hasil deteksi pertama.



Gambar 4.8 Alur Penghilangan Data Berbahasa Non Inggris yang Masih Tersisa

4.2.6. Stopword Removal

Sebelum menghapus stopwords, tentu stopwords harus dibuat terlebih dahulu. Setelah stopwords dibuat, stopwords removal dilakukan dengan menghapus kata-kata termasuk dalam stopwords pada bahan makanan. Contoh dapat dilihat pada Tabel 3.8. Cuplikan stopwords dapat dilihat pada Tabel 4.1. Untuk selengkapnya dapat dilihat pada Lampiran B.



Gambar 4.9 Alur Stopword Removal

Tabel 4.1 Penggalan Stopword

| No. | Kategori | Stopword |
|-----|------------|---|
| 1 | Kandungan | Ingredient including traces of may contain made from that contain possible presence of not specified |
| 2 | Peringatan | to keep at unopened must be to consume preferably before stored in refrigerator in a non-metallic container consume within protected from warm and dry warning after opening store and preferably consume before best before shake well |

| No. | Kategori | Stopword |
|-----|--------------------|--|
| | | before opening after opening eat quickly keep dry not suitable children keep between away direct deciatatlon store in cool and dry place once opened container avoiding sharp edges keep it in the freezer |
| 3 | Informasi tambahan | net weight this is regularly checked during minimum durability period properly from organic farming no ingredients indicated from biological agriculture percentage of fat lower collagen ratio less than made in serving suggestion printed on top |

| No. | Kategori | Stopword |
|-----|----------|--|
| | | produced by see distributed by nutritional in varying proportions dosage weight best additional information vegetarians no preservatives unspecified superior quality proud of our simple our family recipe for allergens nutritional values averages |

4.2.7. Tokenizing

Setelah bahan makanan bersih dari stopwords, maka tokenizing berdasarkan koma (,) dilakukan. Karakter koma (,) perlu diganti dengan baris baru (\n). Contoh dapat dilihat pada Tabel 3.9.

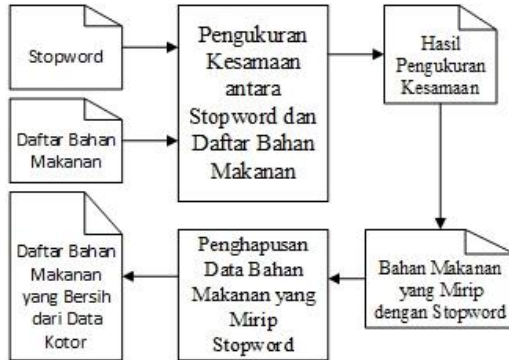


Gambar 4.10 Alur Tokenizing

4.2.8. Penghilangan Data Kotor yang Mirip Stopword

Selain stopwords, data bahan makanan yang bersifat kotor juga perlu dihilangkan. Untuk mengetahui data kotor tersebut maka diperlukan pengukuran kesamaan menggunakan levensthein

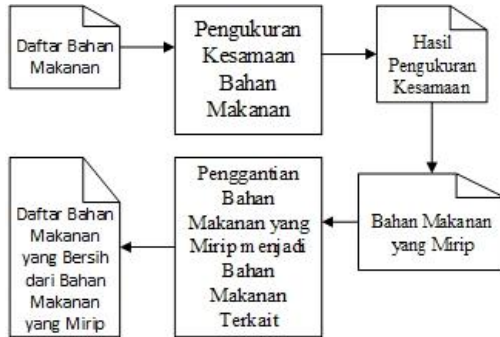
distance dengan membandingkan masing-masing stopword dengan data tersebut sehingga data bahan makanan yang mirip stopword dapat dihilangkan. Contoh dapat dilihat pada Gambar 3.5.



Gambar 4.11 Alur Penghilangan Data Kotor yang Mirip Stopword

4.2.9. Pengukuran Kesamaan pada Bahan Makanan

Setelah bahan makanan melalui tokenizing, maka bahan makanan akan diukur kemiripannya menggunakan levensthein distance dan jaccard similarity untuk memperbaiki kesalahan penulisan. Masing-masing bahan makanan akan diukur kesamaannya dengan membandingkannya dengan bahan makanan lainnya. Pengukuran kesamaan bahan makanan dilakukan dengan menggunakan library Levensthein distance dan Jaccard distance yang ada di nltk[24]. Setelah hasil pengukuran dihasilkan, bahan makanan yang mirip akan dipilih untuk diganti dengan bahan makanan terkait sehingga bahan makanan tidak mengalami salah penulisan. Contoh dapat dilihat pada Tabel 3.10 dan Tabel 3.11.



Gambar 4.12 Alur Pengukuran Kesamaan antar Bahan Makanan

4.2.10. Supervisi Manual terhadap Bahan Makanan

Setelah bahan makanan melalui pengukuran kesamaan, data bahan makanan diperiksa secara manual apakah data bahan makanan tersebut memang sebuah bahan makanan dan sudah benar penulisannya. Jika bahan makanan tersebut masih salah penulisannya maka diperbaiki dan jika bahan makanan tersebut tidak benar-benar sebuah bahan makanan maka dihapus. Oleh karena hal ini dilakukan secara manual, tidak semua bahan makanan telah melalui pemeriksaan.

4.3. Integrasi Data

4.3.1. Impor Data ke Basis Data

Impor ke basis data diperlukan untuk menghasilkan file turtle yang digunakan sebagai masukan saat melakukan index. Untuk melakukan impor ke basis data, maka data produk dan bahan makanan harus dalam bentuk CSV yang sudah sesuai struktur pada basis data.

Tabel 4.2 Penggalan CSV Hasil Data Cleansing

| Id | Code | Product_name | Ingredient |
|-------------------|-------------------|-----------------------------------|-----------------------------|
| 0846548063 851 | 0846548063 851 | cibo vita inc,peanut crunch | peanut,corn,sugar,h oney |

| Table Name | Field Name | Data Type |
|------------------------------|---------------------------|------------------|
| halal.foodproducts | id | int(10) unsigned |
| | fCode | varchar(255) |
| | fName | varchar(255) |
| | fManufacture | varchar(255) |
| | fVerify | int(11) |
| | fView | int(11) |
| | weight | int(11) |
| | calories | int(11) |
| | totalFat | double(8,2) |
| | saturatedFat | double(8,2) |
| | transFat | double(8,2) |
| | cholesterol | double(8,2) |
| | sodium | double(8,2) |
| | totalCarbohydrates | double(8,2) |
| | dietaryFiber | double(8,2) |
| | sugar | double(8,2) |
| | protein | double(8,2) |
| | vitaminA | int(11) |
| | vitaminC | int(11) |
| | calcium | int(11) |
| | iron | int(11) |
| | user_id | int(10) unsigned |
| | created_at | timestamp |
| updated_at | timestamp | |
| halal.foodproduct_ingredient | foodproduct_id | int(10) unsigned |
| | ingredient_id | int(10) unsigned |
| | created_at | timestamp |
| | updated_at | timestamp |
| | foodproduct_ingredient_id | int(10) unsigned |
| halal.ingredients | id | int(10) unsigned |
| | iName | varchar(255) |
| | iType | int(11) |
| | eNumber | varchar(255) |
| | updated_at | timestamp |

Gambar 4.13 Struktur Basis Data Halal Nutrition Food yang Terkait dengan Data Open Food Fact

Data CSV hasil *cleansing* seperti pada Tabel 4.2 perlu diubah menjadi 3 file CSV, sesuai struktur yang ada pada Gambar 4.13.

4.3.2. Melakukan Independent Ranking

Independent ranking dilakukan dengan mengecek apakah bahan makanan memiliki tautan di DBpedia untuk mengetahui seberapa banyak bahan makanan dari suatu produk, yang memiliki tautan di DBpedia. Selain itu, independent ranking

juga bertujuan untuk mengetahui suatu bahan makanan ada di produk mana saja. Independent ranking dihasilkan dari kode program yang telah dibuat pada penelitian sebelumnya oleh Ahmad Choirun Najib[5].

4.3.3. Menghasilkan File Turtle

File turtle akan dihasilkan dari kode program yang dibuat pada penelitian sebelumnya oleh Ahmad Choirun Najib[5] yang telah disempurnakan dimana kode program tersebut membuat file turtle dan menuliskannya dengan mengambil data dari basis data Halal Nutrition Food.

4.3.4. Mengindeks Produk dan Bahan Makanan

Data produk dan komposisi produk yang sudah tersimpan dalam bentuk file turtle akan diindex agar dapat muncul saat pencarian dilakukan. Pengindeksan dilakukan dengan menjalankan program java (.jar) yang telah dibuat pada penelitian sebelumnya oleh Ahmad Choirun Najib[5] dan Azmi Adi Firmansyah[6].

4.3.5. Menghasilkan Produk Terkait

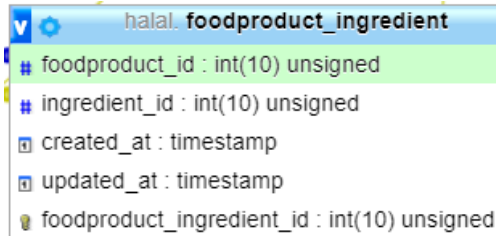
Produk terkait menampilkan produk yang mirip dengan produk yang dicari menggunakan metode cosine similarity dan euclidean distance dimana dihasilkan dari kode program yang dibuat pada penelitian Halal Nutrition Food sebelumnya oleh Azmi Adi Firmansyah[6].

4.4. Graph Partitioning

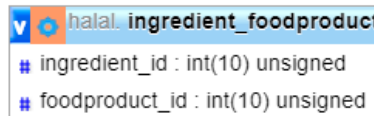
4.4.1. Membuat File Masukan METIS

File masukan untuk METIS dibuat dengan melakukan query pada basis data Halal Nutrition Food dan menuliskannya ke file. Query dilakukan untuk mengambil produk dan bahan makanan apa saja yang dikandungnya dari tabel foodproduct_ingredient; dan bahan makanan dan produk apa saja yang mengandungnya dari tabel ingredient_foodproduct yang ada di basis data Halal Nutrition Food. Hasil keduanya disusun ke dalam file sesuai dengan ketentuan pada Gambar

3.7. Struktur yang ada pada tabel `foodproduct_ingredient` dan `ingredient_foodproduct` dapat dilihat pada Gambar 4.14 dan Gambar 4.15.



Gambar 4.14 Struktur Tabel `foodproduct_ingredient`



Gambar 4.15 Struktur Tabel `ingredient_foodproduct`

4.4.2. Melakukan Graph Partitioning di METIS

Graph partitioning dilakukan dengan memasukkan file masukan ke program METIS. Graph akan dipartisi menjadi 20 kelompok sesuai kelompok produk yang ada di pencarian produk halal MUI seperti pada Tabel 4.3[25].

Tabel 4.3 Kelompok Produk pada Pencarian Produk Halal MUI

| No. | Nama Kelompok |
|-----|--------------------|
| 1 | Olahan daging |
| 2 | Olahan ikan |
| 3 | Olahan susu |
| 4 | Olahan telur |
| 5 | Olahan tumbuhan |
| 6 | Olahan tepung |
| 7 | Nasi dan lauk pauk |
| 8 | Mie dan pasta |
| 9 | Roti dan kue |
| 10 | Makanan ringan |
| 11 | Olahan coklat |
| 12 | Es krim |

| No. | Nama Kelompok |
|-----|--------------------|
| 13 | Selai dan jeli |
| 14 | Minuman |
| 15 | Minyak dan lemak |
| 16 | Bumbu dan rempah |
| 17 | Perasa dan aroma |
| 18 | Ekstrak dan enzyme |
| 19 | Pemanis |
| 20 | Aditif |

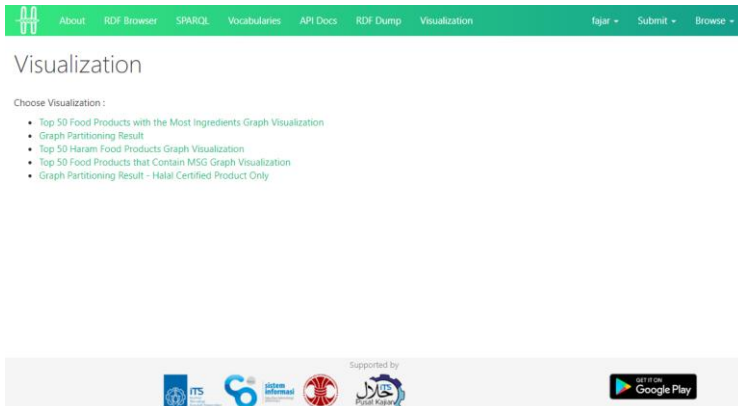
Hasil graph partitioning menunjukkan produk/bahan makanan dan kelompoknya yang masih berupa angka sehingga untuk mengetahui nama kelompok, setiap kelompok perlu dicari produk/bahan makanan apa yang paling mendominasi pada kelompok tersebut.

4.5. Visualisasi Graf

Setelah graph partitioning, data akan diubah menjadi bentuk JSON sebagai masukan untuk visualisasi graf oleh library D3js.

4.5.1. Desain Antarmuka Visualisasi Graf

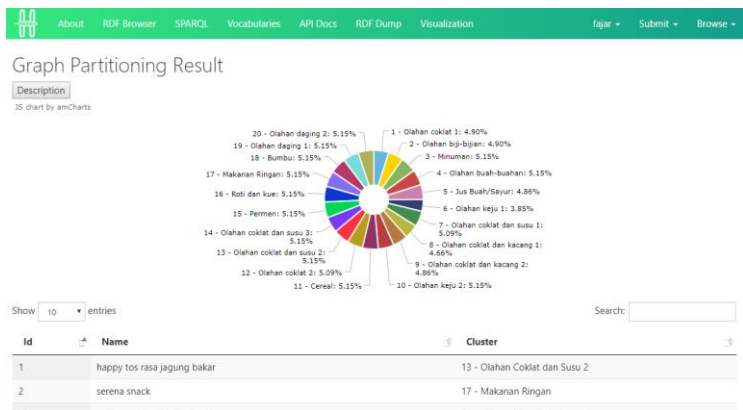
Berikut adalah desain antarmuka halaman visualisasi graf.



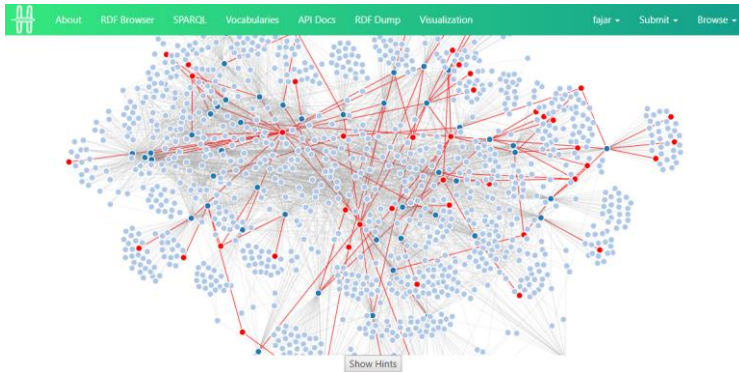
Gambar 4.16 Desain Antarmuka Halaman Utama Graph Visualization



Gambar 4.17 Desain Antarmuka Halaman Visualisasi Graf Hubungan Produk dan Bahan Makanan



Gambar 4.18 Desain Antarmuka Halaman Visualisasi Hasil Graph Partitioning



Gambar 4.19 Desain Antarmuka Halaman Visualisasi Graf Produk Berstatus Haram



Gambar 4.20 Desain Antarmuka Halaman Visualisasi Graf Produk Ber-MSG

4.6. Pengujian

4.6.1. Pengujian Jumlah Data yang Dihasilkan setelah Pengukuran Kesamaan Bahan Makanan

Pengujian pengujian jumlah data yang dihasilkan setelah pengukuran kesamaan bahan makanan akan diterangkan pada Tabel 4.4.

Tabel 4.4 Rancangan Pengujian Jumlah Data Bahan Makanan yang Dihasilkan setelah Pengukuran Kesamaan Bahan Makanan

| Butir Uji | Cara Pengujian | Hasil yang Diharapkan |
|-----------|---|--|
| JD1 | Menghitung jumlah data yang dihasilkan setelah melalui pengukuran kesamaan menggunakan Levenshtein distance | Jumlah data yang dihasilkan setelah melalui pengukuran kesamaan menggunakan Levenshtein distance |
| JD2 | Menghitung jumlah data yang dihasilkan setelah melalui pengukuran kesamaan menggunakan Jaccard distance | Jumlah data yang dihasilkan setelah melalui pengukuran kesamaan menggunakan Jaccard distance |

4.6.2. Pengujian Fungsional Visualisasi Graf

Rancangan pengujian fungsionalitas disesuaikan dengan visualisasi graf yang baik menurut Raga'ad M. Tarawneh[8]. Pengujian fungsionalitas visualisasi graf akan diterangkan pada Tabel 4.5.

Tabel 4.5 Rancangan Pengujian Fungsionalitas Visualisasi Graf

| Komponen | Butir Uji | Cara Pengujian | Hasil yang Diharapkan |
|----------|-----------|--|--|
| Zooming | FV1 | Melakukan zoom in dan zoom out pada visualisasi graf | Graf bisa di-zoom in dan zoom out |
| Panning | FV2 | Melakukan drag pada layar untuk melihat graf dari pandangan lain | Graf dapat dilihat dari sudut pandang lain |

| Komponen | Butir Uji | Cara Pengujian | Hasil yang Diharapkan |
|-----------|-----------|--|--|
| Node | FV3 | Melakukan select atau mengarahkan pointer ke salah satu node | Node dapat diselect atau muncul keterangan ketika dihover |
| Edge | FV4 | Mengarahkan pointer ke edge graf | Edge graf dapat terlihat jelas menunjukkan hubungan antar nodes |
| Filtering | FV5 | Melakukan filter untuk menunjukkan hal yang lebih difokuskan | Visualisasi graf dapat difilter untuk menunjukkan sesuatu yang lebih fokus |

BAB V IMPLEMENTASI

Pada bab ini akan dibahas dan dijelaskan mengenai proses implementasi penelitian berdasarkan perancangan yang telah dibahas pada bab sebelumnya.

5.1. Lingkungan Penelitian

Dalam melakukan implementasi pada tugas akhir ini memerlukan perangkat keras dan perangkat lunak pendukung. Perangkat keras dan lunak akan diterangkan pada Tabel 5.1 dan Tabel 5.2.

Tabel 5.1 Spesifikasi Perangkat Keras dalam Pengembangan Aplikasi

| Perangkat Keras | |
|-----------------|--------------------------|
| Perangkat | Spesifikasi |
| Jenis | Acer Aspire E5-475G-5574 |
| CPU | Intel Core i5 7200U |
| RAM | 20 GB |
| Disk | 1000 GB HDD |

Tabel 5.2 Spesifikasi Perangkat Lunak dalam Pengembangan Aplikasi

| Perangkat Lunak | |
|---|---------------------|
| Perangkat | Kegunaan |
| Windows 10 Education 64-bit, Linux Ubuntu | Sistem Operasi |
| Xampp 7.0.15 dengan PHP 5.6.14 | Webserver |
| Laravel 5.1.46 | PHP Web Framework |
| Spyder | Python IDE |
| EmEditor | Editor |
| METIS | Graph Partitioning |
| D3js | Graph Visualization |

5.2. Akuisisi Data

Sebagai masukan untuk data cleansing, data MongoDB dump Open Food Fact dipilih kolom id, code, brands, product_name, ingredient_text_fr dan ingredients_text_en lalu diekspor ke

dalam bentuk CSV dengan separator titik koma (;). Kolom brands dan product_name digabung karena pada struktur basis data Halal Nutrition Food tidak memiliki kolom brands.

Pada awalnya, data memiliki 442.467 dokumen, setelah diseleksi dan diekspor ke CSV data memiliki 379928 dokumen (produk makanan). Dari 379.928 produk, 347.727 produk memiliki bahan makanan, selebihnya tidak. Hingga akhirnya, ada 344.929 produk yang diproses ke data cleansing.

Pada data CSV ditemui bahan makanan yang terpisah ke baris selanjutnya sehingga jumlah baris antar kolom tidak sama dan perlu disatukan ke dalam satu baris. Hal itu dikarenakan ada karakter `\r\n` dan `\n` disela sela `ingredient_text_fr` maupun `ingredient_text_en` pada baris tertentu. Oleh karena itu, karakter `\r\n` atau `\n` dihapus menggunakan regular expression dan `replace`.

Pada data CSV ditemui jumlah kolom yang tidak konsisten karena disela-sela `ingredient_text_fr` maupun `ingredient_text_en` (bahan makanan) terdapat separator CSV yang mengakibatkan bahan makanan terpisah ke banyak kolom sehingga jumlah kolom tidak konsisten. Oleh karena itu, karakter titik koma (;) diganti dengan koma (,) pada baris yang memiliki jumlah kolom tidak konsisten menggunakan regular expression dan `replace`. Berikut contoh penghapusan karakter untuk memperbaiki kolom dan baris.

Tabel 5.3 Contoh Perbaikan Jumlah Kolom dan Baris sebelum Data Cleansing

| Karakter yang direplace | Direplace dengan karakter | Contoh sebelum | Contoh sesudah |
|-------------------------|---------------------------|---|--|
| <code>[;]\nt</code> | <code>[,]t</code> | acidifiant (E270), antioxydant (E300); Teneur totale en sel de | acidifiant (E270), antioxydant (E300),teneur totale en sel de cuisine: 1.7 %." |

| Karakter yang direplace | Direplace dengan karakter | Contoh sebelum | Contoh sesudah |
|-------------------------|---------------------------|------------------|----------------|
| | | cuisine: 1.7 %." | |

Setelah melalui penggantian seperti pada Tabel 5.3, data CSV sudah memiliki jumlah kolom yang konsisten; dan bahan makanan sudah masuk ke kolom `ingredient_text_fr` atau `ingredient_text_en` sehingga tidak ada yang terpisah pada baris yang berbeda. Data sebelum cleansing terdapat pada Lampiran B.

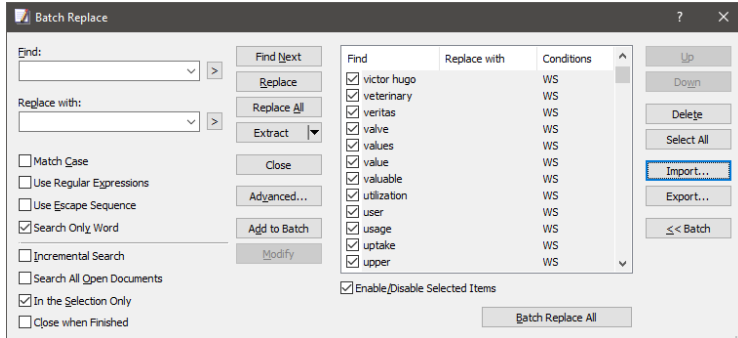
5.3. Data Cleansing

Dalam *data cleansing*, penggantian dan penghapusan karakter dilakukan dengan menggunakan fitur Batch Replace pada perangkat lunak EmEditor[26]. Contoh file impor yang berisi daftar yang ingin direplace ditampilkan pada Gambar 5.1.

| | 1 | 2 | 3 | 4 |
|---|----|-------|---|-----|
| 1 | on | kata1 | | WS↓ |
| 2 | on | ,, | , | WS↓ |

Gambar 5.1 Contoh File Impor Batch Replace EmEditor

Kolom pertama berisi on/off yang berarti apakah kata/karakter pada kolom kedua akan diganti atau tidak. Kolom kedua berisi daftar kata/karakter. Kolom ketiga berisi hasil ganti kata/karakter pada kolom kedua. Kolom keempat berisi WS yang berarti hanya mencari suatu kata dan hanya mencari pada teks yang mengalami *selection*. Tampilan batch replace pada EmEditor ditampilkan pada Gambar 5.2.



Gambar 5.2 Batch Replace pada EmEditor

5.3.1. Casefolding

Casefolding dilakukan dengan menggunakan fitur convert lowercase pada EmEditor.

5.3.2. Penghilangan/Penggantian Karakter yang Tidak Diperlukan

Penghilangan/penggantian karakter yang tidak diperlukan dilakukan dengan menggunakan fitur batch replace pada EmEditor.

5.3.3. Penerjemahan Bahan Makanan yang Berbahasa Perancis ke Bahasa Inggris

Penerjemahan bahan makanan yang berbahasa perancis ke bahasa inggris dilakukan dengan membuat file teks (.txt) yang berisi bahan makanan berbahasa perancis dari kolom ingredient_text_fr lalu memisah file teks menjadi 5 mb agar dapat diunggah ke SDL Document Translation[20] untuk diterjemahkan. Terdapat 178.083 produk dari 344.929 produk berbahasa makanan berbahasa perancis yang diterjemahkan ke bahasa inggris.

5.3.4. Pengubahan Huruf Aksen menjadi Non Aksen

Pengubahan huruf aksen menjadi non aksen dilakukan dengan menggunakan plugin python script pada notepad++[27].

Berikut adalah python script untuk mengubah huruf aksen menjadi non aksen. Contoh dapat dilihat pada Tabel 3.7.

```

1. “diinputkan per baris ke python script console
   Notepad++”
2. import Npp import *
3. import unicodedata
4. “diinputkan setelah select text berhuruf aksen”
5. eText = editor.getText() #mengambil teks berhuruf aksen
6. uText = unicode(eText, "UTF-8")
7. nText = unicodedata.normalize("NFKD",uText)
8. “diinputkan setelah pindah ke dokumen baru”
9. editor.addText(nText.encode('ASCII', 'ignore'))

```

Kode Program 5.1 Pengubahan Huruf Aksen menjadi Non Aksen

5.3.5. Penghilangan Data Berbahasa Non Inggris yang Masih Tersisa

Pendeteksian bahasa dilakukan menggunakan library langdetect[21]. File masukan dibuka dan dibaca per baris. File masukan adalah daftar bahan makanan yang sudah tidak berhuruf aksen. File masukan memiliki struktur seperti yang terlihat pada Gambar 5.3. Method untuk mendeteksi bahasa perancis, inggris dan selain keduanya dibuat dan dipanggil saat pencetakan.

```

abadejo alaska↓
abailloe↓
abajo↓
abalone↓
abalone room water↓
abalone shellfish↓

```

Gambar 5.3 Cuplikan Daftar Bahan Makanan

```

1. from langdetect import detect_langs
2. “deteksi bahasa en,fr,tidak keduanya”
3. def detect(string):
4.     res = detect_langs(string)
5.     for item in res:
6.         if item.lang == "fr" or item.lang == "en":
7.             return item.lang

```

```

8.     return None
9.     “membuka file masukan, membuat file keluaran dan
    mencetak bahan makanan sertahasil deteksi ke file
    keluaran”
10. with open('inputlangdetect.txt') as f:
11.     lines = f.read().split("\n")
12.     h = open('outputlangdetect.csv', 'w')
13.     for x in range(0, len(lines)-1, 1):
14.         h.write(lines[x]+' '+repr(detect(lines[x])))
15.         h.write('\n')
16.     h.close()

```

Kode Program 5.2 Deteksi Bahasa Menggunakan Langdetect

Cuplikan hasil dari deteksi bahasa menggunakan langdetect dapat dilihat pada Gambar 5.4. Untuk Selengkapnya dapat dilihat pada Lampiran B.

| | | |
|---------------|-----------------|-------|
| <u>clouet</u> | <u>bressons</u> | 'fr'↓ |
| <u>clouon</u> | | None↓ |
| cloves | chili | 'en'↓ |
| cloves | garlic | 'en'↓ |
| cloves | ginger | 'en'↓ |

Gambar 5.4 Cuplikan Hasil Deteksi Bahasa menggunakan Langdetect

Setelah pendeteksian pertama, bahan makanan yang berbahasa perancis akan ditranslate ke bahasa inggris, sedangkan bahan makanan yang berbahasa selain inggris dan perancis dideteksi menggunakan pengecekan di *dictionary* bahasa inggris[22] dan perancis[23]. File masukan dibuka dan dibaca per baris. File masukan adalah hasil deteksi bahasa langdetect yang berbahasa selain bahasa inggris dan perancis. File masukan memiliki struktur seperti yang terlihat pada Gambar 5.3. Deteksi dilakukan dengan mengecek apakah bahan makanan ada di dalam dictionary bahasa inggris. Jika ada, maka bahan makanan tersebut berbahasa inggris. Begitu juga untuk pengecekan dengan dictionary bahasa perancis. Namun, jika bahan makanan tidak ada di dalam kedua dictionary, maka bahan makanan tersebut bukan berbahasa inggris maupun perancis.

```

1. with open('inputdictionarycheck.tsv') as f:
2.     lines = f.read().split("\n")

```



```

3.     o = open('outputdictionarycheck.csv','w')
4.     for x in range(0,len(lines),1):
5.         if lines[x] in open('wordenglish.txt').read()
and lines[x] in open('wordfrench.txt').read() :
6.             o.write(lines[x]+' '+'en & fr")
7.             o.write('\n')
8.         elif lines[x] in
open('wordenglish.txt').read() :
9.             o.write(lines[x]+' '+'en")
10.            o.write('\n')
11.        elif lines[x] in open('wordfrench.txt').read()
:
12.            o.write(lines[x]+' '+'fr")
13.            o.write('\n')
14.        else :
15.            o.write(lines[x]+' '+'none")
16.            o.write('\n')
17.    o.close()

```

Kode Program 5.3 Deteksi Bahasa Menggunakan Pengecekan Dictionary

Cuplikan hasil dari pengecekan menggunakan pengecekan dictionary dapat dilihat pada Gambar 5.5. Untuk selengkapnya dapat dilihat pada Lampiran B.

| | |
|-------------------------|----------|
| <u>geflugelbouillon</u> | none↓ |
| <u>gelati</u> | en & fr↓ |
| gelatin | en & fr↓ |
| gelatinous | en↓ |
| <u>gelul</u> | fr↓ |

Gambar 5.5 Cuplikan Hasil Deteksi Bahasa menggunakan Pengecekan Dictionary

Setelah pengecekan pada dictionary, bahan makanan yang berbahasa perancis akan diterjemahkan ke bahasa inggris, sedangkan bahan makanan yang berbahasa selain inggris dan perancis akan dihapus. Terdapat 11.769 bahan makanan berbahasa selain inggris yang dihapus. *Dictionary* inggris dan perancis dapat dilihat pada Lampiran B.

5.3.6. Stopword Removal

Stopword removal dilakukan dengan menggunakan fitur batch replace EmEditor. Terdapat 491.134 penghapusan stopwords pada data. Contoh dapat dilihat pada Tabel 3.8. Untuk daftar stopwords dapat dilihat pada Lampiran B.

5.3.7. Tokenizing

Tokenizing dilakukan dengan mengganti tanda koma (,) dengan baris baru (\n) menggunakan fitur replace pada EmEditor.

5.3.8. Penghilangan Data Kotor yang Mirip Stopword

Pengukuran kesamaan dilakukan dengan membandingkan bahan makanan dan stopwords menggunakan Levenshtein distance yang ada di library nltk[24]. File masukan, bahan makanan dan daftar stopwords, dibuka dan dibaca per baris. File masukan memiliki struktur seperti yang terlihat pada Gambar 5.3. Penghitungan dilakukan dengan kondisi jika kedua kata yang diukur memiliki kesamaan karakter pertama dan kedua. Jika nilai t lebih besar atau sama dengan threshold yang ditentukan, 0.7, maka kedua kata yang diukur beserta hasil Levenshtein distance akan dikeluarkan dalam bentuk file dan bahan makanan yang mirip dengan stopwords akan dihapus.

Persamaan 5.1 menerangkan nilai t pada pengukuran kesamaan bahan makanan dengan stopwords menggunakan Levenshtein distance.

$$t(x, y) = \frac{\text{len}(x) - \text{levdis}(x, y)}{\text{len}(x)} \quad 5.1$$

Keterangan :

x = Bahan Makanan yang Dibandingkan dengan Stopword

y = Stopword

$\text{len}(x)$ = Panjang Karakter Bahan Makanan yang Dibandingkan

$\text{levdis}(x, y)$ =

Nilai Levenshtein Distance antara Bahan Makanan dengan Stopword

Berikut adalah baris kode pengukuran kesamaan bahan makanan dengan stopwords menggunakan Levenshtein distance.

```

1. import nltk
2. with open('stopword.tsv') as file1:
3.     linesf1 = file1.read().split("\n")
4.     file2 = open('inputpkbahndanstopword.txt','r')
5.     linesf2 = file2.read().split("\n")
6.     o = open('outputpkbahndanstopword.csv','w')
7.     for x in range(0,len(linesf1)-1,1):
8.         for y in range(0,len(linesf2)-1,1):
9.             "Dengan kondisi jika kedua kata yang diukur memiliki
             kesamaan karakter pertama dan kedua"
10.            if
             (list(linesf1[x])[0]==list(linesf2[y])[0] and
             list(linesf1[x])[1]==list(linesf2[y])[1]) :
11.                levdis =
             nltk.edit_distance(linesf1[x],linesf2[y])
12.                treshold = 0.7
13.                t = ((len(linesf1[x])-
             levdis)/len(linesf1[x]))
14.                if(t >= treshold):
15.                    o.write(linesf1[x] + ',' +
             linesf2[y] + ',' + repr(levdis))
16.                    o.write('\n')
17.            o.close()

```

Kode Program 5.4 Pengukuran Kesamaan Data Bahan Makanan dengan Stopword

File masukan penghilangan data kotor yang mirip stopwords dapat dilihat pada Lampiran B. Terdapat 100.410 penghapusan bahan makanan yang mirip dengan stopwords pada data. Cuplikan hasil pengukuran kesamaan data bahan makanan dengan stopwords menggunakan Levenshtein distance dapat dilihat pada Gambar 5.6. Untuk selengkapnya dapat dilihat pada Lampiran B.

| | | |
|-------------|---------------------|----|
| type of | <u>typeof</u> | 2↓ |
| type of | <u>ty6eof</u> | 2↓ |
| type of | <u>typeof</u> | 1↓ |
| typical | <u>typic</u> | 2↓ |
| utilization | <u>ut lisation</u> | 2↓ |
| utilization | <u>utilisatiol</u> | 2↓ |
| utilization | <u>utilisqtion</u> | 2↓ |
| utilization | <u>utifisatione</u> | 3↓ |

Gambar 5.6 Cuplikan Hasil Pengukuran Kesamaan Stopword dan Bahan Makanan

5.3.9. Pengukuran Kesamaan pada Bahan Makanan

Kesamaan bahan makanan diukur menggunakan Levenshtein distance dan Jaccard distance pada library nltk[24]. File masukan dibuka dan dibaca perbaris. File masukan bahan makanan dibuka dan dibaca perbaris lalu dibandingkan satu sama lain. File masukan memiliki struktur seperti yang terlihat pada Gambar 5.3. Penghitungan dilakukan dengan kondisi jika kedua kata yang diukur memiliki kesamaan karakter pertama dan kedua; dan panjang bahan makanan pertama lebih besar atau sama dengan bahan makanan kedua. Jika nilai t lebih besar atau sama dengan threshold yang ditentukan, 0.7, maka kedua kata yang diukur beserta hasil levenshtein distance akan dikeluarkan dalam bentuk file dan bahan makanan yang mirip akan diganti dengan bahan makanan terkait.

Persamaan 5.2 menerangkan nilai t pada pengukuran kesamaan bahan makanan menggunakan Levenshtein distance, sedangkan persamaan 5.3 menerangkan nilai t pada pengukuran kesamaan bahan makanan menggunakan Jaccard distance.

$$t(x,y) = \frac{\text{len}(x) - \text{levdis}(x,y)}{\text{len}(x)} \quad 5.2$$

Keterangan :

x = Bahan Makanan yang Dibandingkan

y = Bahan Makanan yang lain

$\text{len}(x)$ = Panjang Karakter Bahan Makanan yang Dibandingkan

$\text{levdis}(x,y)$ = Nilai Levenshtein Distance antar Bahan Makanan

$$t(x,y) = 1 - \text{jacdis}(x,y) \quad 5.3$$

Keterangan :

x = Bahan Makanan yang Dibandingkan

y = Bahan Makanan yang lain

$\text{jacdis}(x,y)$ = Jaccard Distance antar Bahan Makanan

Kode program pengukuran kesamaan bahan makanan menggunakan Levenshtein distance dapat dilihat pada Kode Program 5.5.

```

1. import nltk
2. with open('inputpkbahanmakanan.txt') as f:
3.     lines = f.read().split("\n")
4.     o = open('outputpkbahanmakananlev.csv','w')
5.     for x in range(0,len(lines),1):
6.         for y in range(len(lines)-1,x,-1):
7.             if (list(lines[x])[0]==list(lines[y])[0]
and list(lines[x])[1]==list(lines[y])[1] and
len(lines[x])>=len(lines[y])) :
8.                 levdis =
nltk.edit_distance(lines[x],lines[y])
9.                 treshold = 0.7
10.                t = ((len(lines[x])-
levdis)/len(lines[x]))
11.                if(t >= treshold):
12.                    o.write(lines[x] + '\t' + lines[y]
+ '\t' + repr(levdis))
13.                    o.write('\n')
14.                o.close()

```

Kode Program 5.5 Pengukuran Kesamaan Bahan Makanan Menggunakan Levenshtein Distance

Kode program pengukuran kesamaan bahan makanan menggunakan Jaccard distance dapat dilihat pada Kode Program 5.6.

```

1. import nltk
2. with open('inputpkbahanmakanan.txt') as f:
3.     lines = f.read().split("\n")
4.     o = open('outputpkbahanmakananjac.csv','w')
5.     for x in range(0,len(lines),1):
6.         for y in range(len(lines)-1,x,-1):
7.             s1=set(list(lines[x]))
8.             s2=set(list(lines[y]))
9.             if (list(lines[x])[0]==list(lines[y])[0]
and list(lines[x])[1]==list(lines[y])[1] and
len(lines[x])>=len(lines[y])) :
10.                jacdis = nltk.jaccard_distance(s1,s2)
11.                treshold = 0.7

```

```

12.         t=1-jacdis
13.         if(t >= treshold):
14.             o.write(lines[x] + '\t' + lines[y]
+ '\t' + repr(jacdis))
15.             o.write('\n')
16.     o.close()

```

Kode Program 5.6 Pengukuran Kesamaan Bahan Makanan Menggunakan Jaccard Distance

File masukan pengukuran kesamaan bahan makanan dapat dilihat pada Lampiran B. Cuplikan hasil pengukuran kesamaan pada bahan makanan dapat dilihat pada Gambar 5.7 dan Gambar 5.8. Untuk selengkapnya dapat dilihat pada Lampiran B.

| | | |
|--------------------|--------------------|----|
| cheddar cheese | chessar cheese | 3↓ |
| cheddar cheeses | chessar cheese | 3↓ |
| cheddar cheeses | cheddarcteesee | 3↓ |
| chlorella powder | chlorella powder1 | 1↓ |
| chloride potassium | chloride potasslum | 1↓ |
| chocolateof | chocolates | 2↓ |

Gambar 5.7 Cuplikan Hasil Pengukuran Kesamaan pada Bahan Makanan Menggunakan Levenshtein Distance

| | | |
|-------------------------|---------------------|----------------------|
| beef flavored stock | beef stock flavored | 0.0↓ |
| black tea decaffeinated | black tea infused | 0.14285714285714285↓ |
| beef ground beef | berries ground | 0.25↓ |
| broccoli powder | brocolli powder | 0.0↓ |
| brown rice | brown rive | 0.2↓ |
| canaola oil | canolia oil | 0.0↓ |

Gambar 5.8 Cuplikan Hasil Pengukuran Kesamaan pada Bahan Makanan Menggunakan Jaccard Distance

5.3.10. Supervisi Manual terhadap Bahan Makanan

Setelah bahan makanan melalui pengukuran kesamaan, data bahan makanan diperiksa secara manual apakah data bahan makanan tersebut memang sebuah bahan makanan dan sudah benar penulisannya. Supervisi manual dilakukan dengan mencari bahan makanan di mesin pencarian Google. Jika bahan makanan tersebut ada di Google, maka bahan makanan tersebut tidak dihapus. Jika bahan makanan tersebut masih salah

penulisannya maka bahan makanan tersebut diperbaiki sesuai penulisan yang ada di Google dan jika bahan makanan tersebut tidak benar-benar sebuah bahan makanan maka bahan makanan tersebut dihapus dari daftar bahan makanan. Oleh karena hal ini dilakukan secara manual, tidak semua bahan makanan telah melalui pemeriksaan.

5.4. Integrasi Data

5.4.1. Impor Data ke Basis Data

Berikut transformasi data hasil data cleansing menjadi data yang siap diimpor ke basis data Halal Nutrition Food.

- File Impor Tabel Foodproducts

Tabel 5.4 Transformasi ke File Impor Tabel Foodproducts

| Kolom | Nilai | Keterangan |
|--------------------|--------------|---|
| Id | 44300 dst... | Lebih besar dari id terbesar |
| fCode | Sesuai | Kolom code dari data hasil cleansing |
| fName | Sesuai | Kolom productname dari data hasil cleansing |
| fManufacture | | Karena tidak ada data manufaktur |
| fVerify | 1 | Karena sudah diverifikasi admin |
| fView | 0 | Default |
| Weight | 0 | Default |
| Calories | 0 | Default |
| totalFat | 0.00 | Default |
| saturatedFat | 0.00 | Default |
| transFat | 0.00 | Default |
| cholesterol | 0.00 | Default |
| sodium | 0.00 | Default |
| totalCarbohydrates | 0.00 | Default |
| dietaryFiber | 0.00 | Default |
| sugar | 0.00 | Default |
| protein | 0.00 | Default |

| Kolom | Nilai | Keterangan |
|------------|------------------------|---------------|
| vitaminA | 0 | Default |
| vitaminC | 0 | Default |
| calcium | 0 | Default |
| iron | 0 | Default |
| User_id | 29 | Id user |
| Created_at | 2018-06-16 00:00:00 | Tanggal impor |
| Updated_at | 2018-06-16 00:00:00 | Tanggal impor |

- File Impor Tabel Ingredients

Tabel 5.5 Transformasi ke File Impor Tabel Ingredients

| Kolom | Nilai | Keterangan |
|------------|------------------------|--|
| Id | 1101 dst... | Lebih besar dari id terbesar |
| iName | (nama bahan makanan) | Sesuai bahan makanan yang ditokenisasi dari kolom ingredient pada data hasil cleansing |
| iType | 0 atau 1 | 0 → non aditive 1 → aditive |
| eNumber | E100 dsb... | Sesuai bahan makanan dari kolom ingredient pada data hasil cleansing dimana bahan makanan tersebut sama dengan e-number tertentu |
| Created_at | 2018-06-16 00:00:00 | Tanggal impor |
| Updated_at | 2018-06-16 00:00:00 | Tanggal impor |

- File Impor Tabel Foodproduct_ingredient

Tabel 5.6 Transformasi ke File Impor Tabel Foodproduct_ingredient

| Kolom | Nilai | Keterangan |
|----------------|------------------------|---|
| Foodproduct_id | 44300 dst... | Sesuai kolom id pada tabel foodproducts |
| Ingredient_id | 1101 dst... | Sesuai kolom id pada tabel ingredients |
| Created_at | 2018-06-16 00:00:00 | Tanggal impor |
| Updated_at | 2018-06-16 00:00:00 | Tanggal impor |

5.4.2. Melakukan Independent Ranking

Independent ranking dengan menjalankan kode program php. Independent ranking dihasilkan dari kode program yang telah dibuat pada penelitian sebelumnya oleh Ahmad Choirun Najib[5].

5.4.3. Menghasilkan File Turtle

File turtle akan dihasilkan dari kode program php yang dibuat pada penelitian sebelumnya oleh Ahmad Choirun Najib[5] yang telah disempurnakan dimana kode program tersebut membuat file turtle dan menulisnya dengan mengambil data dari basis data Halal Nutrition Food. Kode tersebut berupa sebuah method yang dijalankan melalui web browser. Untuk kode selengkapnya dapat dilihat pada Lampiran A.

5.4.4. Mengindeks Produk dan Bahan Makanan

Pengindeksan produk dan bahan makanan dilakukan dengan menjalankan file java (.jar) yang berisi perintah menjalankan pengindeksan oleh Apache Lucene. Kode program pengindeksan dibuat pada penelitian sebelumnya oleh Ahmad Choirun Najib[5] dan Azmi Adi Firmansyah[6]. Sedangkan kode program untuk mengeksekusi program pengindeksan tersebut dapat dilihat pada Kode Program 5.7.

```

1. $eksekusiIndexFoodProduct = 'java -jar Halal.jar -docs
   /var/www/halal/public/resources/foodproducts';
2. $a = shell_exec($eksekusiIndexFoodProduct);
3. $eksekusiIndexIngredients = 'java -jar Halal.jar -docs
   /var/www/halal/public/resources/ingredients';
4. $b = shell_exec($eksekusiIndexIngredients);
5. echo $a;
6. echo $b;

```

Kode Program 5.7 Eksekusi Program Pengindeksan

5.4.5. Menghasilkan Produk Terkait

Produk terkait yang mirip dihasilkan oleh kode program yang dibuat pada penelitian Halal Nutrition Food sebelumnya oleh Azmi Adi Firmansyah[6]. Kode program untuk menjalankan program untuk menghasilkan produk terkait yang mirip dengan produk yang dicari dapat dilihat pada Kode Program 5.8.

```

1. $eksekusi = 'java -jar ./Similarity36.jar';
2. $output = shell_exec($eksekusi);
3. echo '<pre>'.$output.'</pre>'

```

Kode Program 5.8 Eksekusi Program untuk Menghasilkan Produk Terkait yang Mirip

5.5. Graph Partitioning

5.5.1. Mengubah File Masukan METIS

Untuk membuat file masukan METIS, kode program PHP dibuat untuk melakukan query ke kolom foodproduct_id dan ingredient_id pada tabel food product_ingredient; dan kolom ingredient_id dan foodproduct_id pada tabel ingredient_foodproduct. Hasil keduanya ditulis ke file. File tersebut berisi daftar id produk disertai bahannya; dan id bahan makanan disertai produk yang mengandungnya. File tersebut disesuaikan dengan ketentuan seperti pada Gambar 3.7. Kode program selengkapnya dapat dilihat pada Lampiran A.

5.5.2. Melakukan Graph Partitioning

Ada 2 data graf yang dipartisi, yaitu graf 311.659 produk dan 37.864 bahan makanan (349.523 nodes dan 2.081.821 edges),

dan graf produk bersertifikat halal yang memiliki bahan makanan (267 nodes dan 622 edges). Graph partitioning dilakukan dengan menjalankan METIS (gpmmetis) dengan parameter file masukan dan jumlah partisi yaitu 20 seperti pada Gambar 5.9. Pada file keluaran METIS, kelompok dimulai dari angka 0 dan seterusnya sehingga kelompok 0 akan diganti menjadi kelompok 1, 1 akan diganti menjadi kelompok 2, dan seterusnya. Keluaran graph partitioning dapat dilihat pada Lampiran C.

```

fajarberlian@fajarberlian: ~
fajarberlian@fajarberlian:~$ gpmmetis input.graph 20
*****
METIS 5.0 Copyright 1998-13, Regents of the University of Minnesota
(HEAD: , Built on: Feb 24 2018, 02:05:47)
size of idx_t: 32bits, real_t: 32bits, idx_t *: 64bits

Graph Information -----
Name: input.graph, #Vertices: 349523, #Edges: 2081821, #Parts: 20

Options -----
ptype=kway, objtype=cut, ctype=shem, rtype=greedy, iptype=metisrb
dbglvl=0, ufactor=1.030, no2hop=NO, minconn=NO, contig=NO, nooutput=NO
seed=-1, niter=10, ncuts=1

Direct k-way Partitioning -----
- Edgecut: 1424450, communication volume: 958282.

- Balance:
  constraint #0: 1.030 out of 0.000

- Most overweight partition:
  pid: 3, actual: 18001, desired: 17476, ratio: 1.03.

- Subdomain connectivity: max: 19, min: 19, avg: 19.00

- The original graph had 3220 connected components and the resulting
  partitioning after removing the cut edges has 36316 components.

Timing Information -----
I/O: 0.489 sec
Partitioning: 10.209 sec (METIS time)
Reporting: 0.271 sec

Memory Information -----
Max memory used: 134.816 MB
*****

```

Gambar 5.9 Eksekusi Graph Partitioning pada METIS

Hasil graph partitioning menunjukkan produk/bahan makanan dan kelompoknya yang masih berupa angka. Oleh karena itu, pemberian nama kelompok dilakukan dengan mencari anggota produk/bahan makanan yang mendominasi pada masing-masing kelompok. Untuk mencarinya, penghitungan jumlah kata perlu dilakukan untuk mengetahui kata yang sering muncul. Penghitungan jumlah kata yang paling sering muncul dilakukan dengan menggunakan wordcounter360.com[28].

5.6. Visualisasi Graf

Sebagai data yang digunakan saat visualisasi, ada 3 data JSON yang dibuat. Nodes pada file JSON memiliki atribut id, name, group dan color yang menjelaskan id, nama node, grup dan warna node. Edges memiliki atribut source, target, value dan colour yang menjelaskan id node asal, id node tujuan, tebal dan warna edge. Pada file JSON visualisasi graf hubungan antara produk dan bahan makanan, nodes memiliki 2 macam nilai group untuk menunjukkan produk atau bahan makanan serta warnanya, sedangkan edges memiliki nilai value yang sama semua dan tidak memiliki atribut colour. Pada file JSON visualisasi produk haram dan produk ber-MSG, nodes bahan yang haram/MSG memiliki nilai color “red”, sedangkan edges yang menghubungkan bahan tersebut dan produk memiliki nilai colour “red” dan memiliki nilai value lebih besar dari edge lainnya untuk mempertebal dan meng-highlight bahwa produk dan bahan tersebut haram/ber-MSG. Visualisasi hasil graf partitioning dilakukan dengan menunjukkan nama kelompok serta anggotanya dan jumlah anggota kelompok. File JSON selengkapnya dapat dilihat pada Lampiran B. Kode program javascript untuk menampilkan visualisasi graf dapat dilihat pada Kode Program 5.9.

```

1. function createV4SelectableForceDirectedGraph(svg, graph)
   {
2.     var width = +svg.attr("width"),
3.         height = +svg.attr("height");
4.     let parentWidth =
   d3v4.select('svg').node().parentNode.clientWidth;
5.     let parentHeight =
   d3v4.select('svg').node().parentNode.clientHeight;
6.     var svg = d3v4.select('svg')
7.         .attr('width', parentWidth)
8.         .attr('height', parentHeight)
9.         .append('g')
10.        .classed('g-main', true);
11.    var rect = gMain.append('rect')
12.        .attr('width', parentWidth)
13.        .attr('height', parentHeight)
14.        .style('fill', 'white')

```

```

15.     var gDraw = gMain.append('g');
16.     var zoom = d3v4.zoom() //zooming and panning
17.     .on('zoom', zoomed)
18.     gMain.call(zoom);
19.     function zoomed() {
20.         gDraw.attr('transform', d3v4.event.transform);
21.     }
22.     var color = d3v4.scaleOrdinal(d3v4.schemeCategory20);
23.     var nodes = {};
24.     var i;
25.     for (i = 0; i < graph.nodes.length; i++) {
26.         nodes[graph.nodes[i].id] = graph.nodes[i];
27.         graph.nodes[i].weight = 1.01;
28.     }
29.     var link = gDraw.append("g") //edges properties
30.     .attr("class", "link")
31.     .selectAll("line")
32.     .data(graph.links)
33.     .enter().append("line")
34.     .attr("stroke", function(d) { return
Math.sqrt(d.value/2); });
35.     var node = gDraw.append("g") //nodes properties
36.     .attr("class", "node")
37.     .selectAll("circle")
38.     .data(graph.nodes)
39.     .enter().append("circle")
40.     .attr("r", 5)
41.     .attr("fill", function(d) {
42.         if ('color' in d)
43.             return d.color;
44.         else
45.             return color(d.group);
46.     })
47.     .call(d3v4.drag())
48.     .on("start", dragstarted)
49.     .on("drag", dragged)
50.     .on("end", dragended)
51.     .on('click', connectedNodes);
52.     node.append("title") //pointer hover
53.     .text(function(d) {
54.         if ('name' in d)
55.             return d.name;
56.         else
57.             return d.id;
58.     });

```

```

59.     var simulation = d3v4.forceSimulation() //layout
60.     .force("link", d3v4.forceLink()
61.         .id(function(d) { return d.id; })
62.         .distance(function(d) {
63.             return 1000;
64.         })
65.     )
66.     .force("charge", d3v4.forceManyBody())
67.     .force("center", d3v4.forceCenter(parentWidth / 2,
parentHeight / 2))
68.     .force("x", d3v4.forceX(parentWidth/2))
69.     .force("y", d3v4.forceY(parentHeight/2));
70.     var toggle = 0;
71.     simulation
72.         .nodes(graph.nodes)
73.         .on("tick", ticked);
74.     simulation.force("link")
75.         .links(graph.links);
76.     function ticked() {
77.         // update node and line positions at every step of
78.         // the force simulation
79.         link.attr("x1", function(d) { return d.source.x;
})
80.             .attr("y1", function(d) { return d.source.y;
})
81.             .attr("x2", function(d) { return d.target.x;
})
82.             .attr("y2", function(d) { return d.target.y;
});
83.         node.attr("cx", function(d) { return d.x; })
84.             .attr("cy", function(d) { return d.y; });
85.     }
86.     var linkedByIndex = {};
87.     for (i = 0; i < graph.nodes.length; i++) {
88.         linkedByIndex[i + "," + i] = 1;
89.     };
90.     graph.links.forEach(function (d) {
91.         linkedByIndex[d.source.index + "," + d.target.index] =
1;
92.     });
93.     function neighboring(a, b) {
94.         return linkedByIndex[a.index + "," + b.index];
95.     }
96.     function connectedNodes() {
97.         if (toggle == 0) {

```

```
98.         d = d3.select(this).node().__data__;
99.         node.style("opacity", function (o) {
100.             return neighboring(d, o) | neighboring(o, d) ?
1 : 0.1; //neighboring true -> opacity 1, false -> opacity
0.1
101.         });
102.         link.style("opacity", function (o) {
103.             return o.source === d | o.target === d ? 1 :
0;
104.         });
105.         toggle = 1;
106.     } else {
107.         node.style("opacity", 1);
108.         link.style("opacity", 1);
109.         toggle = 0;
110.     }
111. }
112. //grag node
113. function dragstarted(d) {
114.     if (!d3.event.active)
simulation.alphaTarget(0.3).restart();
115.     d.fx = d.x;
116.     d.fy = d.y;
117. }
118. function dragged(d) {
119.     d.fx = d3.event.x;
120.     d.fy = d3.event.y;
121. }
122. function dragended(d) {
123.     if (!d3.event.active)
simulation.alphaTarget(0);
124. }
125. return graph;
126. }
```

Kode Program 5.9 Visualisasi Graf

Halaman ini sengaja dikosongkan

BAB VI

HASIL DAN PEMBAHASAN

Pada bab ini akan dijelaskan hasil serta analisis terhadap hasil yang diperoleh dari proses implementasi yang telah dibahas pada bab sebelumnya.

6.1. Hasil

6.1.1. Hasil Pengujian Jumlah Data yang Dihasilkan setelah Pengukuran Kesamaan Bahan Makanan

Hasil pengujian jumlah data yang dihasilkan setelah pengukuran kesamaan bahan makanan dapat dilihat pada Tabel 6.1

Tabel 6.1 Hasil Pengujian Jumlah Data yang Dihasilkan setelah Pengukuran Kesamaan Bahan Makanan

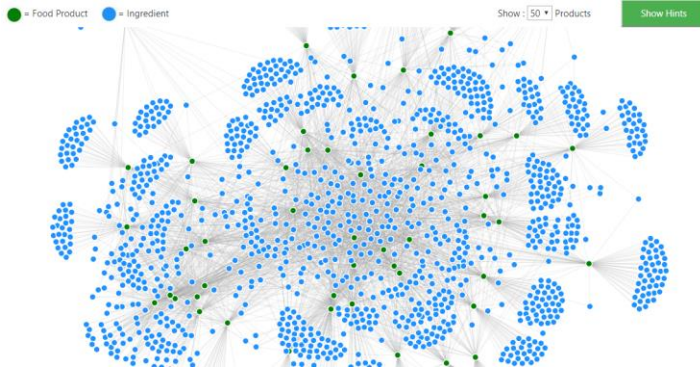
| Butir Uji | Metode Pengukuran Kesamaan | Jumlah Data Bahan Makanan sebelum Pengukuran Kesamaan | Jumlah Data Bahan Makanan setelah Pengukuran Kesamaan | Jumlah Data Bahan Makanan yang Terintegrasi |
|-----------|----------------------------|---|---|---|
| JD1 | Levenshtein distance | 234.417 | 220.947 | 222.024 bahan makanan |
| JD2 | Jaccard distance | | 194.167 | 195.243 bahan makanan |

6.1.2. Hasil Graph Partitioning

Hasil graph partitioning dapat dilihat pada Lampiran C.

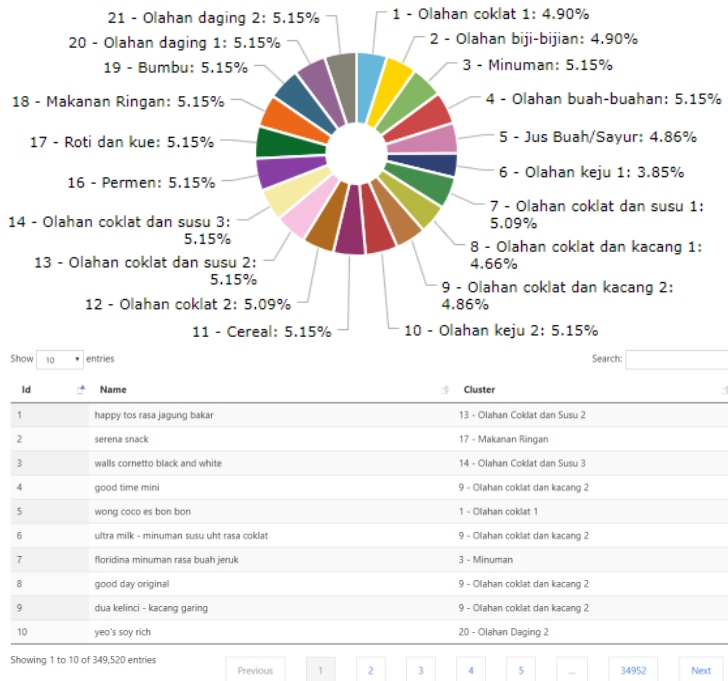
6.1.3. Hasil Visualisasi Graf

Berikut hasil visualisasi graf produk dan bahan makanan.

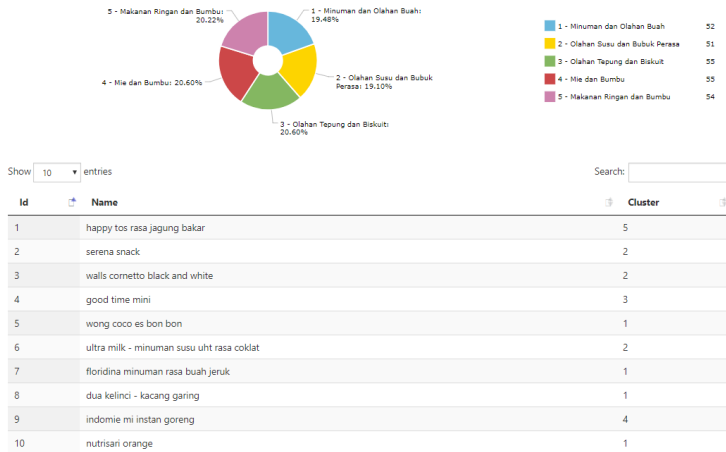


Gambar 6.1 Visualisasi Graf Hubungan Produk dan Bahan Makanan

Node produk dan bahan makanan memiliki warna yang berbeda untuk membedakannya. Edge yang menghubungkannya memiliki satu jenis warna saja.

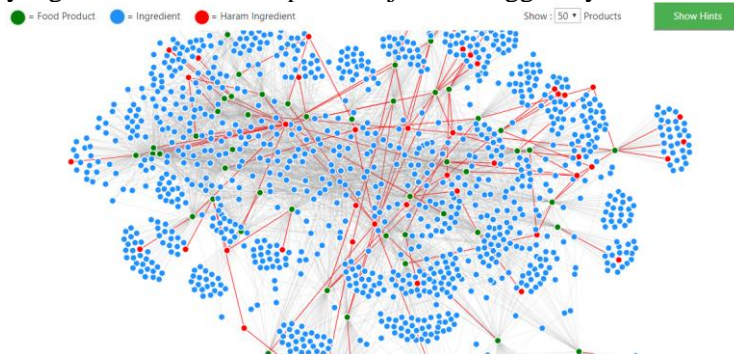


Gambar 6.2 Visualisasi Hasil Graph Partitioning



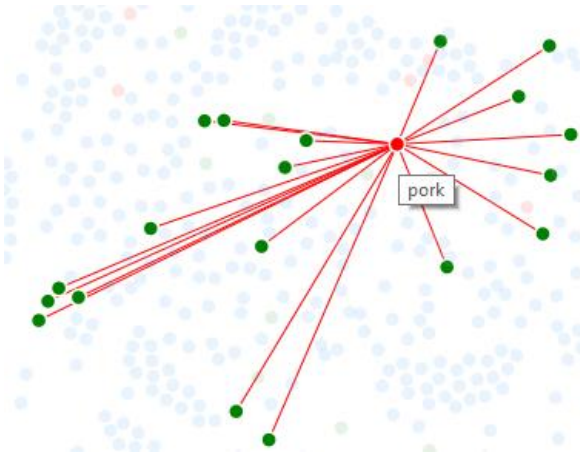
Gambar 6.3 Hasil Graph Partitioning - Produk yang Bersertifikat Halal

Hasil graph partitioning ditunjukkan dengan tabel daftar produk/bahan makanan dan nama kelompoknya serta pie chart yang berisi nama kelompok dan jumlah anggotanya.

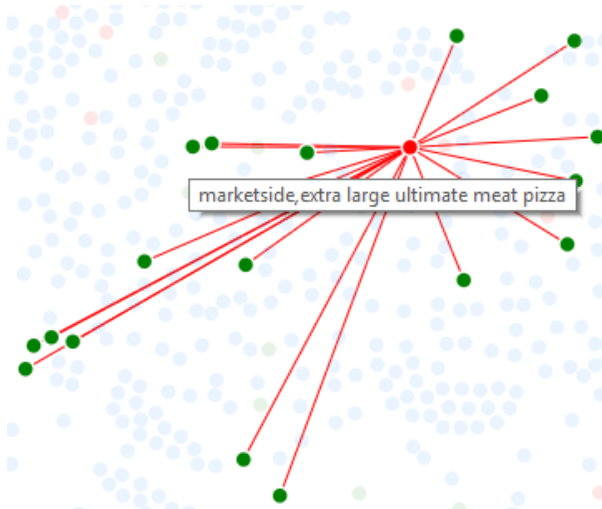


Gambar 6.4 Visualisasi Graf Produk dan Bahan Makanan Berstatus Haram – 1

Node bahan makanan yang haram di warnai merah, dimana warna tersebut berbeda dengan warna node lainnya. Begitu juga edge yang menghubungkan produk dan bahan makanannya. Keterangan nama produk dan nama bahan makanan muncul ketika pointer mouse di-hover pada node. Jika suatu node diklik, maka node lain yang tak terhubung akan tidak terlihat.



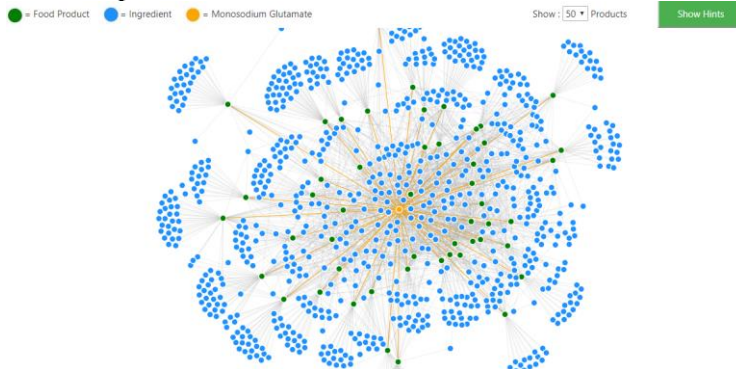
Gambar 6.5 Visualisasi Graf Produk dan Bahan Makanan Berstatus Haram - 2



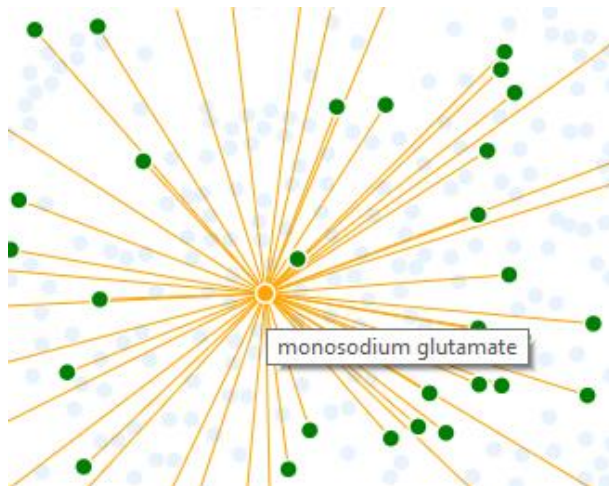
Gambar 6.6 Visualisasi Graf Produk dan Bahan Makanan Berstatus Haram - 3

Node bahan makanan MSG di warnai oranye, dimana warna tersebut berbeda dengan warna node lainnya. Begitu juga edge yang menghubungkan produk dan bahan makanannya.

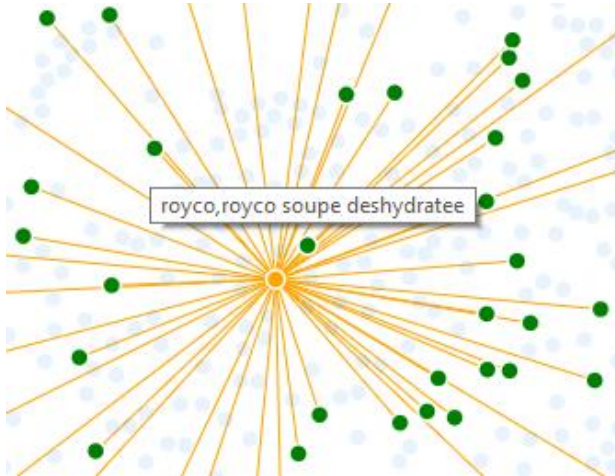
Keterangan nama produk dan nama bahan makanan muncul ketika pointer mouse di-hover pada node. Jika suatu node diklik, maka node lain yang tak terhubung akan tidak terlihat. Jumlah produk yang ditampilkan juga dapat diatur melalui menu dropdown.



Gambar 6.7 Visualisasi Graf Produk dan Bahan Makanan Ber-MSG – 1



Gambar 6.8 Visualisasi Graf Produk dan Bahan Makanan Ber-MSG – 2



Gambar 6.9 Visualisasi Graf Produk dan Bahan Makanan Ber-MSG –
3

6.1.4. Hasil Pengujian Fungsionalitas Visualisasi Graf

Hasil pengujian fungsionalitas visualisasi graf dapat dilihat pada Tabel 6.2.

Tabel 6.2 Hasil Pengujian Fungsionalitas Visualisasi Graf

| Komponen | Butir Uji | Cara Pengujian | Hasil yang Diharapkan | Hasil Uji |
|----------|-----------|--|--|-----------|
| Zooming | FV1 | Melakukan zoom in dan zoom out pada visualisasi graf | Graf bisa di-zoom in dan zoom out | Memenuhi |
| Panning | FV2 | Melakukan drag pada layar untuk melihat graf dari pandangan lain | Graf dapat dilihat dari sudut pandang lain | Memenuhi |

| Komponen | Butir Uji | Cara Pengujian | Hasil yang Diharapkan | Hasil Uji |
|-----------|-----------|--|--|-----------|
| Nodes | FV3 | Melakukan select atau mengarahkan pointer ke salah satu node | Node dapat diselect atau muncul keterangan ketika dihover | Memenuhi |
| Edges | FV4 | Mengarahkan pointer ke edge graf | Edge graf dapat terlihat jelas menunjukkan hubungan antar nodes | Memenuhi |
| Filtering | FV5 | Melakukan filter untuk menunjukkan hal yang lebih difokuskan | Visualisasi graf dapat difilter untuk menunjukkan sesuatu yang lebih fokus | Memenuhi |

6.2. Pembahasan

6.2.1. Pembahasan Hasil Pengujian Jumlah Data yang Dihasilkan setelah Pengukuran Kesamaan Bahan Makanan

Pada pengujian jumlah data yang dihasilkan setelah pengukuran kesamaan bahan makanan, metode Levenshtein distance memiliki jumlah data yang dihasilkan lebih besar daripada metode Jaccard distance. Hal itu menunjukkan bahwa dengan menggunakan Levenshtein distance, bahan makanan yang mirip lebih sedikit daripada menggunakan Jaccard distance sehingga penggantian bahan makanan yang mirip juga sedikit.

Pada pengukuran kesamaan menggunakan Levenshtein distance, terdapat 13.470 penggantian bahan makanan yang mirip. Sedangkan pada saat menggunakan Jaccard distance, terdapat 40.250. Agar lebih spesifik, pembahasan dititikberatkan ke hasil pengukuran kesamaan yang memiliki

nilai Levenshtein distance terkecil (1) dan nilai Jaccard distance yang terkecil (0). Penghitungan nilai Levenshtein distance dilakukan dengan menghitung jumlah penggantian, penyisipan dan penghapusan yang terjadi di saat suatu kata berubah menjadi suatu kata yang lain, seperti yang dijelaskan pada Gambar 3.5. Sebagai contoh, bahan makanan “tea” dan “ten” memiliki nilai Levenshtein distance sama dengan 1 (paling mirip). Seharusnya bahan makanan “tea” diganti menjadi “ten”, padahal keduanya memiliki beda arti. Oleh karena itu, pembatasan dilakukan dengan adanya persamaan 5.2. Jika nilai t pada persamaan 5.2 lebih kecil dari threshold (0,7) maka Levenshtein distance tidak dihasilkan dan bahan makanan tersebut tidak diganti. Bahan makanan “ten” dimasukkan stopword untuk dihilangkan.

Pengukuran menggunakan Jaccard distance memiliki penggantian yang lebih banyak karena banyak bahan makanan yang dinilai mirip. Banyak bahan makanan yang bernilai 0 atau berarti sama karena penghitungan nilai Jaccard distance diperoleh dengan mempertimbangkan anggota himpunan (karakter yang dimiliki oleh bahan makanan) sehingga jumlah, urutan dan tata letak karakter tidak dipertimbangkan. Sebagai contoh, “calcium phoshate” dan “calcium phospate” adalah bahan makanan yang dinilai sama. Keduanya memiliki karakter yang sama, yaitu {c,a,l,i,u,m, ,p,h,o,s,t,e} sehingga irisan dan gabungan memiliki nilai yang sama dan nilai Jaccard distance-nya sama dengan 0. Hal itu terjadi pada bahan makanan yang sebenarnya memiliki beda arti, seperti:

- black carrotflavouring dan blackcurrant flavoring,
- calcium carbonate coconut oil dan carbonate calcium,
- beetroot concentrate dan beta cerotene,
- dan sebagainya.

Karena terlalu banyak penggantian bahan makanan yang mirip, hasil pengukuran kesamaan menggunakan Jaccard distance tidak digunakan untuk integrasi dan pengukuran kesamaan

menggunakan Levenshtein distance-lah yang diintegrasikan ke Halal Nutrition Food.

6.2.2. Pembahasan Hasil Graph Partitioning

Berikut hasil graph partitioning berserta penamaan dari masing-masing kelompok.

Tabel 6.3 Hasil Graph Partitioning

| Kelompok | Jumlah Anggota | Nama Kelompok | Contoh Anggota |
|----------|----------------|--------------------------|--|
| 1 | 17117 | Olahan coklat 1 | abe's,chocolate chip cookies; alpro,silky smooth chocolate; chocolate extract |
| 2 | 17117 | Olahan biji-bijian | spartan,spanish olives vitarroz,seasoned rice with black beans; carrefour,haricots rouges |
| 3 | 17999 | Minuman | Water; coca-cola,coca cola,original taste; soda; 7 up,7 up-lemon-lime with mojito-150ml-france |
| 4 | 18001 | Olahan buah-buahan | Pomme; apple; fentimans,victorian lemonade |
| 5 | 16992 | Jus Buah/Sayur | Juice; 365 everyday value,100% juice concentrate from tart cherry; golden natural,aloe vera drink |
| 6 | 13458 | Olahan keju 1 | horizon organic,whitewave services inc.,organic mild cheddar cheese; carrefour; emmi,mozzarella maxi; auchan,lait,cheese |
| 7 | 17807 | Olahan coklat dan susu 1 | key food,ice cream bars,vanilla,dark chocolate; milk, winco foods,snickerdoodle cookies; cream |

| Kelompok | Jumlah Anggota | Nama Kelompok | Contoh Anggota |
|----------|----------------|----------------------------|--|
| 8 | 16277 | Olahan coklat dan kacang 1 | ah!!aska,non-dairy chocolate mix,organic cocoa; caramel, ahold,nature's promise,roasted mixed nuts |
| 9 | 16980 | Olahan coklat dan kacang 2 | Chocolate; peanut; butter; target stores,roasted in-shell peanuts; justnuts,justnuts nut bar cashew cranberry &cacao |
| 10 | 17999 | Olahan keju 2 | ht traders,jalapeno cheese straws; cheddar; ateeco inc.,potato & american cheese pierogies |
| 11 | 17992 | Cereal | family gourmet,toasted crispy rice cereal; selection,selection,pains aux5 cereales et aux graines; tipiak,cereales gourmandes |
| 12 | 17793 | Olahan coklat 2 | culinary circle,supervalu inc.,white chocolate hot cocoa mix; tillamook,ice cream,udderly chocolate; damhert,damhert tag sprits koek choco |
| 13 | 17998 | Olahan coklat dan susu 2 | joseph favrichon,muesli chocolat; yogurt; la molienda, milk candy wafers; caramel |
| 14 | 18001 | Olahan coklat dan susu 3 | leticia,oranges au chocolat 150g; georgia peach products inc.,reduced fat milk; cream |
| 15 | 18001 | Permen | Candy, chocolate, corn |
| 16 | 18001 | Roti dan kue | Bread; cookies; brookshire's,rising crust pizza; tylina foods,spring |

| Kelompok | Jumlah Anggota | Nama Kelompok | Contoh Anggota |
|----------|----------------|-----------------|--|
| | | | fling premium cake mix; tylina foods,easter eggs premium sugar cookie mix |
| 17 | 17993 | Makanan Ringan | Potato; charras,tortilla chips; hyvee,scoop corn chips; food lion,food town stores inc.,thin wheat crackers |
| 18 | 17995 | Bumbu | Sauce; bell white vinegar; j&f,chopped garlic; u,curry |
| 19 | 18001 | Olahan daging 1 | indiana jerky,chicken; jambon; singly-id food,nems au porc 300g - 10pcs - surgeles; chargrilled meat |
| 20 | 18001 | Olahan daging 2 | Beef; sausages; turkey; pere dodu,filet de poulet; asda,prime irish beef ribeye steak; carando,italian style meatballs |

Hasil graph partitioning pada Tabel 6.3 didapatkan dari graf 311.659 produk dan 37.864 bahan makanan, bukan dari semua produk dan bahan makanan. Nama kelompok yang dihasilkan tidak sesuai dengan pengelompokan yang dilakukan oleh MUI. Kelompok produk/bahan makanan pada Halal Nutrition Food meliputi olahan coklat, olahan daging, olahan biji-bijian, olahan coklat dan susu, olahan keju, olahan coklat dan kacang, jus buah/sayur, minuman, cereal, olahan buah-buahan, makanan ringan, bumbu dan permen. Adapun kelompok makanan dari pengelompokan oleh MUI yang tidak ada pada kelompok produk/bahan makanan pada Halal Nutrition Food, yaitu es krim, aditif, nasi dan lauk pauk, olahan telur, mie dan pasta, selai dan jeli, pemanis, ekstrak dan enzyme, dan minyak dan lemak. Pengelompokan tidak sesuai karena tidak semua data pada Halal Nutrition Food dijadikan masukan graph partitioning. Selain itu, produk yang berbahan dasar coklat

memiliki jumlah yang banyak di setiap kelompoknya sehingga saat penghitungan kata yang sering muncul, kata coklat muncul pada 12 kelompok produk/bahan makanan.

Tabel 6.4 Hasil Graph Partitioning - Produk yang Tersertifikasi Halal

| Kelompok | Jumlah Anggota | Nama Kelompok | Contoh Anggota Kelompok |
|----------|----------------|------------------------------|--|
| 1 | 52 | Minuman dan olahan buah | pineapple juice; floridina minuman rasa buah jeruk |
| 2 | 51 | Olahan susu dan bubuk perasa | ultra milk - minuman susu uht rasa coklat; artificial vanilla flavor |
| 3 | 55 | Olahan tepung dan kue | corn flour; regal marie biscuits |
| 4 | 55 | Mie dan bumbu | indomie mi rasa ayam bawang; fried onion |
| 5 | 54 | Makanan ringan dan bumbu | happy tos rasa jagung bakar; ginger; pepper powder |

Hasil graph partitioning pada Tabel 6.4 didapatkan dari produk bersertifikasi halal yang memiliki bahan makanan beserta bahan maknanya. Adapun jumlah produk bersertifikat halal yang memiliki bahan makanan, yaitu 51 produk dari 43.958 produk yang bersertifikat halal. Hasil pengelompokan produk/bahan makanan tidak sesuai dengan pengelompokan yang dilakukan oleh MUI karena produk bersertifikat halal yang memiliki bahan makanan hanya berjumlah 51 produk. Oleh karena itulah, graph partitioning dilakukan dengan mengelompokkan produk/bahan makanan ke 5 kelompok saja. Kelompok produk/bahan makanan pada produk bersertifikat halal hanya meliputi minuman, olahan buah-buahan, olahan tepung, kue, mie, bumbu dan makanan ringan sehingga nama kelompok tidak sebanyak pengelompokan yang dilakukan oleh MUI.

6.2.3. Pembahasan Hasil Pengujian Fungsionalitas Visualisasi Graf

Pada visualisasi graf hubungan produk makanan dan bahannya, zooming dapat dilakukan dengan melakukan scroll di layar. Panning dapat dilakukan dengan melakukan drag pada layar. Zooming dan panning dapat mempermudah eksplorasi dan memperjelas nodes maupun edges pada graf. Select dapat dilakukan pada nodes dengan melakukan hover pada nodes sehingga keterangan node akan muncul. Nodes terlihat jelas karena memiliki warna yang berbeda dengan edges. Edges terlihat jelas karena memiliki warna yang berbeda dengan nodes. Filtering digunakan untuk mengetahui node mana yang terhubung dengan node yang diamati. Filtering dilakukan dengan memberi highlight pada node yang terhubung dengan node terkait.

Pada visualisasi graf produk makanan berstatus haram, zooming dan panning juga dapat dilakukan. Nodes dapat memunculkan keterangan dan edges terlihat jelas. Filtering digunakan untuk mengetahui produk mana yang haram dengan memberi highlight pada node produk dan bahan makanan serta edges penghubung keduanya. Begitu juga pada visualisasi graf produk makanan ber-MSG. Selain itu, filtering digunakan untuk mengatur jumlah produk yang ditampilkan pada semua visualisasi graf, yaitu antara 10 – 50 produk.

Halaman ini sengaja dikosongkan

BAB VII

KESIMPULAN DAN SARAN

Bab ini menjelaskan kesimpulan yang didapat dari seluruh proses pengerjaan tugas akhir dan saran perbaikan untuk penelitian kedepannya untuk dapat dikembangkan dari tugas akhir ini.

7.1. Kesimpulan

Kesimpulan yang didapat berdasarkan pengerjaan tugas akhir ini adalah sebagai berikut.

1. Pengukuran kesamaan bahan makanan menggunakan Levenshtein distance lebih akurat dalam mengukur kesamaan daripada menggunakan Jaccard distance, dibuktikan dengan banyaknya file yang dinilai sama saat menggunakan Jaccard Distance padahal memiliki perbedaan arti dan bukan mengalami salah penulisan.
2. Graph partitioning (349.523 nodes dan 2.081.821 edges) bertujuan untuk mengelompokkan produk/bahan makanan menjadi 20 kelompok menggunakan METIS. Hasilnya, pengelompokan tidak sesuai dengan pengelompokan produk makanan yang dilakukan oleh MUI.
3. Visualisasi graf produk dan bahan makanan memenuhi semua pengujian yang dirancang, Namun, untuk jumlah data yang besar, pengguna memerlukan penggunaan fitur zooming dan panning yang lebih intens agar dapat membaca dengan jelas.

7.2. Saran

Saran untuk penelitian selanjutnya adalah sebagai berikut.

1. Pengukuran kesamaan bahan makanan dapat dilakukan menggunakan path-length similarity pada WordNet untuk mengetahui sinonim bahan makanan sehingga

bahan makanan yang sama arti akan memiliki penulisan yang sama pula.

DAFTAR PUSTAKA

- [1] Badan Pusat Statistika, “Penduduk Menurut Wilayah dan Agama yang Dianut | Indonesia,” 2010. [Online]. Available: <https://sp2010.bps.go.id/index.php/site/tabel?tid=321>. [Accessed: 06-Feb-2018].
- [2] LPPOM MUI, “Statistik Sertifikasi Halal Indonesia.” [Online]. Available: http://www.halalmui.org/mui14/index.php/main/go_to_section/59/1368/page/1. [Accessed: 06-Feb-2018].
- [3] “Open Food Fact Blog.” [Online]. Available: <https://en.blog.openfoodfacts.org/>. [Accessed: 07-Feb-2018].
- [4] H. Chen, “String Metrics and Word Similarity Applied to Information Retrieval,” University of Eastern Finland, 2012.
- [5] A. C. Najib, “Rancang Bangun Aplikasi Android Halal Nutrition Food Menggunakan Kombinasi Query Independent dan Query Dependent Ranking,” Institut Teknologi Sepuluh Nopember Surabaya, 2018.
- [6] A. A. Firmansyah, “Pengembangan Pencarian Produk Terkait Menggunakan Euclidean Distance dan Cosine Similarity pada Aplikasi Halal Nutrition Food,” Institut Teknologi Sepuluh Nopember Surabaya, 2018.
- [7] Y. Uylings, “Partitioning of Big Graphs,” University of Amsterdam, 2015.
- [8] R. M. Tarawneh, P. Keller, and A. Ebert, “A General Introduction to Graph Visualization Techniques,” 2011.
- [9] “Welcome To Halal Nutrition Food.” [Online]. Available: <http://halal.addi.is.its.ac.id/about>. [Accessed: 06-Feb-2018].
- [10] “Discover Open Food Fact.” [Online]. Available: <https://world.openfoodfacts.org/discover>. [Accessed: 08-Feb-2018].
- [11] W3C, “RDF 1.1 Primer,” *W3C Working Group Note 24 June 2014*, 2014. [Online]. Available:

- <https://www.w3.org/TR/rdf11-primer/>. [Accessed: 25-Feb-2018].
- [12] V. Levensthein, “Binary Codes Capable of Correcting Deletions, Insertions, and Reversals,” *Sov. Phys. Dokl.*, vol. 10, pp. 707–710, 1965.
- [13] P. Jaccard, “Étude comparative de la distribution florale dans une portion des Alpes et des Jura,” *Bull. la Société Vaudoise des Sci. Nat.*, vol. 37, pp. 547–579, 1901.
- [14] K. Tan, “Basic Concept,” *LuceneTutorial.com*, 2018. [Online]. Available: <http://www.lucene-tutorial.com/basic-concepts.html>. [Accessed: 15-Jun-2018].
- [15] A. Sonawane, “Using Apache Lucene to search text,” *IBM Developer Works*, 2009. [Online]. Available: <https://www.ibm.com/developerworks/library/os-apache-lucenesearch/index.html>. [Accessed: 16-Jun-2018].
- [16] R. J. Trudeau, *Introduction to Graph Theory*. New York: Dover Pub, 1993.
- [17] K. Andreev and H. Racke, “Balanced Graph Partitioning,” 2004.
- [18] G. Karypis, *METIS A Software Package for Partitioning Unstructured Graphs, Partitioning Meshes, and Computing Fill-Reducing Orderings of Sparse Matrices Version 5.1.0*. Minneapolis: Department of Computer Science University of Minnesota, 2013.
- [19] “D3js.” [Online]. Available: <https://d3js.org/>. [Accessed: 07-Feb-2018].
- [20] SDL, “SDL Free Translation.” .
- [21] M. M. Danilak, “Langdetect.” .
- [22] “English Words.” [Online]. Available: <https://github.com/dwyl/english-words/blob/master/words.txt>.
- [23] “French Dictionary.” [Online]. Available: <https://github.com/giacomodrago/ruzzlesolverpro/blob/master/languages/French/dictionary.txt>.
- [24] NLTK Team, “NLTK Metrics.” .

- [25] “Pencarian Produk Halal MUI.” [Online]. Available: http://www.halalmui.org/mui14/index.php/main/ceklogin_halal/produk_halal_masuk/1. [Accessed: 20-Jun-2018].
- [26] Emurasoft Inc., “EmEditor.” .
- [27] Notepad++ Team, “Notepad ++.” .
- [28] “WordCounter360.” .

Halaman ini sengaja dikosongkan

A. LAMPIRAN A : KODE PROGRAM

A.1 Pembuatan File Masukan METIS

Kode program pembuatan file masukan METIS dapat dilihat pada tautan <https://intip.in/TAHalal2018>.

Halaman ini sengaja dikosongkan

B. LAMPIRAN B : DATA

Data yang ada di bawah ini dapat dilihat pada tautan <https://intip.in/TAHalal2018>.

- B.1 Contoh file rdf open food fact
- B.2 Contoh file turtle pada halal nutrition food
- B.3 Data sebelum data cleansing
- B.4 Data masukan deteksi bahasa menggunakan langdetect
- B.5 Data hasil deteksi bahasa menggunakan langdetect
- B.6 Data masukan deteksi bahasa menggunakan pengecekan dictionary
- B.7 Data dictionary bahasa inggris dan perancis
- B.8 Data hasil deteksi bahasa menggunakan pengecekan dictionary
- B.9 Data daftar stopword
- B.10 Data daftar bahan makanan sebagai masukan pengukuran kesamaan bahan makanan
- B.11 Data hasil pengukuran kesamaan menggunakan levenshtein distance
- B.12 Data hasil pengukuran kesamaan menggunakan jaccard distance
- B.13 Data daftar bahan makanan setelah pengukuran kesamaan
- B.14 Data daftar bahan makanan yang terintegrasi
- B.15 Data impor csv ke basis data halal nutrition food

- B.16 Data file turtle yang telah diindeks
- B.17 Data file masukan METIS
- B.18 Data file keluaran METIS
- B.19 Data file JSON sebagai masukan visualisasi graf

C. LAMPIRAN C : HASIL GRAPH PARTITIONING

Data keluaran dari METIS dan hasil graph partitioning dapat dilihat di tautan <https://intip.in/TAHalal2018>.

Halaman ini sengaja dikosongkan

BIODATA PENULIS



Penulis lahir di Bojonegoro pada tanggal 4 April 1996. Penulis berasal dari Tuban dan telah menempuh pendidikan formal yaitu; SDN Sokosari 01 Tuban, SMP Plus Ar-Rahmat Bojonegoro, dan MAN 3 Malang. Pada tahun 2014, penulis melanjutkan pendidikan di Departemen Sistem Informasi, FTIF, Institut Teknologi Sepuluh Nopember (ITS) Surabaya dan terdaftar sebagai mahasiswa dengan NRP 05211440000144. Selama menjadi mahasiswa, penulis juga aktif mengikuti kegiatan kemahasiswaan seperti himpunan mahasiswa dan panitia event tingkat nasional. Pada tahun keempat, penulis mengambil bidang minat Laboratorium Akuisisi Data dan Diseminasi Informasi (ADDI). Penulis dapat dihubungi melalui email fajarberli@gmail.com.