

**ANALISIS SENTIMEN ULASAN DI *GOOGLE MAPS* MENGGUNAKAN
METODE *SUPPORT VECTOR MACHINE* DENGAN SELEKSI FITUR *MUTUAL
INFORMATION* DAN *CHI SQUARE* UNTUK REKOMENDASI DESTINASI
TEMPAT PARIWISATA DI KOTA YOGYAKARTA**

TUGAS AKHIR

Tugas Akhir ini sebagai salah satu syarat untuk memperoleh gelar sarjana
Teknik Informatika Universitas Pembangunan Nasional “Veteran” Yogyakarta



Disusun oleh:

Yuniar Rahmatika

123160168

**PROGRAM STUDI INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNIK INDUSTRI
UNIVERSITAS PEMBANGUNAN NASIONAL “VETERAN”
YOGYAKARTA
2021**

HALAMAN PENGESAHAN PEMBIMBING

ANALISIS SENTIMEN ULASAN DI GOOGLE MAPS MENGGUNAKAN METODE SUPPORT VECTOR MACHINE DENGAN SELEKSI FITUR MUTUAL INFORMATION DAN CHI SQUARE UNTUK REKOMENDASI DESTINASI TEMPAT PARIWISATA DI KOTA YOGYAKARTA

Disusun oleh :

Yuniar Rahmatika
123160168



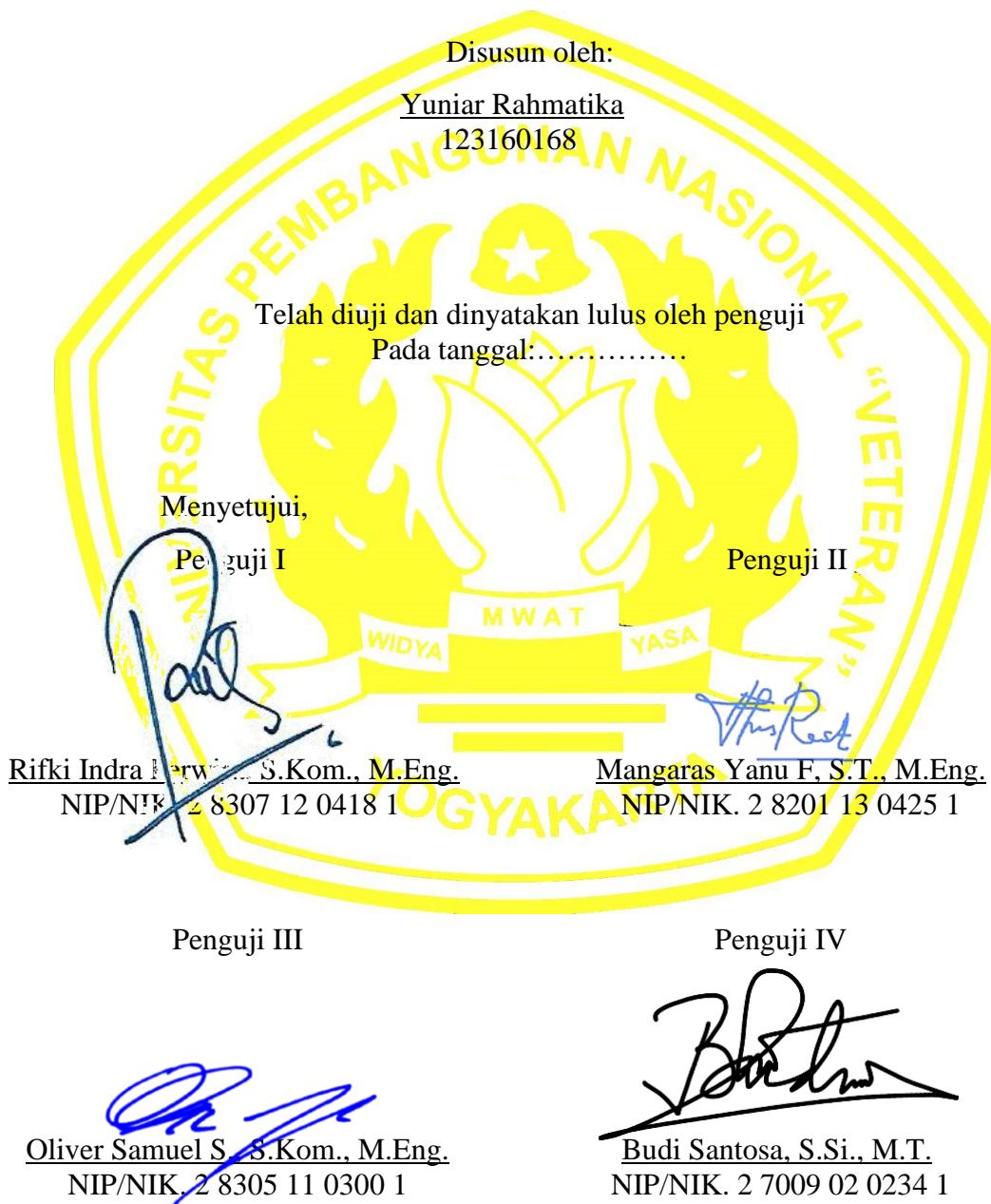
Mengetahui,
Koordinator Program Studi

Dr. Heriyanto, A.Md., S.Kom., M.Cs.

NIP/NIK. 2 7706 11 0301 1

HALAMAN PENGESAHAN PENGUJI

ANALISIS SENTIMEN ULASAN DI GOOGLE MAPS MENGGUNAKAN METODE SUPPORT VECTOR MACHINE DENGAN SELEKSI FITUR MUTUAL INFORMATION DAN CHI SQUARE UNTUK REKOMENDASI DESTINASI TEMPAT PARIWISATA DI KOTA YOGYAKARTA



SURAT PERNYATAAN KARYA ASLI TUGAS AKHIR

Sebagai mahasiswa Program Studi Informatika Fakultas Teknik Industri Universitas Pembangunan Nasional “Veteran” Yogyakarta, yang bertanda tangan dibawah ini, saya :

Nama : Yuniar Rahmatika

No Mhs : 123160168

Menyatakan bahwa karya ilmiah saya yang berjudul :

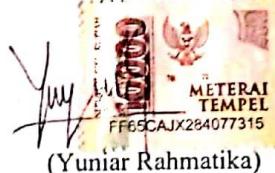
ANALISIS SENTIMEN ULASAN DI GOOGLE MAPS MENGGUNAKAN METODE SUPPORT VECTOR MACHINE DENGAN SELEKSI FITUR MUTUAL INFORMATION DAN CHI SQUARE UNTUK REKOMENDASI DESTINASI TEMPAT PARIWISATA DI KOTA YOGYAKARTA

merupakan karya asli saya dan belum pernah dipublikasikan dimanapun. Apabila di kemudian hari, karya saya disinyalir bukan merupakan karya asli saya, maka saya bersedia menerima konsekuensi apa pun yang diberikan Program Studi Informatika Fakultas Teknik Industri Universitas Pembangunan Nasional “Veteran” Yogyakarta kepada saya.

Demikian surat pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Yogyakarta
Pada tanggal : 9 Juli 2021

Yang menyatakan



(Yuniar Rahmatika)

PERNYATAAN BEBAS PLAGIAT

Saya yang bertanda tangan dibawah ini :

Nama : Yuniar Rahmatika

No. Mhs : 123160168

Fakultas/Prodi : Teknik Industri/ Informatika

Dengan ini saya menyatakan bahwa judul Tugas Akhir

ANALISIS SENTIMEN ULASAN DI GOOGLE MAPS MENGGUNAKAN METODE SUPPORT VECTOR MACHINE DENGAN SELEKSI FITUR MUTUAL INFORMATION DAN CHI SQUARE UNTUK REKOMENDASI DESTINASI TEMPAT PARIWISATA DI KOTA YOGYAKARTA

adalah hasil kerja saya sendiri dan benar bebas dari plagiat kecuali cuplikan serta ringkasan yang terdapat di dalamnya telah saya jelaskan sumbernya (situs) dengan jelas. Apabila pernyataan ini terbukti tidak benar maka saya bersedia menerima sanksi sesuai peraturan Mendiknas RI No 17 Tahun 2010 dan Peraturan Perundang-undangan yang berlaku.

Demikian surat pernyataan ini saya buat dengan penuh tanggung jawab.

Yogyakarta, 9 Juli 2021

Yang membuat pernyataan



Yuniar Rahmatika

NIM 123160168

ABSTRAK

Analisis sentimen menggunakan algoritma *machine learning* untuk proses klasifikasinya. Algoritma *machine learning* yang sering kali digunakan antara lain *Support Vector Machine*, *Naïve Bayes*, *Decision Tree* dan *K-Nearest Neighbor*. *Support Vector Machine* merupakan metode klasifikasi yang memiliki performansi lebih baik dibanding metode yang lainnya. Namun, *Support Vector Machine* memiliki kekurangan dalam pemilihan fitur yang sesuai. Seleksi fitur *Mutual Information* dan *Chi Square* dapat mengatasi permasalahan pemilihan fitur pada metode *Support Vector Machine*. Akan tetapi, belum diketahui perbandingan tingkat akurasi dari seleksi fitur *Mutual Information* dan *Chi Square* pada metode *Support Vector Machine* untuk permasalahan yang sama. Oleh karena itu, pada penelitian ini akan melakukan analisis sentimen dengan menggunakan data dari penilaian tempat pariwisata yang ada pada fitur ulasan di *Google Maps* dengan membandingkan seleksi fitur *Chi Square* dan *Mutual Information* sehingga dapat diketahui seleksi fitur mana yang memiliki performansi yang lebih baik ketika diterapkan pada metode *Support Vector Machine*. Dari hasil pengujian model klasifikasi dari kedua metode tersebut, memperlihatkan bahwa seleksi fitur *Mutual Information* memiliki hasil akurasi lebih baik sebesar 90.3% dengan presisi sebesar 90.2% dan *recall* sebesar 90.3% dibandingkan seleksi fitur *Chi Square* yang hanya menghasilkan akurasi sebesar 90.19% dengan presisi sebesar 89.93% dan *recall* sebesar 90.19%.

Kata Kunci : Analisis Sentimen, Seleksi Fitur, *Support Vector Machine*, *Chi Square*, *Mutual Information*

ABSTRACT

Sentiment analysis uses machine learning algorithms for the classification process. Machine learning algorithms that are often used include Support Vector Machine, Naïve Bayes, Decision Tree and K-Nearest Neighbor. Support Vector Machine is a classification method that has better performance than other methods. However, the Support Vector Machine lacks in the selection of appropriate features. Mutual Information and Chi Square feature selection can solve the problem of feature selection in the Support Vector Machine method. However, the comparison of the accuracy of the Mutual Information and Chi Square feature selection in the Support Vector Machine method is not yet known for the same problem. Therefore, this study will conduct a sentiment analysis using data from the assessment of tourism places in the review feature on Google Maps by comparing the Chi Square and Mutual Information features so that it can be seen which feature selection has better performance when applied to the Support method. From the results of the classification testing of the two methods, the Mutual Information feature selection results in a better accuracy of 90.3% with a precision of 90.2% and recall of 90.3% compared to the Chi Square feature selection which only produces an accuracy of 90.19% with a precision of 89.93% and recall of 90.19%.

Keyword : Sentiment Analysis, Feature Selection, Support Vector Machine, Chi Square, Mutual Information

KATA PENGANTAR

Segala puji syukur penulis panjatkan kepada Allah SWT yang telah melimpahkan rahmat-Nya serta hidayah-Nya sehingga penulis dapat menyelesaikan tugas akhir yang berjudul “Analisis Sentimen Ulasan di *Google Maps* menggunakan Metode *Support Vector Machine* dengan Seleksi Fitur *Mutual Information* dan *Chi Square* untuk Rekomendasi Destinasi Tempat Pariwisata di Kota Yogyakarta”. Tugas akhir ini disusun untuk memenuhi salah satu syarat kelulusan S1 di Program Studi Teknik Informatika, UPN “Veteran” Yogyakarta. Dalam penyusunan tugas akhir ini tentu saja penulis mendapatkan bantuan, bimbingan serta dukungan oleh beberapa pihak. Oleh karena itu, penulis ingin mengucapkan terima kasih kepada :

1. Allah SWT yang senantiasa memberikan petunjuk dan kemudahan.
2. Mamah, Bapak, Kakak-kakakku dan Adik-adikku yang telah memberikan do'a dan dukungan agar lancar dalam mengerjakan tugas akhir serta keponakanku tercinta Anindya, Marzia, Rafanita.
3. Bapak Rifky Indra Perwira, S.Kom., M.Eng selaku dosen pembimbing I yang telah memberikan waktu dan kesabaran dalam membantu dan membimbing penulis.
4. Bapak Mangaras Yanu Florestiyanto, S.T., M.Eng selaku dosen pembimbing II dan sekaligus dosen wali yang telah memberikan waktu, dukungan, kesabaran, saran dan kemudahan bagi penulis dalam menyelesaikan tugas akhir.
5. Bapak Oliver Samuel S., S.Kom., M.Eng. dan Bapak Budi Santosa, S.Si., M.T. selaku dosen penguji yang telah memberikan bimbingan dan sarannya.
6. Ibu Hafsa, S.Si., M.T selaku dosen wali.
7. Bapak Kintaka terima kasih atas bantuan dan kemudahannya.
8. Seluruh dosen dan staff tata usaha jurusan teknik informatika yang telah membantu selama perkuliahan hingga penyusunan tugas akhir.
9. Teman-teman teknik informatika 2016 yang telah membantu penulis selama masa perkuliahan.
10. Semua rekan-rekan yang telah membantu dan mendo'akan penulis dalam menyelesaikan tugas akhir ini yang tidak bisa disebutkan satu-persatu.

Akhir kata, semoga Allah SWT memberikan balasan yang setimpal untuk kebaikan kalian semua. Semoga tugas akhir ini dapat memberikan manfaat, dan pengetahuan sehingga dapat menambah informasi bagi para pembaca. Penulis juga ingin mengucapkan maaf apabila masih banyak terdapat kekurangan dalam penulisan ini.

Yogyakarta, 16 Juli 2021

Penulis

DAFTAR ISI

| | |
|-----------------------------------------------------------|-------|
| HALAMAN JUDUL | i |
| HALAMAN PENGESAHAN PEMBIMBING | ii |
| HALAMAN PENGESAHAN PENGUJI..... | iii |
| SURAT PERNYATAAN | iv |
| PERNYATAAN BEBAS PLAGIAT | v |
| ABSTRAK | vi |
| ABSTRACT | vii |
| KATA PENGANTAR..... | viii |
| DAFTAR ISI | ix |
| DAFTAR GAMBAR..... | xiii |
| DAFTAR TABEL | xvi |
| DAFTAR LAMPIRAN | xviii |
| BAB I..... | 1 |
| PENDAHULUAN | 1 |
| 1.1 Latar Belakang Masalah | 1 |
| 1.2 Rumusan Masalah..... | 2 |
| 1.3 Batasan Masalah | 2 |
| 1.4 Tujuan Penelitian | 3 |
| 1.5 Manfaat Penelitian | 3 |
| 1.6 Metodologi Penelitian dan Pengembangan Sistem..... | 3 |
| 1.6.1 Metodologi penelitian..... | 3 |
| 1.6.2 Metodologi pengembangan sistem | 4 |
| 1.7 Sistematika Penulisan | 4 |
| BAB II | 5 |
| TINJAUAN LITERATUR | 5 |
| 2.1 <i>Google Maps</i> | 5 |
| 2.2 <i>Web Scraping</i> | 5 |
| 2.3 Analisis Sentimen | 5 |
| 2.4 <i>Text Mining</i> | 6 |
| 2.5 <i>Text Preprocessing</i> | 6 |
| 2.5.1 <i>Case folding</i> | 6 |
| 2.5.2 <i>Remove punctuation</i> | 6 |
| 2.5.3 <i>Remove number</i> | 7 |
| 2.5.4 <i>Remove repeated characters</i> | 7 |

| | | |
|----------------------------------------------------|----------------------------------------------------------------|----|
| 2.5.5 | <i>Spelling normalization</i> | 7 |
| 2.5.6 | <i>Tokenizing</i> | 7 |
| 2.5.7 | <i>Filtering</i> | 8 |
| 2.5.8 | <i>Stemming</i> | 8 |
| 2.6 | Ekstraksi Fitur (<i>Term Weighting TF-IDF</i>)..... | 10 |
| 2.6.1 | <i>Term frequency</i> (TF)..... | 10 |
| 2.6.2 | <i>Invers document frequency</i> (IDF)..... | 11 |
| 2.6.3 | <i>Term frequency-invers document frequency</i> (TF-IDF) | 11 |
| 2.7 | Seleksi Fitur | 11 |
| 2.7.1 | <i>Chi square</i> | 11 |
| 2.7.2 | <i>Mutual information</i> | 12 |
| 2.8 | <i>Support Vector Machine</i> | 13 |
| 2.9 | <i>Confusion Matrix</i> | 14 |
| 2.10 | <i>K-Fold Cross Validation</i> | 15 |
| 2.11 | Pembobotan Top Rekomendasi | 15 |
| 2.12 | Penelitian Sebelumnya..... | 16 |
| BAB III | | 23 |
| METODOLOGI PENELITIAN DAN PENGEMBANGAN SISTEM..... | | 23 |
| 3.1 | Metodologi Penelitian..... | 23 |
| 3.1.1 | Pengumpulan data | 23 |
| 3.1.1.1 | <i>Studi literatur</i> | 24 |
| 3.1.1.2 | <i>Web scraping</i> | 24 |
| 3.1.2 | <i>Data labeling</i> | 24 |
| 3.1.3 | <i>Text preprocessing</i> | 25 |
| 3.1.3.1 | <i>Case folding</i> | 25 |
| 3.1.3.2 | <i>Remove punctuation</i> | 26 |
| 3.1.3.3 | <i>Remove number</i> | 26 |
| 3.1.3.4 | <i>Remove repeated characters</i> | 27 |
| 3.1.3.5 | <i>Spelling normalization</i> | 27 |
| 3.1.3.6 | <i>Tokenizing</i> | 28 |
| 3.1.3.7 | <i>Filtering</i> | 28 |
| 3.1.3.8 | <i>Stemming</i> | 28 |
| 3.1.4 | Ekstraksi fitur (<i>term weighting TF-IDF</i>) | 29 |
| 3.1.5 | Seleksi fitur..... | 32 |

| | | |
|---------------------------------|------------------------------------------------------------------------------------|-----|
| 3.1.5.1 | <i>Chi square</i> | 33 |
| 3.1.5.2 | <i>Mutual information</i> | 37 |
| 3.1.6 | <i>Support vector machine</i> | 43 |
| 3.1.6.1 | <i>Support vector machine</i> dengan seleksi fitur <i>chi square</i> | 43 |
| 3.1.6.2 | <i>Support vector machine</i> dengan seleksi fitur <i>mutual information</i> | 49 |
| 3.1.7 | Validasi dan pengujian model | 55 |
| 3.1.7.1 | <i>Confusion matrix</i> | 55 |
| 3.1.7.2 | <i>K-fold cross validation</i> | 55 |
| 3.2 | Metodologi pengembangan sistem | 56 |
| 3.2.1 | Pengumpulan kebutuhan sistem | 57 |
| 3.2.1.1 | Pengumpulan data | 57 |
| 3.2.1.2 | Analisis kebutuhan data <i>input</i> | 58 |
| 3.2.1.3 | Analisis kebutuhan data proses | 58 |
| 3.2.1.4 | Analisis kebutuhan data <i>output</i> | 59 |
| 3.2.1.5 | Analisis kebutuhan perangkat keras dan perangkat lunak | 59 |
| 3.2.2 | Perancangan sistem..... | 60 |
| 3.2.2.1 | Arsitektur sistem | 60 |
| 3.2.2.2 | Perancangan proses | 61 |
| 3.2.2.3 | Perancangan basis data..... | 79 |
| 3.2.2.4 | Perancangan antarmuka..... | 83 |
| 3.2.3 | Pengujian sistem | 97 |
| BAB IV | | 99 |
| HASIL, PENGUJIAN DAN PEMBAHASAN | | 99 |
| 4.1. | Hasil Penelitian | 99 |
| 4.1.1 | Implementasi analisis sentimen ulasan di <i>google maps</i> | 99 |
| 4.1.1.1 | Modul <i>web scraping</i> | 99 |
| 4.1.1.2 | Modul text preprocessing | 100 |
| 4.1.1.3 | Modul <i>Data labeling</i> | 104 |
| 4.1.1.4 | Modul ekstraksi fitur (<i>term weighting TF-IDF</i>)..... | 104 |
| 4.1.1.5 | Modul seleksi fitur | 104 |
| 4.1.1.6 | Modul <i>support vector machine</i> | 106 |
| 4.1.2 | Implementasi sistem | 110 |
| 4.1.2.1 | Implementasi Halaman admin..... | 110 |
| 4.1.2.2 | Implementasi Halaman <i>user</i> | 122 |

| | |
|----------------------------------------------|-----|
| 4.2. Pengujian | 128 |
| 4.2.1 Pengujian model | 128 |
| 4.2.1.1 <i>Confusion matrix</i> | 128 |
| 4.2.1.2 <i>K-fold cross validation</i> | 129 |
| 4.2.2 Pengujian sistem | 130 |
| 4.3. Pembahasan | 131 |
| BAB V | 133 |
| PENUTUP | 133 |
| 5.1 Kesimpulan | 133 |
| 5.2 Saran | 133 |
| DAFTAR PUSTAKA..... | 134 |

DAFTAR GAMBAR

| | |
|--------------------------------------------------------------------------------------------------------------------------------|----|
| Gambar 2.1. SVM Berusaha Menemukan <i>Hyperplane</i> Terbaik Yang Memisahkan Kedua Class –1 Dan +1 (Nugroho et al., 2003)..... | 13 |
| Gambar 3.1 Metodologi Penelitian..... | 23 |
| Gambar 3.2. Metodologi Pengembangan Sistem | 57 |
| Gambar 3.3. Arsitektur Sistem | 60 |
| Gambar 3.4. <i>Data Flow Diagram Level 0</i> | 61 |
| Gambar 3.5. <i>Data Flow Diagram Level 1</i> | 62 |
| Gambar 3.6. <i>Data Flow Diagram Level 2</i> | 64 |
| Gambar 3.7. <i>Flowchart Analisis Sentimen</i> | 65 |
| Gambar 3.8. <i>Flowchart Text Preprocessing</i> | 66 |
| Gambar 3.9. <i>Flowchart Case Folding</i> | 67 |
| Gambar 3.10. <i>Flowchart Remove Punctuation</i> | 68 |
| Gambar 3.11. <i>Flowchart Remove Number</i> | 69 |
| Gambar 3.12. <i>Flowchart Remove Repeated Characters</i> | 70 |
| Gambar 3.13. <i>Flowchart Spelling Normalization</i> | 71 |
| Gambar 3.14. <i>Flowchart Tokenizing</i> | 72 |
| Gambar 3.15. <i>Flowchart Filtering</i> | 73 |
| Gambar 3.16. <i>Flowchart Stemming</i> | 74 |
| Gambar 3.17. <i>Flowchart Ekstraksi Fitur (Term Weighting TF-IDF)</i> | 75 |
| Gambar 3.18. <i>Flowchart Seleksi Fitur Chi Square</i> | 76 |
| Gambar 3.19. <i>Flowchart Seleksi Fitur Mutual Information</i> | 77 |
| Gambar 3.20. <i>Flowchart Tahap Pelatihan Support Vector Machine</i> | 78 |
| Gambar 3.21. <i>Flowchart Tahap Pengujian Support Vector Machine</i> | 78 |
| Gambar 3.22. <i>Entity Relationship Diagram</i> | 79 |
| Gambar 3.23. Relasi Antar Tabel | 83 |
| Gambar 3.24. Perancangan Antarmuka Halaman <i>Login</i> | 84 |
| Gambar 3.25. Perancangan Antarmuka Halaman <i>Dashboard</i> | 84 |
| Gambar 3.26. Perancangan Antarmuka Halaman <i>About</i> | 85 |
| Gambar 3.27. Perancangan Antarmuka Halaman Data Wisata | 85 |
| Gambar 3.28. Perancangan Antarmuka Tombol Tambah Data Wisata | 86 |
| Gambar 3.29. Perancangan Antarmuka Tombol Detail Data Wisata..... | 86 |
| Gambar 3.30. Perancangan Antarmuka Tombol Edit Data Wisata..... | 87 |
| Gambar 3.31. Perancangan Antarmuka Halaman Data Kategori Wisata..... | 87 |
| Gambar 3.32. Perancangan Antarmuka Tombol Tambah Data Kategori Wisata | 88 |
| Gambar 3.33. Perancangan Antarmuka Tombol Edit Data Kategori Wisata | 88 |
| Gambar 3.34. Perancangan Antarmuka Halaman Data <i>Training</i> | 89 |
| Gambar 3.35. Perancangan Antarmuka Tombol Detail Data <i>Training</i> | 89 |
| Gambar 3.36. Perancangan Antarmuka Halaman Data Sentimen | 90 |
| Gambar 3.37. Perancangan Antarmuka Tombol Tambah Data Sentimen | 90 |
| Gambar 3.38. Perancangan Antarmuka Halaman Analisis | 91 |
| Gambar 3.39. Perancangan Antarmuka Halaman Riwayat Analisis | 91 |
| Gambar 3.40. Perancangan Antarmuka Tombol Detail Riwayat Analisis | 92 |
| Gambar 3.41. Perancangan Antarmuka Halaman Grafik Seleksi Fitur..... | 92 |
| Gambar 3.42. Perancangan Antarmuka Halaman Grafik Sentimen..... | 93 |
| Gambar 3.43. Perancangan Antarmuka Halaman Grafik Pengujian | 93 |

| | |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----|
| Gambar 3.44. Perancangan Antarmuka Halaman Grafik Pengujian Rata-Rata | 94 |
| Gambar 3.45. Perancangan Antarmuka Halaman <i>Home</i> | 94 |
| Gambar 3.46 Perancangan Antarmuka Halaman Top Rekomendasi | 95 |
| Gambar 3.47. Perancangan Antarmuka Halaman Destinasi Wisata | 95 |
| Gambar 3.48. Perancangan Antarmuka Halaman <i>View Details</i> Wisata..... | 96 |
| Gambar 3.49. Perancangan Antarmuka <i>Link Detail</i> Deskripsi Wisata | 96 |
| Gambar 3.50. Perancangan Antarmuka Halaman <i>About</i> | 97 |
| Gambar 3.51. Perancangan Antarmuka Halaman <i>Contact</i> | 97 |
| Gambar 4.1. Implementasi <i>User Interface</i> Pada Halaman <i>Login</i> | 110 |
| Gambar 4.2. Implementasi <i>User Interface</i> Pada Halaman <i>Dashboard</i> | 111 |
| Gambar 4.3. Implementasi <i>User Interface</i> Pada Halaman <i>About</i> | 111 |
| Gambar 4.4. Implementasi <i>User Interface</i> Pada Halaman Data Wisata | 112 |
| Gambar 4.5. Implementasi <i>User Interface</i> Pada Tombol Tambah Data Wisata | 112 |
| Gambar 4.6. Implementasi <i>User Interface</i> Pada Tombol Detail Data Wisata | 113 |
| Gambar 4.7. Implementasi <i>User Interface</i> Pada Tombol Edit Data Wisata | 113 |
| Gambar 4.8. Implementasi <i>User Interface</i> Pada Tombol <i>Delete</i> Data Wisata..... | 114 |
| Gambar 4.9. Implementasi <i>User Interface</i> Pada Halaman Data Kategori Wisata | 114 |
| Gambar 4.10. Implementasi <i>User Interface</i> Pada Tombol Tambah Data Kategori Wisata | 115 |
| Gambar 4.11. Implementasi <i>User Interface</i> Pada Tombol Edit Data Kategori Wisata.... | 115 |
| Gambar 4.12. Implementasi <i>User Interface</i> Pada Tombol <i>Delete</i> Data Kategori Wisata . | 116 |
| Gambar 4.13. Implementasi <i>User Interface</i> Pada Halaman Data <i>Training</i> | 116 |
| Gambar 4.14. Implementasi <i>User Interface</i> Pada Tombol Detail Data <i>Training</i> | 117 |
| Gambar 4.15. Implementasi <i>User Interface</i> Pada Halaman Data Sentimen | 117 |
| Gambar 4.16. Implementasi <i>User Interface</i> Pada Tombol Tambah Data Sentimen | 118 |
| Gambar 4.17. Implementasi <i>User Interface</i> Pada Halaman Analisis | 118 |
| Gambar 4.18. Implementasi <i>User Interface</i> Pada Halaman Riwayat Analisis..... | 119 |
| Gambar 4.19. Implementasi <i>User Interface</i> Pada Tombol Detail Riwayat Analisis..... | 119 |
| Gambar 4.20. Implementasi <i>User Interface</i> Pada Halaman Grafik Seleksi Fitur | 120 |
| Gambar 4.21. Implementasi <i>User Interface</i> Pada Halaman Grafik Sentimen..... | 121 |
| Gambar 4.22. Implementasi <i>User Interface</i> Pada Halaman Grafik Pengujian..... | 121 |
| Gambar 4.23. Implementasi <i>User Interface</i> Pada Halaman Grafik Pengujian Rata-Rata. | 122 |
| Gambar 4.24. Implementasi <i>User Interface</i> Pada Halaman <i>Home</i> | 122 |
| Gambar 4.25. Implementasi <i>User Interface</i> Pada Menu Menuju Ke Halaman Destinasi Wisata | 123 |
| Gambar 4.26. Implementasi <i>User Interface</i> Pada Panduan Menggunakan Sistem Ini Untuk User Yang Ingin Mencari Top Rekomendasi | 123 |
| Gambar 4.27. Implementasi <i>User Interface</i> Pada Menu Menuju Ke Halaman Top Rekomendasi | 124 |
| Gambar 4.28. Implementasi <i>User Interface</i> Pada Halaman Top Rekomendasi Semua Kategori Wisata | 124 |
| Gambar 4.29. Implementasi <i>User Interface</i> Pada Halaman Top Rekomendasi Berdasarkan Kategori Wisata Yang Dipilih | 125 |
| Gambar 4.30. Implementasi <i>User Interface</i> Pada Halaman Top Rekomendasi Berdasarkan Kata Kunci Yang Diketikkan Pada Menu Pencarian Yang Berada Pada Halaman <i>Home</i> | 125 |
| Gambar 4.31. Implementasi <i>User Interface</i> Pada Halaman Destinasi Wisata | 126 |

| | |
|----------------------------------------------------------------------------------------------------|-----|
| Gambar 4.32. Implementasi <i>User Interface</i> Pada Halaman <i>View Details</i> Wisata | 126 |
| Gambar 4.33. Implementasi <i>User Interface</i> Pada <i>Link More</i> Detail Deskripsi Wisata..... | 127 |
| Gambar 4.34. Implementasi <i>User Interface</i> Pada Halaman <i>About</i> | 127 |
| Gambar 4.35. Implementasi <i>User Interface</i> Pada Halaman <i>Contact</i> | 128 |

DAFTAR TABEL

| | |
|---------------------------------------------------------------------------------------------------------|----|
| Tabel 2.1. Contoh <i>Case Folding</i> | 6 |
| Tabel 2.2. Contoh <i>Remove Punctuation</i> | 6 |
| Tabel 2.3. Contoh <i>Remove Number</i> | 7 |
| Tabel 2.4. Contoh <i>Remove Repeated Characters</i> | 7 |
| Tabel 2.5. Contoh <i>Spelling Normalization</i> | 7 |
| Tabel 2.6. Contoh <i>Tokenizing</i> | 7 |
| Tabel 2.7. Contoh <i>Filtering</i> | 8 |
| Tabel 2.8. Kombinasi Awalan-Akhiran Yang Tidak Diperbolehkan..... | 9 |
| Tabel 2.9. Aturan Pemenggalan Kata Dasar | 9 |
| Tabel 2.10. Contoh <i>Stemming</i> Nazief dan Adriani..... | 10 |
| Tabel 2.11. Kontingensi <i>term</i> atau kata..... | 12 |
| Tabel 2.12. <i>Confusion Matrix</i> nilai jumlah kemunculan <i>term</i> atau kata | 12 |
| Tabel 2.13 <i>Confusion Matrix</i> | 14 |
| Tabel 2.14 <i>State of the Art</i> | 16 |
| Tabel 2.15 Lanjutan <i>State of the Art</i> | 17 |
| Tabel 2.16 Lanjutan <i>State of the Art</i> | 18 |
| Tabel 2.17 Lanjutan <i>State of the Art</i> | 19 |
| Tabel 2.18 Lanjutan <i>State of the Art</i> | 20 |
| Tabel 2.19 Lanjutan <i>State of the Art</i> | 21 |
| Tabel 2.20 Lanjutan <i>State of the Art</i> | 22 |
| Tabel 3.1. Contoh Data Ulasan..... | 24 |
| Tabel 3.2. Contoh Data Ulasan Yang Sudah Diberi Label..... | 25 |
| Tabel 3.3. Contoh Hasil <i>Text Preprocessing</i> | 25 |
| Tabel 3.4. Contoh Hasil <i>Case Folding</i> | 25 |
| Tabel 3.5. Lanjutan Contoh Hasil <i>Case Folding</i> | 26 |
| Tabel 3.6. Contoh Hasil <i>Remove Punctuation</i> | 26 |
| Tabel 3.7. Contoh Hasil <i>Remove Number</i> | 26 |
| Tabel 3.8. Lanjutan Contoh Hasil <i>Remove Number</i> | 27 |
| Tabel 3.9. Contoh Hasil <i>Remove Repeated Characters</i> | 27 |
| Tabel 3.10. Contoh Hasil <i>Spelling Normalization</i> | 27 |
| Tabel 3.11. Contoh Hasil <i>Tokenizing</i> | 28 |
| Tabel 3.12. Contoh Hasil <i>Filtering</i> | 28 |
| Tabel 3.13. Contoh Hasil <i>Stemming</i> | 28 |
| Tabel 3.14. Lanjutan Contoh Hasil <i>Stemming</i> | 29 |
| Tabel 3.15. Hasil Perhitungan TF, DF, IDF Semua Fitur | 30 |
| Tabel 3.16. Lanjutan Hasil Perhitungan TF, DF, IDF Semua Fitur | 31 |
| Tabel 3.17. Hasil Perhitungan TF-IDF Semua Fitur | 31 |
| Tabel 3.18. Lanjutan Hasil Perhitungan TF-IDF Semua Fitur | 32 |
| Tabel 3.53. <i>Confusion Matrix</i> Metode SVM dengan <i>Chi Square</i> | 55 |
| Tabel 3.54. <i>Confusion Matrix</i> Metode SVM dengan <i>Mutual Information</i> | 55 |
| Tabel 3.55. Pengujian <i>K-Fold Cross Validation</i> Metode SVM dengan <i>Chi Square</i> | 55 |
| Tabel 3.56. Lanjutan Pengujian <i>K-Fold Cross Validation</i> Metode SVM dengan <i>Chi Square</i> | 56 |
| Tabel 3.57. Pengujian <i>K-Fold Cross Validation</i> Metode SVM dengan <i>Mutual Information</i> | 56 |

| | |
|---------------------------------------------------------------------------------------------------------------------|-----|
| Tabel 3.58. Perbandingan Hasil Rata-Rata Pengujian <i>K-Fold Cross Validation</i> | 56 |
| Tabel 3.59. Kebutuhan Perangkat Keras | 59 |
| Tabel 3.60. Kebutuhan Perangkat Lunak | 59 |
| Tabel 3.61. Struktur Tabel Admin..... | 80 |
| Tabel 3.62. Struktur Tabel Wisata..... | 80 |
| Tabel 3.63. Struktur Tabel Kategori Wisata..... | 80 |
| Tabel 3.64. Struktur Tabel <i>Training</i> | 81 |
| Tabel 3.65. Struktur Tabel Sentimen..... | 81 |
| Tabel 3.66. Struktur Tabel Analisis Ulasan..... | 81 |
| Tabel 3.67. Lanjutan Struktur Tabel Analisis Ulasan | 82 |
| Tabel 3.68. Struktur Tabel <i>Chisquare</i> | 82 |
| Tabel 3.69. Struktur Tabel MI..... | 82 |
| Tabel 3.70. Struktur Tabel Pengujian..... | 82 |
| Tabel 3.71. Pengujian <i>Black Box</i> | 97 |
| Tabel 3.72. Lanjutan Pengujian <i>Black Box</i> | 98 |
| Tabel 4.1. <i>Confusion Matrix</i> Metode SVM dengan seleksi fitur <i>Chi Square</i> | 129 |
| Tabel 4.2. <i>Confusion Matrix</i> Metode SVM dengan seleksi fitur <i>Mutual Information</i> | 129 |
| Tabel 4.3. Pengujian <i>K-Fold Cross Validation</i> Metode SVM dengan seleksi fitur <i>Chi Square</i> | 129 |
| Tabel 4.4. Pengujian <i>K-Fold Cross Validation</i> Metode SVM dengan seleksi fitur <i>Mutual Information</i> | 130 |
| Tabel 4.5. Perbandingan Hasil Rata-Rata Pengujian <i>K-Fold Cross Validation</i> | 130 |
| Tabel 4.6. Pengujian <i>Black Box</i> | 130 |
| Tabel 4.7. Lanjutan Pengujian <i>Black Box</i> | 131 |

DAFTAR LAMPIRAN

| | |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----|
| Lampiran A Tabel Menentukan Banyaknya Fitur Yang Akan Digunakan Berdasarkan Hasil Rata-Rata Akurasi Tertinggi Pengujian Penggunaan 100 Fitur Sampai 2650 Fitur (Semua Fitur)..... | 139 |
| Lampiran B Contoh Data Labeling Ulasan | 143 |

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Analisis sentimen adalah mengekstraksi pendapat, sentimen, evaluasi, dan emosi orang tentang suatu topik tertentu yang tertulis menggunakan teknik pemrosesan bahasa alami (Sentiaji & Bachtiar, 2014). Analisis sentimen menggunakan algoritma *machine learning* untuk proses klasifikasinya. Algoritma *machine learning* yang sering kali digunakan antara lain *Support Vector Machine*, *Naïve Bayes*, *Decision Tree* dan *K-Nearest Neighbor*. Dalam beberapa penelitian sebelumnya, yang dilakukan oleh (Nurajijah & Riana, 2019) tentang Algoritma *Naïve Bayes*, *Decision Tree*, dan SVM untuk Klasifikasi Persetujuan Pembiayaan Nasabah Koperasi Syariah, terbukti bahwa algoritma SVM lebih tinggi dibandingkan dengan algoritma *Naïve Bayes* dan *Decision Tree* dengan hasil akurasi sebesar 89.86% sedangkan algoritma *Naïve Bayes* hanya memperoleh hasil akurasi sebesar 77.29% dan algoritma *Decision Tree* sebesar 89.02%. Pada penelitian yang dilakukan oleh (Sutoyo & Almaarif, 2020) tentang *Twitter Sentiment Analysis of the Relocation of Indonesia's Capital City*, terbukti bahwa algoritma SVM mendapatkan hasil akurasi, presisi, *recall* dan *f-measure* paling unggul sebesar 97.72%, 96.01%, 99.18%, dan 97.57% dibandingkan dengan algoritma *Naïve Bayes*, *Logistic Regression* dan *K-Nearest Neighbor*. Jadi, pada penelitian ini akan menggunakan algoritma *Support Vector Machine* untuk klasifikasi analisis sentimen penilaian positif dan negatif pada fitur ulasan di *Google Maps* mengenai tempat pariwisata di Kota Yogyakarta. Akan tetapi, *Support Vector Machine* memiliki kekurangan dalam pemilihan fitur yang sesuai. Pemilihan fitur ini dapat mempengaruhi hasil akurasi karena tidak semua fitur sama pentingnya. Maka diperlukan seleksi fitur untuk menghilangkan data yang belebihan, tidak relevan, dan menghilangkan data yang *noise* sehingga dapat meningkatkan hasil akurasi pada algoritma *Support Vector Machine*.

Beberapa seleksi fitur yang dapat digunakan antara lain *Mutual Information* dan *Chi Square*. Seleksi fitur *Mutual Information* merupakan sebuah metode pengukuran kuantitatif dari jumlah ketergantungan (informasi) antara dua variabel acak (Sulaiman & Labadin, 2015). Berdasarkan penelitian yang dilakukan oleh (Irham et al., 2019) tentang Klasifikasi Berita Bahasa Indonesia menggunakan *Mutual Information* dan *Support Vector Machine*, terbukti bahwa metode klasifikasi *Support Vector Machine* yang menggunakan seleksi fitur *Mutual Information* memiliki hasil akurasi lebih baik dibandingkan dengan metode *Support Vector Machine* tanpa seleksi fitur *Mutual Information* dengan hasil akurasi sebesar 93.98%. Pada penelitian ini, penggunaan seleksi fitur *Mutual Information* dapat menyeleksi fitur-fitur yang tidak menggambarkan sebuah kelas, sehingga dapat meningkatkan performansi metode klasifikasi *Support Vector Machine* agar lebih cepat dan efektif karena tidak perlu menggunakan semua fitur yang ada. Seleksi fitur *Chi Square* merupakan sebuah metode yang menggunakan teori statistika untuk menguji independensi sebuah *term* dengan kategorinya (Ling, Kencana, & Oka, 2014). Dalam penelitian yang dilakukan oleh (Ukhti Ikhsani Larasati et al., 2019) tentang *Improve the Accuracy of Support Vector Machine Using Chi Square Statistic and Term Frequency Inverse Document Frequency on Movie Review Sentiment Analysis*, membahas mengenai pengaruh penerapan seleksi fitur *Chi*

Chi Square dan pembobotan TF-IDF untuk meningkatkan akurasi metode klasifikasi *Support Vector Machine* dalam analisis sentimen ulasan film pada situs *Internet Movie Database* (IMDB). Dari hasil penelitian menunjukkan bahwa penerapan seleksi fitur *Chi Square* dan Pembobotan TF-IDF dapat menaikkan hasil akurasi menjadi 80.2% dari hasil akurasi sebelumnya sebesar 68.7% yang hanya menggunakan metode *Support Vector Machine*. Berdasarkan beberapa penelitian sebelumnya, dapat ditarik kesimpulan bahwa penggunaan seleksi fitur *Mutual Information* dan *Chi Square* dapat mengatasi permasalahan pemilihan fitur pada metode *Support Vector Machine*. Keduanya, terbukti dapat meningkatkan tingkat akurasi metode *Support Vector Machine*. Akan tetapi, belum diketahui perbandingan tingkat akurasi dari seleksi fitur *Mutual Information* dan *Chi Square* pada metode *Support Vector Machine* untuk permasalahan yang sama.

Maka pada penelitian ini, akan membandingkan hasil akurasi dari metode klasifikasi *Support Vector Machine* menggunakan seleksi fitur *Mutual Information* dan metode klasifikasi *Support Vector Machine* menggunakan seleksi fitur *Chi Square* dalam menganalisis sentimen tempat pariwisata di Kota Yogyakarta pada ulasan *google maps*. Sehingga, dapat diketahui seleksi fitur mana yang memiliki hasil akurasi paling tinggi jika diterapkan pada metode klasifikasi *Support Vector Machine* untuk permasalahan yang sama. Kemudian, model dari metode klasifikasi *Support Vector Machine* dengan seleksi fitur yang memiliki akurasi paling tinggi akan digunakan untuk menentukan daftar rekomendasi destinasi tempat pariwisata di Kota Yogyakarta.

1.2 Rumusan Masalah

Berdasarkan latar belakang permasalahan yang telah dikemukakan diatas, maka dapat dirumuskan masalah yang ada yaitu :

1. Bagaimana melakukan analisis sentimen terhadap penilaian tempat pariwisata di Kota Yogyakarta berdasarkan data dari ulasan di *Google Maps* sehingga menghasilkan rekomendasi destinasi tempat pariwisata di Kota Yogyakarta.
2. Bagaimana hasil perbandingan akurasi dari seleksi fitur pada analisis sentimen terhadap penilaian tempat pariwisata di Kota Yogyakarta berdasarkan data dari ulasan di *Google Maps*.

1.3 Batasan Masalah

Batasan masalah pada analisis sentimen ulasan di *Google Maps* menggunakan metode *Support Vector Machine* dengan seleksi fitur *Mutual Information* dan *Chi Square* untuk rekomendasi destinasi tempat pariwisata di Kota Yogyakarta, yaitu :

1. Data yang digunakan untuk analisis sentimen adalah ulasan dari *Google Maps* yang diambil pada bulan agustus sampai oktober 2020.
2. Data yang akan dianalisis sentimen hanya yang berbahasa Indonesia.
3. Data analisis sentimen dikategorikan dalam sentimen positif dan sentimen negatif.
4. Data daftar pariwisata di Kota Yogyakarta yang akan dianalisis sentimen diambil dari situs *visitingjogja.com* (situs resmi Dinas Pariwisata Daerah Istimewa Yogyakarta).
5. Pariwisata yang akan dianalisis minimal memiliki 100 ulasan di *Google Maps*.
6. Data hasil analisis sentimen dari seleksi fitur yang memiliki akurasi terbaik akan digunakan untuk pembobotan daftar rekomendasi pariwisata di Kota Yogyakarta.

1.4 Tujuan Penelitian

Tujuan dari pembuatan sistem analisis sentimen ulasan di *Google Maps* menggunakan metode *Support Vector Machine* dengan seleksi fitur *Mutual Information* dan *Chi Square* untuk rekomendasi destinasi tempat pariwisata di Kota Yogyakarta, yaitu :

1. Mengimplementasikan metode *Support Vector Machine* untuk mengklasifikasikan penilaian tempat pariwisata ke dalam sentimen positif atau sentimen negatif berdasarkan dari data ulasan di *Google Maps*.
2. Mengetahui perbandingan akurasi antara metode *Support Vector Machine* dengan seleksi fitur *Mutual Information* dan metode *Support Vector Machine* dengan seleksi fitur *Chi Square*.
3. Memberikan informasi atau rekomendasi mengenai destinasi tempat pariwisata di Kota Yogyakarta berdasarkan analisis sentimen positif dari data ulasan di *Google Maps*.

1.5 Manfaat Penelitian

Manfaat yang diharapkan dari dibuatnya sistem yang menerapkan analisis sentimen ulasan di *Google Maps* menggunakan metode *Support Vector Machine* dengan seleksi fitur *Mutual Information* dan *Chi Square* untuk rekomendasi destinasi tempat pariwisata di Kota Yogyakarta, yaitu :

1. Dapat menjadi acuan rekomendasi destinasi tempat pariwisata di Kota Yogyakarta untuk wisatawan yang akan berkunjung ke Kota Yogyakarta.
2. Dari hasil akurasi metode *Support Vector Machine* dengan seleksi fitur *Mutual Information* dan metode *Support Vector Machine* dengan seleksi fitur *Chi Square*, maka dapat diketahui seleksi fitur mana yang lebih unggul untuk dikombinasikan dengan metode *Support Vector Machine*, sehingga dapat dijadikan acuan seleksi fitur yang akan digunakan untuk analisis sentimen.
3. Dapat dijadikan bahan referensi untuk penelitian selanjutnya mengenai analisis sentimen dan komparasi seleksi fitur pada metode *Support Vector Machine*.

1.6 Metodologi Penelitian dan Pengembangan Sistem

1.6.1 Metodologi penelitian

Metodologi penelitian yang digunakan pada penelitian ini adalah metodologi penelitian kuantitatif. Metodologi penelitian kuantitatif merupakan metodologi penelitian yang digunakan untuk menjawab permasalahan yang berkaitan dengan data berupa angka dan data kuantitatif secara lebih terperinci menggunakan analisa statistik, meneliti populasi dan sampel tertentu, instrumen penelitian, teknik pengumpulan data, dan analisis data dalam suatu proposal atau laporan penelitian sesuai dengan kaidah penulisan karya ilmiah yang disyaratkan (Harahap & Lubis, 2019). Metodologi penelitian pada penelitian ini meliputi beberapa tahapan, antara lain :

1. Pengumpulan Data
2. *Data Labelling*
3. *Text Preprocessing*
4. Ekstraksi Fitur (*Term Weighting TF-IDF*)
5. Seleksi Fitur (*Chi Square* dan *Mutual Information*)
6. *Support Vector Machine* (SVM)

7. Pengujian Model

1.6.2 Metodologi pengembangan sistem

Metode pengembangan sistem yang digunakan pada penelitian ini adalah metode *prototype*. *Prototype* merupakan metode yang paling banyak digunakan, yang dibuat secara terstruktur yang melalui beberapa tahapan, jika didalam tahap akhir sistem masih memiliki kekurangan, maka akan dilakukan evaluasi atau perbaikan kembali dari awal. Menurut Ogedebe dan Jacob (2012), tahapan-tahapan metode *prototype* antara lain:

1. Pengumpulan Kebutuhan Sistem
2. Proses Perancangan Yang Cepat
3. Membangun *Prototype*
4. Evaluasi Dan Perbaikan

1.7 Sistematika Penulisan

Sistematika penulisan pada Tugas Akhir ini akan dijelaskan sebagai berikut :

Bab I Pendahuluan

Pada bab pendahuluan akan menjelaskan tentang latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, metodologi penelitian dan pengembangan sistem serta sistematika penulisan.

Bab II Tinjauan Literatur

Pada bab ini menjelaskan tentang teori-teori dasar yang berhubungan dengan penelitian Tugas Akhir. Serta berisi tentang analisis perbandingan penelitian yang akan dilakukan dengan penelitian yang sudah ada.

Bab III Metodologi Penelitian dan Pengembangan Sistem

Pada bab ini membahas tentang metodologi penelitian dan pengembangan sistem yang digunakan.

Bab IV Hasil, Pengujian dan Pembahasan

Pada bab ini menjelaskan tentang implementasi, hasil dan pembahasan pada bab sebelumnya serta pengujian pada penelitian ini untuk proses validasi.

Bab V Penutup

Pada bab ini berisi kesimpulan dari Tugas Akhir yang telah dilakukan dan menyebutkan saran-saran agar penelitian selanjutnya dapat melakukan pengembangan lebih lanjut tentang penelitian ini.

BAB II

TINJAUAN LITERATUR

2.1 Google Maps

Google maps merupakan *service* atau layanan peta *online* dari *Google* yang dapat diakses secara gratis oleh semua orang. *Google Maps* memiliki beberapa fitur yang bermanfaat bagi semua orang, salah satunya fitur *Google review* atau ulasan mengenai suatu tempat wisata. Fitur *Google review* digunakan pengguna *Google Maps* untuk memberikan komentar baik positif maupun negatif mengenai suatu tempat wisata berupa teks dan *rating* berdasarkan kepuasan pengunjung, pengguna juga membagikan foto mengenai tempat wisata tersebut. Berdasarkan penelitian yang dilakukan Haq (Haq, 2020), ditemukan bahwa di era big data ini, dimana data hadir dari setiap pengguna, *Google review* dapat dimanfaatkan secara gratis dan sebagai alat untuk membangun citra dan reputasi sebuah destinasi tempat wisata. Penilaian ulasan dari pengguna dalam fitur *Google review*, dapat memberikan informasi kepada wisatawan yang akan datang mengenai seberapa besar pemeringkatan tempat wisata yang diberikan oleh wisatawan yang pernah berkunjung.

2.2 Web Scraping

Web scraping merupakan sekumpulan teknik yang digunakan untuk secara otomatis mendapatkan beberapa informasi dari situs web alih-alih menyalinnya secara manual (Vargiu & Urru, 2013). Dengan menulis program otomatis sederhana, dapat melakukan *query server web*, meminta data, dan menguraikannya untuk mengekstrak informasi yang dibutuhkan (Mitchell, 2018). Pendapat lain menyatakan *web scraping* merupakan proses pengambilan sebuah dokumen semi-terstruktur dari internet, yang pada umumnya berupa halaman *web* dengan menggunakan *markup language* (HTML/XHTML), yang nantinya dokumen tersebut akan dianalisis dan diambil data tertentu, untuk digunakan dalam kepentingan lain (Turland, 2010). *Web scraping* memiliki beberapa Teknik antara lain, menggunakan *regular expression*, *parsing HTML*, *parsing DOM*, menggunakan XPath, dan menggunakan *Google Sheet*. Berikut ini merupakan langkah-langkah untuk melakukan *web scraping* antara lain :

1. Melakukan *request url* yang dijadikan target.
2. *Request url* yang telah dilakukan akan diproses oleh *server target*.
3. Kemudian, akan diperoleh hasil (data) dari *request url* tersebut berupa teks HTML.
4. Selanjutnya, ekstrak data untuk mengambil teks yang diperlukan.
5. Diperoleh hasil dari ekstrak data atau *output* yang diinginkan.

2.3 Analisis Sentimen

Analisis sentimen dan *opinion mining* adalah bidang studi yang menganalisis opini, sentimen, evaluasi, sikap, dan emosi orang-orang dari bahasa tertulis (Liu & Zhang, 2012). Analisis sentimen atau *opinion mining* dapat digunakan untuk memperoleh gambaran umum persepsi masyarakat terhadap kualitas layanan, apakah cenderung positif, negatif atau netral (Syaifudin & Irawan, 2018). Metode dalam analisis sentimen dibedakan menjadi dua yaitu *machine learning* dan *knowledge-based*. Metode-metode dalam *machine learning* antara lain SVM, *Naïve Bayes*, *Maximum Entropy* dan *Neural Network*. Sedangkan, metode yang termasuk dalam *knowledge-based* antara lain *Lexicon-Based*, PMI (*Pointwise Mutual Information*), *Hybrid Approach*. Tahapan dalam proses analisis sentimen antara lain :

1. *Subjectivity Classification* yaitu tahapan menentukan kalimat yang merupakan opini atau pendapat.

2. *Orientation Detection* yaitu tahapan mengkategorikan sebuah kalimat positif, negatif, atau netral.
3. *Opinion Holder and Target Detection* yaitu tahapan untuk menentukan bagian yang merupakan target atau yang dicari didalam kalimat tersebut.

2.4 Text Mining

Text mining merupakan penambangan sebuah data yang berupa teks dimana sumber data didapatkan dari dokumen yang tujuannya untuk mencari kata-kata yang mewakili isi dari dokumen sehingga dapat dilakukan analisa pada dokumen yang saling berhubungan. *Text mining* digunakan untuk menangani permasalahan seperti klasifikasi, klastering, *information extraction* dan *information retrieval* (Berry & Kogan, 2010). Proses dalam *text mining* sama dengan data *mining* yang membedakan adalah pada data *mining* data yang diolah merupakan data yang terstruktur, sedangkan pada *text mining* data yang diolah adalah data yang semi-terstruktur atau data yang tidak terstruktur. Pada *text mining* data yang semi-terstruktur atau tidak terstruktur tersebut akan diubah menjadi data yang terstruktur. Tahapan dalam *text mining* secara umum adalah *text preprocessing* dan *feature selection* (Feldman & Sanger, 2007).

2.5 Text Preprocessing

Text preprocessing adalah tahapan untuk mempersiapkan teks menjadi data yang akan diolah di tahapan berikutnya (Mustaqhfiri et al., 2012). Dalam *text mining*, *preprocessing* data digunakan untuk mengekstraksi data yang menarik dan berharga dan pengetahuan dari data teks yang tidak terstruktur (Gurusamy & Kannan, 2014). *Text preprocessing* bertujuan untuk mengurangi *noise* atau membersihkan data pada data latih atau data uji sebelum digunakan, dengan cara menyamakan struktur dan mengurangi dimensi yang melebar. Tahapan *text preprocessing* dalam penelitian ini antara lain :

2.5.1 Case folding

Case folding adalah tahapan proses mengubah semua huruf dalam teks dokumen menjadi huruf kecil, serta menghilangkan karakter selain a-z (Mustaqhfiri et al., 2012). *Case folding* diperlukan karena tidak semua data dalam media sosial konsisten untuk menggunakan *lowercase* (huruf kecil). Contoh *case folding* ditunjukkan pada tabel berikut :

Tabel 2.1. Contoh Case Folding

| <i>Input</i> | <i>Output</i> |
|--------------|---------------|
| Indah | indah |

2.5.2 Remove punctuation

Remove punctuation adalah proses untuk menghilangkan tanda baca pada teks dokumen. *Remove Punctuation* berfungsi untuk membersihkan komponen yang tidak berpengaruh terhadap hasil klasifikasi (Negara et al., 2020). Komponen tanda baca yang dihapus antara lain tanda seru, titik, titik koma, *slash*, tanda petik satu, tanda petik dua (kutipan), strip, tanda tanya, tanda kurung, dan sebagainya. Contoh *remove punctuation* ditunjukkan pada tabel berikut :

Tabel 2.2. Contoh Remove Punctuation

| <i>Input</i> | <i>Output</i> |
|--------------------------------------------------------------------|------------------------------------------------------------------|
| bangunan di taman sari sangat indah, pemandunya juga ramah sekali! | bangunan di taman sari sangat indah pemandunya juga ramah sekali |

2.5.3 Remove number

Remove number adalah proses penghapusan karakter angka di dalam teks dokumen. *Remove number* dilakukan karena angka dianggap *delimiter* atau tidak berpengaruh terhadap hasil klasifikasi bahkan dapat mengganggu proses klasifikasi. Contoh *remove number* ditunjukkan pada tabel berikut :

Tabel 2.3. Contoh *Remove Number*

| Input | Output |
|--------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------|
| harga tiket masuk museum kereta keraton cuma 5000 tidak terlalu besar tapi banyak sejarah yang menarik | harga tiket masuk museum kereta keraton tidak terlalu besar tapi banyak sejarah yang menarik |

2.5.4 Remove repeated characters

Remove repeated characters merupakan proses penghapusan yang sama seperti proses *remove punctuation* dan *remove number*, yang membedakan adalah objek yang akan dihapus. *Remove repeated characters* adalah proses penghapus karakter yang berulang didalam teks dokumen. Karakter berulang yang dihapus adalah huruf yang memiliki lebih dari dua karakter, dan akan disisakan dua karakter saja. Contoh *remove repeated characters* ditunjukkan pada tabel berikut :

Tabel 2.4. Contoh *Remove Repeated Characters*

| Input | Output |
|--------------------------------------------|-----------------------------------------|
| kereeeeen sekali museum kereta keraton ini | kereen sekali museum kereta keraton ini |

2.5.5 Spelling normalization

Spelling normalization merupakan proses perbaikan dan substitusi kata-kata yang salah eja ataupun disingkat dengan bentuk tertentu (Simatupang & Utomo, 2019). *Spelling normalization* dilakukan untuk menghindari dimensi kata yang melebar, karena kesalahan ketik pada penulisan atau singkatan, padahal kata tersebut memiliki arti yang sama. Jika tidak dilakukan *spelling normalization* maka kata tersebut akan dianggap sebuah entitas yang berbeda pada penyusunan matriks. Pada penelitian ini menggunakan daftar kamus *spelling normalization* dari penelitian (Noehilasari, 2014) dan ditambah secara manual dari pengamatan data latih yang digunakan pada penelitian ini. Contoh *spelling normalization* ditunjukkan pada tabel berikut :

Tabel 2.5. Contoh *Spelling Normalization*

| Input | Output |
|-------|--------|
| tdk | tidak |
| sya | saya |
| ngga | tidak |
| tpi | tapi |
| aq | aku |
| bgt | banget |

2.5.6 Tokenizing

Tokenizing adalah tahapan pemotongan *string input* menjadi beberapa kata berdasarkan tiap kata yang menyusunnya. Pada tahapan ini, komentar yang didapat dari data media sosial akan dipisah menjadi tiap kata berdasarkan spasi atau karakter lain. Kata-kata yang dihasilkan dari tahapan *tokenizing* ini disebut dengan *token*. Contoh dari *tokenizing* ditunjukkan pada tabel berikut :

Tabel 2.6. Contoh *Tokenizing*

| Input | Output |
|-------------------------------------|--------------------------------------------------------|
| bangunan di taman sari sangat indah | ['bangunan', 'di', 'taman', 'sari', 'sangat', 'indah'] |

2.5.7 Filtering

Filtering adalah tahapan mengambil kata-kata penting yang dihasilkan oleh *token*. Di dalam tahapan *filtering* terdapat algoritma *stopword removal* dan *wordlist*. Pada penelitian ini akan menggunakan algoritma *stopword removal* untuk membuang kata-kata yang tidak penting yang didapat dari hasil token seperti “yang”, “di”, “dari”, “untuk”, “ke”. Kata-kata tersebut merupakan kata yang memiliki frekuensi tinggi kemunculannya pada teks dokumen. Proses *filtering* ini bertujuan untuk mengurangi ukuran indeks, waktu pemrosesan, dan *noise* pada data. Pada penelitian ini menggunakan kamus *filtering* atau *stopword removal* dari (Tala, 2003) Contoh dari tahapan *filtering* atau *stopword removal* ditunjukkan pada tabel berikut :

Tabel 2.7. Contoh *Filtering*

| <i>Input</i> | <i>Output</i> |
|--------------------------------------------------------|----------------------------------------|
| [‘bangunan’, ‘di’, ‘taman’, ‘sari’, ‘sangat’, ‘indah’] | [‘bangunan’, ‘taman’, ‘sari’, ‘indah’] |

2.5.8 Stemming

Stemming adalah tahapan untuk mengubah kata atau token hasil *filtering* menjadi kata dasarnya dengan menghilangkan imbuhan pada kata tersebut. Algoritma *stemming* untuk bahasa Indonesia berbeda dengan bahasa Inggris, karena memiliki morfologi yang berbeda. Pada teks berbahasa Inggris, proses yang diperlukan hanya proses menghilangkan sufiks, sedangkan pada teks berbahasa Indonesia lebih rumit atau kompleks karena terdapat variasi imbuhan yang harus dibuang untuk mendapatkan *root word* (kata dasar) dari sebuah kata (Wahyudi et al., 2017). *Stemming* pada teks bahasa Indonesia lebih kompleks karena selain menghilangkan sufiks, juga menghilangkan prefix, infiks dan konfiks (kombinasi prefiks, sufiks dan infiks). Algoritma Nazief dan Adriani merupakan salah satu algoritma *stemming* yang dapat digunakan untuk mengubah kata atau token hasil *filtering* menjadi kata dasarnya untuk teks berbahasa Indonesia. Algoritma Nazief dan Adriani berdasarkan pada aturan morfologi bahasa Indonesia yang lebih luas, yang dikumpulkan menjadi satu kelompok dan pengkapsulan pada imbuhan yang diperbolehkan dan yang tidak diperbolehkan. Tahapan pada algoritma Nazief dan Adriani sebagai berikut (Adriani et al., 2007) :

1. Cari kata yang akan diubah menjadi kata dasar pada kamus, jika kata tersebut ditemukan maka algoritma dihentikan, dan kata tersebut dianggap sebagai *root word* (kata dasar). Jika tidak ditemukan maka dilanjutkan pada tahapan kedua.
2. Pada tahap ini menghapus *inflectional suffix* (imbuhan infleksional) seperti (“-lah”, “-kah”, “-tah”, “-ku”, “-mu”, “-nya”). Kemudian, menghapus *inflectional possessive pronoun suffixes* seperti (“ku”, “mu”, “nya”) jika ada. Selanjutnya, cek kata pada kamus, jika ditemukan maka algoritma dihentikan. Jika tidak ditemukan maka dilanjutkan pada tahapan ketiga.
3. Pada tahap ini menghapus *derivation suffix* (imbuhan turunan) seperti (“-i”, “-an”, atau “-kan”). Kemudian cek kata tersebut pada kamus, jika ditemukan maka algoritma berhenti. Jika kata tersebut tidak ditemukan maka dilanjutkan pada tahapan ke 3a :
 - a. Jika akhiran (“-an”) pada kata tersebut sudah dihapus, dan terdapat akhiran (“-k”), maka akhiran tersebut juga ikut dihapus. Selanjutnya, cek kata pada kamus, jika ditemukan maka algoritma dihentikan. Jika tidak ditemukan maka dilanjutkan pada tahap ke 3b.

- b. Akhiran (“-i”, “-an”, atau “-kan”) dikembalikan, dan algoritma dilanjutkan ke tahapan ke empat.
4. Pada tahap ini menghapus derivation prefix (awalan turunan) seperti (“be-“, “di-“, “ke-“, “me-“, “pe-“, “se-“, “te-“). Jika pada tahap ketiga ada sufiks yang dihapus maka, dilanjutkan ke tahap 4a. Jika tidak maka dilanjutkan pada tahap 4b :
- a. Pada tahap ini periksa pada tabel kombinasi awalan-akhiran yang tidak diperbolehkan. Jika ditemukan maka algoritma berhenti, jika tidak maka dilanjutkan pada tahap 4b. Berikut merupakan tabel kombinasi awalan-akhiran yang tidak diperbolehkan :

Tabel 2.8. Kombinasi Awalan-Akhiran Yang Tidak Diperbolehkan

| Awalan | Akhiran yang tidak diperbolehkan |
|--------|----------------------------------|
| be- | -i |
| di- | -an |
| ke- | -i, -kan |
| me- | -an |
| se- | -i, -kan |

- b. Pada tahap ini, for $i = 1$ to 3 tentukan tipe awalan selanjutnya hapus awalan. Jika kata dasar sudah ditemukan maka algoritma berhenti, jika belum maka dilanjutkan tahap kelima. Akan tetapi, jika awalan kedua sama dengan awalan pertama algoritma juga berhenti. Proses ini juga harus dilihat terlebih dahulu pada aturan tabel pemenggalan kata dasar sebagai berikut :

Tabel 2.9. Aturan Pemenggalan Kata Dasar

| Aturan | Awalan | Peluruhan |
|--------|------------------|-----------------------------------------------|
| 1 | berV... | ber-V.. be-rV.. |
| 2 | belajar | bel-ajar |
| 3 | berClerC2 | be-ClerC2.. dimana C1!= {‘r’ ‘l’} |
| 4 | terV... | ter-V.. te-rV.. |
| 5 | terCer... | terCer... dimana C!=‘r’ |
| 6 | teClerC2 | te-CleC2... dimana C1!=‘r’ |
| 7 | me{I r w y}V... | me-{I r w y}V... |
| 8 | mem{b f v}... | mem-{b f v}... |
| 9 | mempe... | m-pe.. |
| 10 | mem{r V V}... | me-m{r V V}... me-p{r V V}... |
| 11 | men{c d j z}... | men-{c d j z}... |
| 12 | menV... | me-nV... me-tV... |
| 13 | meng{g h q k}... | meng-{g h q k}... |
| 14 | mengV... | meng-V... meng-kV.. |
| 15 | mengeC... | meng-C... |
| 16 | menyV... | me-ny... men-sV... |
| 17 | memV... | mem-pV... |
| 18 | pe{w y}V... | pe-{w y}V... |
| 19 | perV... | per-V... pe-rV.. |
| 20 | pem{b f v}... | pem-{b f v}... |
| 21 | pem{r V V}... | pe-m{r V V}... pe-p{r V V}... |
| 22 | pen{c d j z}... | pen-{c d j z}... |
| 23 | penV... | pe-nV... pe-tV... |
| 24 | peng{g h q}... | peng-{g h q} |
| 25 | pengV... | peng-V peng-kV |
| 26 | penyV... | pe-nya peny-sV |
| 27 | pelV... | pe-IV...; kecuali untuk kata “pelajar” |
| 28 | peCP... | pe-CP... dimana C!= {r w y l m n} dan P!=‘er’ |
| 29 | perCerV... | per-CerV... dimana C!= {r w y l m n} |

Berikut ini merupakan tahapan untuk menentukan tipe awalan :

1. Jika awalannya (“di-”, “ke-”, “se-”) maka tipe awalannya secara berturut-turut adalah (“di-”, “ke-”, “se-”).
2. Jika awalannya (“te-”, “me-”, “be-”, “pe-”) maka dibutuhkan sebuah proses tambahan untuk menentukan tipe awalannya.
3. Jika dua karakter pertama bukan (“di-”, “ke-”, “se-”, “te-”, “be-”, “me-”, “pe-”) maka berhenti.
4. Jika tipe awalan adalah “none” maka berhenti. Jika tipe awalannya bukan “none”, maka awalan dihapus jika ditemukan.
5. Pada tahap ini melakukan proses *recording* untuk menambah karakter *recording* di awal kata yang dipenggal dengan mengacu pada tabel 2.9.
6. Jika semua tahapan telah dilakukan tetapi *root word* atau kata dasar belum ditemukan pada kamus, maka kata awal dianggap sebagai kata dasar, dan proses algoritma berhenti.

Contoh proses *stemming* ditunjukkan pada tabel berikut :

Tabel 2.10. Contoh Stemming Nazief dan Adriani

| <i>Input</i> | <i>Output</i> |
|----------------------------------------|--------------------------------------|
| ['bangunan', 'taman', 'sari', 'indah'] | ['bangun', 'taman', 'sari', 'indah'] |

2.6 Ekstraksi Fitur (*Term Weighting TF-IDF*)

Ekstraksi fitur adalah proses pengambilan ciri sebuah objek yang dapat menggambarkan karakteristik dari objek tersebut (Satria & Mushthofa, 2013). Ekstraksi fitur yang digunakan pada penelitian ini adalah *Term Weighting TF-IDF*. *Term Weighting TF-IDF* digunakan karena menyertakan frekuensi suatu *term* yang berupa kata, frasa atau unit hasil indeks lainnya dalam dokumen yang dapat digunakan untuk mengetahui konteks dari dokumen tersebut. Maka, untuk setiap kata tersebut diberikan indikator yang disebut *term weight*.

2.6.1 *Term frequency (TF)*

Term Frequency (TF) adalah frekuensi dari kemunculan sebuah *term* (*t*) dalam dokumen (*d*) tersebut. Jika nilai TF yang diperoleh tinggi maka semakin besar pula bobotnya. Ada beberapa jenis formula dalam TF antara lain :

1. TF Biner, hanya memberi nilai 1 atau 0. Jika, suatu kata atau *term* muncul dalam dokumen maka diberi nilai 1, jika tidak ada diberi nilai 0.
2. TF Murni, pada TF murni pemberian nilai sesuai dengan jumlah kata atau *term* yang muncul dalam dokumen tersebut. misalnya, muncul 3 kata yang sama maka kata tersebut diberi nilai 3 sesuai dengan jumlahnya.
3. TF Logaritmik, digunakan untuk menghindari dokumen yang mengandung sedikit kata atau *term* dalam *query*, namun mempunyai frekuensi yang tinggi kemunculannya dalam dokumen tersebut. Persamaan dari TF Logaritmik sebagai berikut :

$$TF = \begin{cases} 1 + \log_{10}(f_{t,d}), & f_{t,d} > 0 \\ 0, & f_{t,d} = 0 \end{cases} \quad (2.1)$$

Keterangan :

$f_{t,d}$: frekuensi *term* (*t*) pada dokumen(*d*)

4. TF Normalisasi, menggunakan perbandingan antara frekuensi sebuah term dengan nilai maksimum dari keseluruhan atau kumpulan frekuensi *term* yang ada pada suatu dokumen. Persamaan dari TF Normalisasi sebagai berikut :

$$TF = 0.5 + 0.5x \left[\frac{f_{t,d}}{\max\{f_{t',d: t',d d}\}} \right] \quad (2.2)$$

2.6.2 *Invers document frequency (IDF)*

Inverse Document Frequency (TF) adalah perhitungan dimana jika semakin sedikit jumlah dokumen yang mengandung kata atau *term* yang dimaksud, maka nilai IDF semakin besar. Persamaan dari IDF sebagai berikut:

$$IDF_j = \log(D/df_j) \quad (2.3)$$

Keterangan :

D : jumlah semua dokumen dalam koleksi.

df_j : jumlah dokumen yang mengandung *term* (*tj*).

2.6.3 *Term frequency-invers document frequency (TF-IDF)*

Term Frequency-Invers Document Frequency (TF-IDF) adalah penggabungan dari perhitungan TF Murni dengan IDF. Persamaan dari TF-IDF sebagai berikut :

$$\begin{aligned} W_{ij} &= tf_{ij} \times idf_j \\ w_{ij} &= tf_{ij} \times \log(D/df_i) \end{aligned} \quad (2.4)$$

Keterangan :

W_{ij} : bobot *term* (*tj*) terhadap dokumen (*di*).

tf_{ij} : jumlah kemunculan *term* (*tj*) dalam dokumen (*di*).

D : jumlah semua dokumen yang ada di dalam *database*.

df_i : jumlah dokumen yang mengandung *term* (*tj*) (minimal ada satu kata *term* (*tj*)).

2.7 Seleksi Fitur

Seleksi fitur merupakan tahapan untuk menghilangkan fitur pengganggu dalam klasifikasi. Seleksi fitur dapat mengurangi *term* yang tidak berpotensi menjadi indeks sehingga mempercepat proses pencarian (Budi & Aji, 2006). Seleksi fitur yang digunakan dalam penelitian ini akan dibandingkan, sehingga diperoleh seleksi fitur yang dapat meningkatkan hasil akurasi dari klasifikasi metode SVM yang lebih baik. Seleksi fitur yang digunakan pada penelitian ini sebagai berikut :

2.7.1 *Chi square*

Uji *Chi Square* berguna untuk menguji hubungan atau pengaruh dua buah variabel nominal dan mengukur kuatnya hubungan antara variabel yang satu dengan variabel nominal lainnya ($C = \text{Coefisien of contingency}$) (Wijayanto, 2009). *Chi Square* menggunakan teori statistika untuk menguji independensi sebuah *term* dengan kategorinya (Ling et al., 2014). Rumus perhitungan *Chi-Square* sebagai berikut (Thaseen & Kumar, 2017) :

$$X^2(t, c) = \frac{N \times (A \times D - C \times B)^2}{(A+C)(B+D)(A+B)(C+D)} \quad (2.5)$$

Keterangan :

A : Banyaknya dokumen yang memiliki *term* *t* pada kelas *c*

B : Banyaknya dokumen yang memiliki *term* *t* pada kelas selain kelas *c*

C : Banyaknya dokumen tanpa *term* *t* di kelas *c*

D : Banyaknya dokumen tanpa *term* *t* di bukan kelas *c*

N : Banyaknya dokumen latih

Variabel-variabel di atas didapatkan dari korelasi antara *term* t dan *class* c dari tabel kontingensi berikut ini :

Tabel 2.11. Kontingensi *term* atau kata

| | c | not c |
|-------|---|-------|
| t | A | B |
| not t | C | D |

Setelah semua fitur atau *term* dihitung nilai *Chi Square*-nya, kemudian semua fitur akan diurutkan berdasarkan nilai *Chi Square* yang paling tinggi ke yang paling rendah.

2.7.2 Mutual information

Mutual Information dari dua variabel acak adalah sebuah pengukuran kuantitatif dari jumlah ketergantungan (informasi) antara dua variabel acak (Sulaiman & Labadin, 2015). *Mutual Information* yang tinggi menunjukkan penurunan ketidakpastian yang besar, *Mutual Information* yang rendah menunjukkan sedikit penurunan, sedangkan *Mutual Information* yang bernilai nol antara dua variabel acak berarti variabel tersebut independen (Latham & Roudi, 2009). Berikut ini persamaan dari *Mutual Information* (Irham et al., 2019) :

$$I(U, C) = \sum_{et \in \{1,0\}} \sum_{ec \in \{1,0\}} P(U = et, C = ec) \log_2 \frac{P(U = et, C = ec)}{P(U = et)P(C = ec)} \quad (2.6)$$

Keterangan :

U : nilai dari *term* atau kata

C : nilai dari kelas

et : bernilai 0 jika tidak mengandung term t, sedangkan bernilai 1 jika mengandung term t

ec : bernilai 0 jika *term* tidak berada di kelas c, sedangkan bernilai 1 jika term berada dikelas c

Jika persamaan tersebut diturunkan maka persamaan *Mutual Information* di atas akan menjadi persamaan seperti berikut ini :

$$I(U, C) = \frac{N_{11}}{N} \log_2 \frac{N \cdot N_{11}}{N_1 \cdot N_1} + \frac{N_{01}}{N} \log_2 \frac{N \cdot N_{01}}{N_0 \cdot N_1} + \frac{N_{10}}{N} \log_2 \frac{N \cdot N_{10}}{N_1 \cdot N_0} + \frac{N_{00}}{N} \log_2 \frac{N \cdot N_{00}}{N_0 \cdot N_0} \quad (2.7)$$

Keterangan :

N : jumlah dokumen yang memiliki et dan ec atau ($N = N_{00} + N_{01} + N_{10} + N_{11}$)

$N_{1\cdot}$: jumlah dokumen yang memiliki et atau ($N_{1\cdot} = N_{10} + N_{11}$)

$N_{1\cdot}$: jumlah dokumen yang memiliki ec atau ($N_{1\cdot} = N_{01} + N_{11}$)

$N_{0\cdot}$: jumlah dokumen yang tidak memiliki et atau ($N_{0\cdot} = N_{01} + N_{00}$)

N_0 : jumlah dokumen yang tidak memiliki ec atau ($N_0 = N_{10} + N_{00}$)

Kemudian untuk menempatkan nilai jumlah kemunculan *term* atau kata digunakan tabel *confusion matrix* seperti berikut ini :

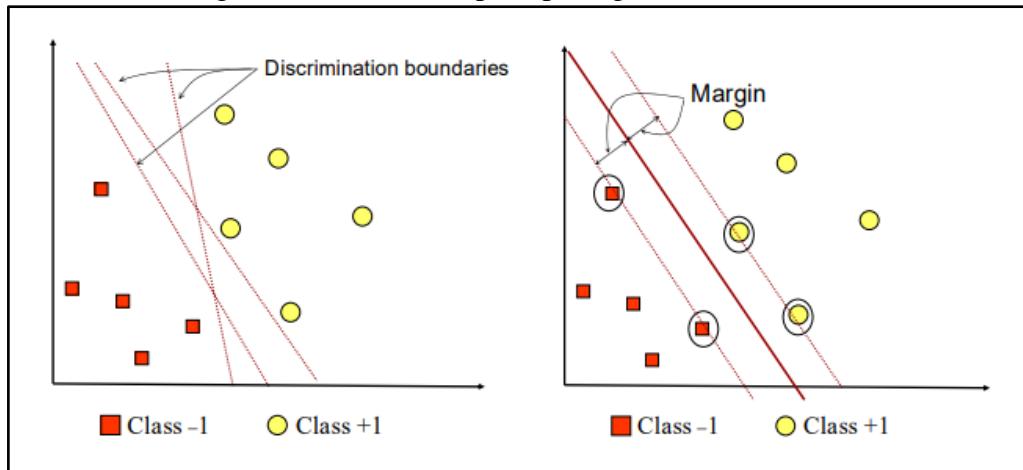
Tabel 2.12. Confusion Matrix nilai jumlah kemunculan *term* atau kata

| | ec = 1 | ec = 0 |
|--------|----------|----------|
| et = 1 | N_{11} | N_{10} |
| et = 0 | N_{01} | N_{00} |

Setelah semua fitur dihitung nilainya untuk kelas positif, maka akan dilakukan perhitungan nilai *Mutual Information* fitur tersebut untuk kelas negatif. Kemudian hasil nilai *Mutual Information* fitur dari kedua kelas tersebut akan dibandingkan, dan akan dipilih nilai *Mutual Information* terbesar dari fitur tersebut. Kemudian semua fitur akan diurutkan berdasarkan nilai *Mutual Information* yang paling tinggi ke yang paling rendah.

2.8 Support Vector Machine

Support Vector Machine (SVM) merupakan salah satu metode klasifikasi dengan menggunakan *machine learning (supervised learning)* yang memprediksi kelas berdasarkan model atau pola dari hasil proses *training* (Maulana & Redjeki, 2017). Pada SVM dokumen yang akan diklasifikasi diubah terlebih dahulu menjadi *support vector*. Biasanya teks dokumen diubah menjadi vector multidimensioal tf-idf (Ipmauti, 2016). Klasifikasi pada SVM dilakukan dengan mencari *hyperplane* atau kernel maksimal yang memisahkan dua buah kelas. Dua buah kelas tersebut antara lain kelas sentimen positif (berlabel +1) dengan dan kelas sentimen negatif (berlabel -1) seperti pada gambar berikut ini.



Gambar 2.1. SVM Berusaha Menemukan *Hyperplane* Terbaik Yang Memisahkan Kedua Class -1 Dan +1 (Nugroho et al., 2003)

Pada gambar di atas dua buah kelas dipisahkan oleh bidang pembatas yang sejajar atau kernel, sehingga diperoleh (Nugroho et al., 2003):

1. Diasumsikan kedua kelas -1 dan +1 dapat terpisah secara sempurna oleh *hyperplane* atau bidang pembatas berdimensi d , yang didefinisikan :

$$w \cdot x + b = 0 \quad (2.8)$$

2. *Pattern* x_i yang termasuk ke dalam kelas -1 (kelas negatif) dapat dirumuskan sebagai *pattern* yang memenuhi pertidaksamaan sebagai berikut :

$$w \cdot x_i + b \leq -1 \quad (2.9)$$

3. *Pattern* x_i yang termasuk ke dalam kelas 1 (kelas positif) dapat dirumuskan sebagai *pattern* yang memenuhi pertidaksamaan sebagai berikut :

$$w \cdot x_i + b \geq 1 \quad (2.10)$$

4. Fungsi untuk mencari nilai *Lagrange Multiplier* α sebagai berikut :

$$\max_{\alpha} L_D = \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i,j=1}^n \alpha_i \alpha_j y_i y_j K(x_i, x_j) \quad (2.11)$$

dengan syarat : $\alpha_i \geq 0$ ($i = 1, 2, \dots, n$) dan $\sum_{i=1}^n \alpha_i y_i = 0$

Setelah diperoleh nilai dari fungsi di atas, maka selanjutnya mencari nilai w , nilai bias, dan mencari fungsi keputusan klasifikasi *sign* ($f(x)$) dengan persamaan atau *formula* sebagai berikut (Sembiring, 2007) :

5. Persamaan untuk mencari nilai w sebagai berikut :

$$w = \sum_{i=1}^n \alpha_i y_i x_i \quad (2.12)$$

6. Persamaan untuk mencari nilai bias sebagai berikut :

$$b = -\frac{1}{2} (x_i^+ \cdot w + x_i^- \cdot w) \quad (2.13)$$

7. Persamaan untuk mencari fungsi keputusan klasifikasi $\text{sign}(f(x))$ sebagai berikut :

$$f(x) = \text{sign} \left[\sum_{i=1}^m \alpha_i y_i K(x, x_i) + b \right] \quad (2.14)$$

Keterangan :

x : Titik data masukan *support vector machine*

w : Parameter *hyperplane* yang dicari (garis yang tegak lurus antara garis *hyperplane* dan titik *support vector*)

n : Banyaknya data

Ld : Dualitas *Lagrange Multipier*

α_i : Nilai bobot setiap titik data

m : jumlah *support vector/titik data* yang memiliki $\alpha_i > 0$

$K(x, x_i)$: fungsi kernel

b : parameter *hyperplane* yang dicari (nilai bias)

Fungsi *sign()* merupakan fungsi normalisasi, jika nilai x di dalam fungsi *sign()* lebih dari 0 atau > 0 maka fungsi tersebut memberikan nilai 1 (kelas positif). Jika nilai x dalam fungsi *sign()* kurang dari 0 atau < 0 maka fungsi tersebut memberikan nilai -1(kelas negatif).

Berikut ini merupakan beberapa fungsi kernel yang sering digunakan dalam metode *Support Vector Machine* (Durgesh & Lekha, 2010) :

1. Linear Kernel

$$K(x_i, x_j) = x_i^T x_j \quad (2.15)$$

Dengan x_i merupakan data latih dan x_j adalah data uji.

2. Polynomial Kernel

$$K(x_i, x_j) = (\gamma x_i^T x_j + r)^p, \quad \gamma > 0 \quad (2.16)$$

Dengan x_i merupakan data latih, x_j adalah data uji, p adalah derajat *polynomial*, γ adalah *scalling* parameter dari jarak *Euclidean* dan r adalah nilai konstanta.

3. Radial Basis Function (RBF) Kernel

$$K(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|^2), \quad \gamma > 0 \quad (2.17)$$

Dengan x_i merupakan data latih, x_j adalah data uji, dan γ adalah *scalling* parameter dari jarak *Euclidean*.

4. Sigmoid (Tangen Hiperbolik) Kernel

$$K(x_i, x_j) = \tanh(\gamma x_i^T x_j + r) \quad (2.18)$$

Dengan x_i merupakan data latih, x_j adalah data uji, γ adalah *scalling* parameter dari jarak *Euclidean* dan r adalah nilai konstanta.

2.9 Confusion Matrix

Confusion matrix merupakan sebuah tabel yang menyatakan jumlah data uji yang benar diklasifikasikan dan jumlah data uji yang salah diklasifikasikan (Indriani, 2014). *Confusion matrix* memiliki empat kemungkinan yang merepresentasikan hasil dari proses klasifikasi, diantaranya TP, TN, FP, FN. Empat kemungkinan *confusion matrix* ditunjukkan pada tabel berikut :

Tabel 2.13 Confusion Matrix

| | | <i>Predicted Class</i> | |
|---------------------|-----------------|------------------------|-----------------|
| | | <i>Positive</i> | <i>Negative</i> |
| <i>Actual Class</i> | <i>Positive</i> | TP | FN |
| | <i>Negative</i> | FP | TN |

Keterangan :

- TP (*True Positive*) : Jumlah data yang diprediksi pada kelas positif dan benar berada pada kelas positif.
- FN (*False Negative*) : Jumlah data yang diprediksi pada kelas negatif namun seharusnya berada pada kelas positif.
- FP (*False Positive*) : Jumlah data yang diprediksi pada kelas positif namun seharusnya berada pada kelas negatif.
- TN (*True Negative*) : Jumlah data yang diprediksi pada kelas negatif dan benar berada pada kelas negatif.

Kemudian setelah jumlah data didapat pada tabel *confusion matrix*, maka akan dihitung nilai akurasi, presisi dan *recall* menurut (Han & Kamber, 2006) :

a. Akurasi

akurasi merupakan persentase sampel set pengujian yang ada diklasifikasikan dengan benar berdasarkan model. Nilai akurasi merepresentasikan kedekatan antara *actual value* dan *predicted value*. Persamaan untuk nilai akurasi ditunjukkan sebagai berikut :

$$\text{Akurasi} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{FN} + \text{TN}} \times 100\% \quad (2.19)$$

b. Presisi

Presisi merupakan tingkat ketepatan informasi yang diminta pengguna dengan jawaban yang diberikan oleh sistem. Nilai presisi merepresentasikan perbandingan antara jumlah data yang diprediksi pada kelas positif dan benar berada pada kelas positif, dengan keseluruhan data yang diprediksi pada kelas positif. Persamaan untuk nilai presisi ditunjukkan sebagai berikut :

$$\text{Presisi} = \frac{\text{TP}}{\text{TP} + \text{FP}} \times 100\% \quad (2.20)$$

c. Recall

Recall merupakan perbandingan antara jumlah data yang diprediksi pada kelas positif dan benar berada pada kelas positif, dengan keseluruhan data yang benar positif. Dengan kata lain, *recall* merupakan tingkat keberhasilan sistem untuk menemukan kembali sebuah informasi. Persamaan untuk nilai *recall* ditunjukkan sebagai berikut :

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \times 100\% \quad (2.21)$$

2.10 *K-Fold Cross Validation*

Validasi yang digunakan pada penelitian ini adalah *k-fold cross validation*. Pengujian *k-fold cross validation* membagi data sebanyak *k* bagian secara acak dengan masing-masing bagian memiliki jumlah yang sama. Kemudian data yang sudah dibagi menjadi *k* bagian akan menjadi data latih dan data uji secara bergantian sebanyak *k* iterasi. Tujuan digunakannya *k-fold cross validation* pada penelitian ini adalah untuk memvalidasi algoritma klasifikasi yang digunakan lebih teruji dan kinerja yang dihasilkan bersifat valid (Pulungan, 2019).

2.11 Pembobotan Top Rekomendasi

Pembobotan top rekomendasi tempat wisata bertujuan untuk mengetahui bobot ranking dari setiap wisata. Pada penelitian ini, menggunakan persamaan pembobotan top rekomendasi berdasarkan penelitian yang dilakukan oleh (Murnawan, 2017) dan (Azmi et al., 2020) yang ditunjukkan sebagai berikut :

$$\text{Popularitas Wisata } X = \frac{\text{Total Ulasan Wisata } X}{\text{Total Ulasan Semua Wisata}} \times 100\% \quad (2.22)$$

$$Bobot Ranking Wisata X = \frac{\text{Total Ulasan Positif Wisata } X}{\text{Total Ulasan Wisata } X} \times \text{Popularitas Wisata } X \quad (2.23)$$

Keterangan :

Total ulasan wisata x : Jumlah ulasan positif dan ulasan negatif wisata x.

Total Ulasan Semua Wisata : Jumlah seluruh data ulasan semua wisata.

Setelah semua wisata telah diketahui bobot rankingnya, kemudian akan dibuat daftar top rekomendasi berdasarkan bobot ranking wisata tertinggi ke terendah. Jika terdapat wisata yang memiliki bobot ranking yang sama, maka akan berada pada peringkat yang sama.

2.12 Penelitian Sebelumnya

Adapun beberapa penelitian sebelumnya yang dijadikan referensi pada penelitian ini sebagai berikut :

Tabel 2.14 State of the Art

| Model | Masalah Penelitian | Dataset | Pengukuran | Hasil |
|---------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| (Nurajijah & Riana, 2019) | Belum adanya kajian mengenai beberapa algoritma klasifikasi untuk menentukan persetujuan pemberian kredit | Dataset pembiayaan dari salah satu koperasi syariah yang diambil dari bulan januari 2015 sampai september 2018 sebanyak 819 record data latih (543 berlabel lancar dan 276 berlabel macet) dan 50 record data uji (35 berlabel lancar dan 15 berlabel macet). | Akurasi, presisi, recall, dan AUC. | <p>Metode SVM terbukti lebih unggul dari metode <i>Naïve Bayes</i> dan <i>Decision Tree</i> dengan hasil akurasi 89.86%, presisi kelas lancar 88.46%, presisi kelas macet 93.67%, recall kelas lancar 97.42%, recall kelas macet 75%, dan AUC sebesar 0.939.</p> <p>Metode <i>Naïve Bayes</i> hanya memiliki akurasi 77.29%, presisi kelas lancar 90.29%, presisi kelas macet 61.97%, recall kelas lancar 73.66%, recall kelas macet 84.42% dan AUC sebesar 0.882.</p> <p>Metode <i>Decision Tree</i> memiliki akurasi 89.02%, presisi kelas lancar 89.95%, presisi kelas macet 86.90%, recall kelas lancar 93.92%, recall kelas macet 79.35% dan AUC sebesar 0.882.</p> |
| (Sutoyo & Almaarif, 2020) | Perbandingan metode klasifikasi SVM, <i>Naïve Bayes Classifier</i> , <i>Logistic Regression</i> dan <i>k-nearest neighbor</i> pada analisis sentimen untuk mencari opini public tentang wacana pemindahan ibu kota dari Jakarta ke Penajam Paser Utara dan Kutai Kartanegara. | Data <i>twitter</i> dari <i>tweet</i> pengguna mengenai wacana relokasi ibu kota yang diambil dari tanggal 20 agustus 2019 sampai 19 september 2019 menggunakan API dari <i>twitter</i> . Dataset berjumlah 2891 <i>tweet</i> dengan | Akurasi, presisi, recall, f-measure. | <p>Metode SVM terbukti lebih unggul dibandingkan metode NBC, <i>Logistic Regression</i> dan KNN dengan hasil akurasi 97.72%, presisi 96.01%, recall 99.18%, dan f-measure 97.57%.</p> <p>Metode NBC hanya memiliki akurasi 91.65%, presisi 93.45%, recall 88.07%, f-measure 90.58%.</p> <p>Metode <i>Logistic Regression</i> hanya</p> |

Tabel 2.15 Lanjutan State of the Art

| Model | Masalah Penelitian | Dataset | Pengukuran | Hasil |
|---------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| | | rincian 1638 <i>tweet</i> berlabel positif dan 1253 <i>tweet</i> berlabel negatif. | | memiliki akurasi sebesar 96.58%, presisi 94.47%, <i>recall</i> 98.35%, <i>f-measure</i> 96.37%. Metode KNN hanya memiliki akurasi sebesar 90.70%, presisi 95.75%, <i>recall</i> 83.54%, <i>f-measure</i> 89.23%. |
| (Irham et al., 2019) | Pengaruh dari penggunaan metode <i>stopword removal</i> dan <i>stemming</i> pada <i>text preprocessing</i> , pengaruh metode seleksi fitur <i>mutual information</i> (MI) dalam menyeleksi fitur, dan performansi SVM dalam mengklasifikasikan data berita berdasarkan topiknya. | 360 data teks berita bahasa Indonesia yang telah diberi label berdasarkan topiknya (setiap topik memiliki 30 data berita). | Pengujian pertama : pengaruh penggunaan <i>stopword removal</i> dan <i>stemming</i> pada tahap <i>text preprocessing</i> terhadap hasil dan kinerja model. Pengujian kedua : pengaruh penggunaan metode seleksi fitur MI terhadap hasil dan kinerja model. Pengujian ketiga : menguji performansi metode SVM untuk membuktikan jika metode klasifikasi yang dipakai merupakan metode terbaik untuk klasifikasi berita bahasa Indonesia. | Hasil pengujian pertama yaitu nilai akurasi tertinggi didapatkan dari penggunaan <i>stemming</i> tanpa <i>stopword removal</i> . Sedangkan, waktu komputasi paling cepat ketika menggunakan <i>stopword removal</i> tanpa <i>stemming</i> . Hasil pengujian kedua yaitu pada perhitungan akurasi pada indeks <i>fold</i> pertama dari <i>5-fold</i> diperoleh hasil akurasi tertinggi sebesar 93.98% dengan waktu komputasi hanya 100.95 detik ketika menggunakan seleksi fitur MI (jumlah fitur sebanyak 5000). Sedangkan, tanpa seleksi fitur MI hanya memperoleh akurasi 93.63% dengan waktu komputasi 197.66 detik (lebih lama). Hasil pengujian ketiga yaitu hasil akurasi tertinggi dan waktu komputasi tercepat diperoleh oleh metode SVM sebesar 94.24%. Sedangkan metode NN akurasi 94.14%, KNN akurasi 91.13%, NB 93%. |
| (Ukhti Ikhsani Larasati et al., 2019) | Metode SVM bekerja kurang baik pada data berukuran besar. Selain itu, pada proses <i>text mining</i> memiliki kendala dalam menentukan jumlah atribut yang digunakan. Semakin banyak atribut yang digunakan juga dapat mengurangi kinerja pengklasifikasi | ulasan film yang telah diberi label yang diambil dari UCI repository of machine learning. | akurasi | Akurasi SVM tanpa seleksi fitur <i>Chi Square</i> dan TFIDF sebesar 68.7%, sedangkan akurasi <i>SVM+Chi Square+TFIDF</i> sebesar 80.2%. oleh karena itu, penerapan seleksi fitur <i>Chi Square</i> dan pembobotan TFIDF pada metode SVM dapat meningkatkan hasil akurasi SVM dalam analisis sentimen ulasan |

Tabel 2.16 Lanjutan State of the Art

| Model | Masalah Penelitian | Dataset | Pengukuran | Hasil |
|------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| | sehingga akurasi menjadi rendah. | | | film. |
| (Thaseen & Kumar, 2017) | Jumlah data yang besar mengakibatkan gagalnya pengklasifikasi tunggal untuk mencapai tingkat deteksi serangan yang tinggi dengan pengurangan tingkat alarm palsu. | 33000 <i>records</i> dataset NSL-KDD yang berisi lima kategori lalu lintas jaringan (normal, DoS, R2L, <i>probe</i> , U2R) dimana 8325 <i>records</i> digunakan sebagai data latih dan 24.975 digunakan sebagai data uji. | <i>Kappa statistic</i> , MAE, RMSE, presisi, <i>recall</i> , <i>f-measure</i> , ROC, <i>Mean Accuracy</i> , <i>False Alarm Rate</i> , komparasi akurasi dan waktu pada metode sebelumnya. | <i>Kappa statistic</i> = 0.9351. MAE = 0.0243. RMSE = 0.1102. Presisi tertinggi pada kelas DoS = 0.998. <i>Recall</i> tertinggi pada kelas DoS = 0.999. <i>f-measure</i> tertinggi pada kelas DoS = 0.998 ROC tertinggi pada kelas DoS dan <i>probe</i> = 1. <i>Mean accuracy</i> tertinggi diperoleh ketika $c = 1$ dan $\gamma = 0.07$ sebesar 98.1% dengan <i>false alarm rate</i> sebesar 1.9%. Model deteksi instruksi menggunakan pemilihan fitur fitur <i>chi square</i> dan multi kelas SVM yang dioptimalkan dengan parameter teknik penyetalan (metode yang diusulkan) memiliki hasil akurasi yang stabil dan hasil rata-rata akurasi tertinggi sebesar 98% dengan waktu paling cepat 0.13 dibandingkan metode sebelumnya. |
| (Pratama & Trilaksono, 2015) | Mengklasifikasikan bentuk dan jenis <i>tweet</i> keluhan yang masuk pada akun @SpeedyTelkomsel menggunakan metode SVM dengan kernel <i>Gaussian RBF</i> dan hasil perbandingan ekstraksi fitur dan penggabungan metode ekstraksi fitur yang digunakan terhadap performansi model yang dibangun. | Kumpulan <i>tweet</i> yang masuk ke akun @SpeedyTelkomsel sebanyak 600 <i>tweet</i> , dimana 480 <i>tweet</i> digunakan sebagai data <i>training</i> dan 120 <i>tweet</i> digunakan sebagai data <i>testing</i> . | Akurasi, presisi, <i>recall</i> , <i>f-measure</i> | Metode TF menghasilkan akurasi paling baik sebesar 82.50% untuk klasifikasi bentuk keluhan. Metode <i>Chi Square</i> menghasilkan akurasi paling baik sebesar 86.67% untuk klasifikasi jenis keluhan. Penggabungan metode TF+IG+ <i>Chi Square</i> meningkatkan hasil akurasi untuk klasifikasi bentuk keluhan sebesar 83.33% dan untuk klasifikasi jenis keluhan sebesar 89.17%. Rata-rata nilai presisi, <i>recall</i> , <i>f-measure</i> untuk klasifikasi bentuk |

Tabel 2.17 Lanjutan State of the Art

| Model | Masalah Penelitian | Dataset | Pengukuran | Hasil |
|-------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| | | | | keluhan setiap parameter sebesar 83.67%, 83.33%, 83.29%. Rata-rata nilai presisi, <i>recall</i> , <i>f-measure</i> untuk klasifikasi jenis keluhan setiap parameter sebesar 89.76%, 89.17%, 89.34%. |
| (Handhayani & Hiryanto, 2017) | Memilih mata kuliah yang memiliki kedekatan hubungan dengan lamanya waktu belajar mahasiswa menggunakan metode <i>mutual information</i> (MI) pada klasifikasi metode SVM. | Kumpulan data siswa ilmu komputer di universitas x ditahun 2008 sampai 2012 yang terdiri dari 240 alumni dan 25 mata kuliah. | Rata-rata akurasi pada percobaan pertama (2 kelas) dan kedua (3 kelas) | Percobaan pertama dengan 2 kelas yaitu kinerja SVM menghasilkan akurasi sebesar 83.64% dan mengalami kenaikan menjadi 85.72% setelah diterapkannya pemilihan fitur menggunakan MI. Percobaan kedua dengan 3 kelas yaitu kinerja SVM <i>one vs one</i> menghasilkan akurasi 77.2% dan SVM <i>one vs rest</i> menghasilkan akurasi 77.2%. setelah menerapkan pemilihan fitur MI mengalami kenaikan menjadi 80.58% dan 80.82%. Algoritma SVM+MI yang diusulkan memiliki akurasi lebih tinggi dibandingkan metode <i>Decision Tree</i> dan <i>Gaussian NB</i> . |
| (Somantri & Dairoh, n.d.) | Mencari model terbaik untuk analisis sentimen penilaian positif dan negatif tempat pariwisata di Kota Tegal dengan membandingkan performansi metode <i>Naïve Bayes</i> dan <i>Decision Tree</i> . | Kumpulan komentar opini ulasan dari situs <i>google maps</i> dengan jumlah data sebanyak 120 data <i>file</i> yang diambil pada tahun 2017 sampai 2018. | Akurasi | Metode <i>Naïve Bayes</i> memiliki akurasi yang lebih baik sebesar 73.33% dibandingkan dengan metode <i>Decision Tree</i> yang hanya memperoleh akurasi sebesar 60.83%. |
| (Murnawan, 2017) | Menentukan pemeringkatan popularitas tujuan pariwisata dengan analisis sentimen penilaian positif dan negatif di media sosial <i>instagran</i> dan <i>facebook</i> menggunakan | Melakukan <i>crawling</i> pada media sosial <i>facebook</i> dan <i>Instagram</i> pada 50 <i>point of interest</i> pada pariwisata di Bali yang sudah ditentukan. | Akurasi | Metode <i>Naïve Bayes</i> menghasilkan akurasi sebesar 65.65% ketika menggunakan data uji 100 <i>phrase</i> dan memperoleh akurasi sebesar 82.67% ketika menggunakan data uji 5000 <i>phrase</i> . |

Tabel 2.18 Lanjutan State of the Art

| Model | Masalah Penelitian | Dataset | Pengukuran | Hasil |
|--------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| | metode klasifikasi <i>Naïve Bayes</i> . | | | |
| (Utama et al., 2019) | Mengetahui tingkat akurasi, presisi, <i>recall</i> yang dihasilkan oleh metode SVM dalam menganalisis sentimen masyarakat mengenai efektifitas sistem ganjil genap pada jalan tol Bekasi di media sosial dengan menggunakan seleksi fitur <i>information gain</i> untuk mengurangi <i>noise</i> pada data yang digunakan, sehingga dapat digunakan menjadi bahan evaluasi oleh pemerintah atau jasa marga selaku pembuat program. | Kumpulan komentar masyarakat terhadap efektifitas sistem ganjil genap di jalan tol Bekasi yang diambil dari media sosial seperti <i>youtube</i> , <i>twitter</i> , <i>Instagram</i> dan <i>facebook</i> . Data yang digunakan sebanyak 440 data (220 data berlabel positif dan 220 data berlabel negatif). | Akurasi, presisi, <i>recall</i> | Model SVM memiliki performansi lebih baik dengan tingkat akurasi 78.18%, presisi 74.03%, dan <i>recall</i> 86.82%, dibandingkan metode KNN yang hanya memperoleh akurasi sebesar 57.05%. |
| (Fransiska et al., 2020) | <i>Review</i> yang cukup banyak dan tidak terstruktur yang diberikan pelanggan by.U di <i>google playstore</i> sehingga sulit memperoleh informasi tingkat kepuasan, saran dan kritikan dari pelanggan. Banyaknya studi penelitian mengenai metode TFIDF+SVM untuk metode klasifikasi tapi sedikit studi yang membahas tentang pengaruh penggunaan TFIDF terhadap performansi metode SVM. | Data diambil dari <i>review</i> aplikasi by.U pada <i>google playstore</i> dengan cara <i>scraping</i> , sebanyak 8.925 data diperoleh dimana 4.874 data berlabel positif dan 4.051 data berlabel negatif. | Akurasi, presisi, <i>f-measure</i> , <i>recall</i> . | Metode SVM+TFIDF menghasilkan rata-rata nilai akurasi sebesar 84.67%, 84.9%, <i>recall</i> 84.7%, <i>f-measure</i> 84.8%. Hasil akurasi tertinggi diperoleh pada <i>fold</i> kedua sebesar 86.1%. Penggunaan TFIDF pada metode SVM tidak terlalu berpengaruh tapi menghasilkan hasil yang lebih baik. |
| (Ulfia et al., 2018) | Perlunya pemilihan fitur untuk mempersingkat waktu pelatihan dan prediksi waktu kalkulasi. | <i>Crawling</i> 500 data <i>twitter</i> berbahasa inggris yang dilabeli secara manual, akan tetapi data yang digunakan hanya 248 data | Akurasi, presisi, <i>recall</i> , waktu klasifikasi. | Penggunaan seleksi fitur <i>Mutual Information</i> meningkatkan performansi dari model yang dibuat oleh metode <i>Naïve Bayes Classifier</i> dengan hasil akurasi sebesar 97.2% yang sebelumnya tanpa |

Tabel 2.19 Lanjutan State of the Art

| Model | Masalah Penelitian | Dataset | Pengukuran | Hasil |
|----------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| | | yang berisi <i>hash-tag</i> #lombok dan #wonderfullombok karena pada proses <i>text preprocessing</i> ada beberapa <i>tweet</i> yang hanya berisi <i>hash-tag</i> sehingga kosong, dan <i>tweet</i> hasil <i>retweet</i> juga dikecualikan. | | <i>Mutual Information</i> hanya memperoleh akurasi 96.2%. Penggunaan seleksi fitur <i>mutual information</i> juga memberikan pengaruh terhadap kenaikan presisi dan <i>recall</i> , waktu klasifikasi juga berkurang 51.52%. |
| (Tineges et al., 2020) | Tingkat akurasi model klasifikasi yang dihasilkan dari metode SVM dan tingkat kepuasan pengguna <i>internet service provider</i> indihome. | Kumpulan <i>tweet</i> yang tertuju pada akun <i>twitter</i> @IndiHome sebanyak 10.000 <i>tweet</i> yang diambil pada tanggal 16-23 maret 2020 (pada awal pemberitahuan WFH dan <i>social distancing</i>). | Akurasi, presisi, <i>recall</i> , <i>f1-score</i> , AUC, <i>error rate</i> . | Metode SVM menghasilkan akurasi 87%, presisi 86%, <i>recall</i> 95%, <i>error rate</i> 13%, <i>f1-score</i> 90% dan AUC 0.93. |
| (Putu & Amrullah, 2021) | Beragam topik dan komentar dari wisatawan tentang wisata Lombok yang disampaikan melalui akun media sosialnya, sehingga sangat sulit untuk menentukan klasifikasi sentimen publik dan topik pembicaraan secara manual. | Sebanyak 12.971 data <i>tweet</i> dengan topik pariwisata Lombok yang diambil melalui API <i>twitter</i> dengan menggunakan <i>hashtag</i> yang diambil dalam rentang data 5 tahun terakhir (2014 sampai 2019). | Akurasi, presisi, <i>recall</i> , spesifitas, nilai koherensi. | Hasil analisis sentimen dengan data 9.496 <i>tweet</i> (8.996 <i>tweet</i> sentimen positif dan 500 <i>tweet</i> sentimen negatif), dari data tersebut didapatkan bahwa wisatawan lebih banyak memberikan respon positif mengenai wisata Lombok. Metode <i>Naïve Bayes</i> mampu melakukan klasifikasi sentimen wisatawan dengan baik dengan hasil akurasi 92%, presisi, 100%, <i>recall</i> 83.84%, dan spesifitas 100%. Metode LDA menghasilkan topik terbaik untuk kelas positif pada 8 topik dengan nilai koherensi 0.613, sedangkan untuk kelas negatif pada 12 topik dengan nilai koherensi 0.528. |
| (Hasanah & Yustanti, 2021) | Kabupaten banyuwangi merupakan tempat pariwisata sehingga diperlukan tanggapan dari wisatawan untuk | 50 data tempat wisata dan data sumber yang diambil dari situs seperti <i>TripAdvisor</i> , <i>Instagram</i> , dan | Akurasi, presisi, <i>recall</i> . | Metode <i>Naïve Bayes</i> menghasilkan akurasi 74%, presisi 61%, dan <i>recall</i> 55%. Meskipun metode <i>Naïve Bayes</i> memiliki performansi yang baik, |

Tabel 2.20 Lanjutan State of the Art

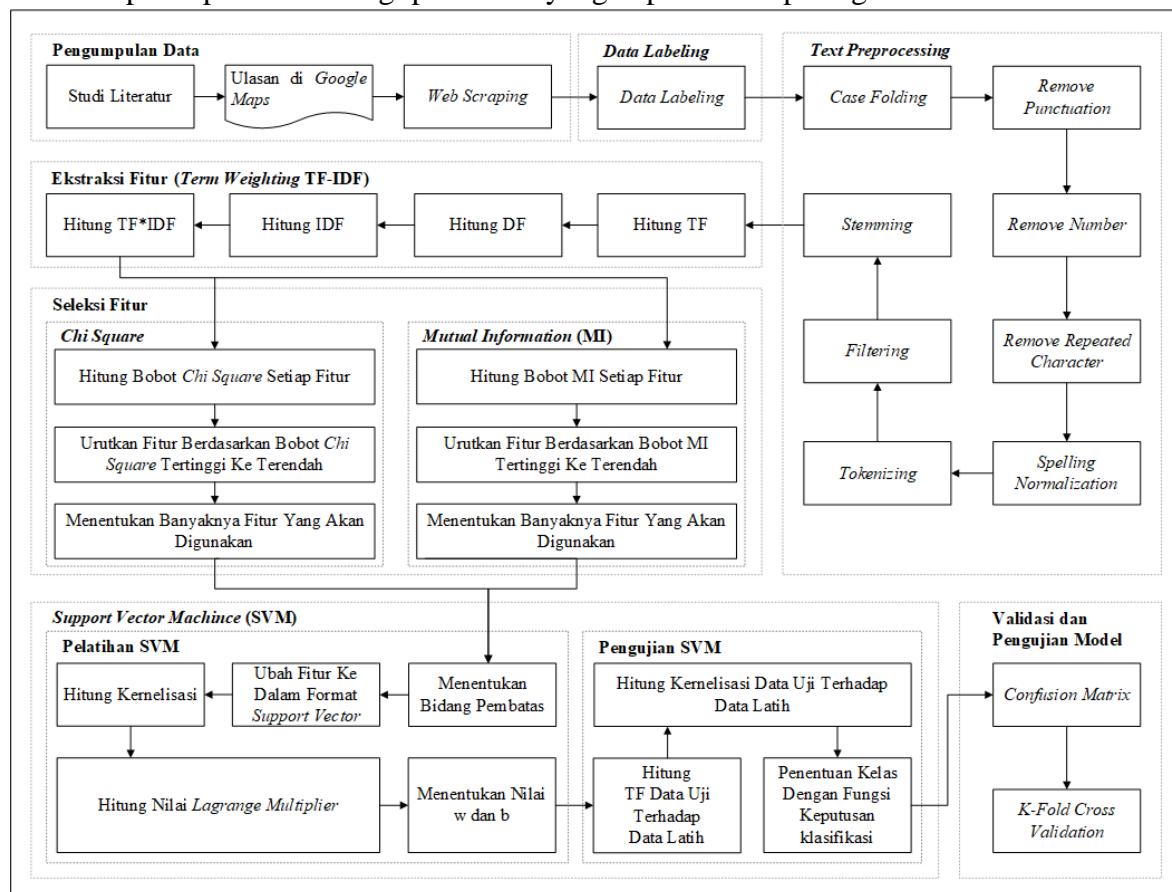
| Model | Masalah Penelitian | Dataset | Pengukuran | Hasil |
|---------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| | mengetahui popularitas suatu wisata agar dapat memperbaiki wisata yang kurang populer. | Twitter. | | akan tetapi kurang dapat mengklasifikasikan kalimat negatif karena pengaruh jumlah data latih negatif yang relatif lebih sedikit dibandingkan jumlah data latih positif. Metode AHP menghasilkan peringkat tempat wisata di kabupaten Banyuwangi yang valid karena memiliki nilai konsistensi dibawah 0.1. |
| (Azmi et al., 2020) | Pemanfaatan analisis sentiment data dari media sosial <i>Instagram</i> untuk mengetahui reputasi tempat wisata di Pulau Lombok, dalam prosesnya mengalami kesulitan karena terdapat kata yang tidak baku, emotikon, dan kesalahan eja. | 600 dataset yang diambil dari media sosial <i>Instagram</i> yang berkaitan dengan kata kunci pantailombok, gililombok, dan gunung lombok. | Akurasi, presisi, <i>recall</i> , <i>f1-score</i> | Metode <i>Naive Bayes</i> menghasilkan akurasi sebesar 59%, presisi sebesar 25% untuk sentimen netral, presisi sebesar 67% untuk sentimen positif, <i>recall</i> sebesar 20% untuk sentimen netral, <i>recall</i> sebesar 73% untuk sentimen positif, <i>f1-score</i> sebesar 22% untuk sentimen netral dan <i>f1-score</i> sebesar 70% untuk sentimen positif. Tempat wisata di Pulau Lombok yang popular berdasarkan kategori pantai adalah pantai Kuta dengan persentase sebesar 85%, kategori gili adalah Gilair dengan persentase sebesar 47%, dan kategori gunung adalah Gunung Rinjani dengan presentase sebesar 60%. |

BAB III

METODOLOGI PENELITIAN DAN PENGEMBANGAN SISTEM

3.1 Metodologi Penelitian

Metodologi penelitian merupakan cara ilmiah yang digunakan untuk mencapai tujuan dan menentukan jawaban dari permasalahan yang akan diselesaikan. Metodologi penelitian yang digunakan pada penelitian ini adalah metodologi penelitian kuantitatif. Metodologi penelitian kuantitatif merupakan metodologi penelitian yang digunakan untuk menjawab permasalahan yang berkaitan dengan data berupa angka dan data kuantitatif secara lebih terperinci menggunakan analisa statistik, meneliti populasi dan sampel tertentu, instrumen penelitian, teknik pengumpulan data, dan analisis data dalam suatu proposal atau laporan penelitian sesuai dengan kaidah penulisan karya ilmiah yang disyaratkan (Harahap & Lubis, 2019). Tahapan metodologi penelitian yang akan dilakukan antara lain pengumpulan data, *data labelling*, *text preprocessing*, ekstraksi fitur (*term weighting TF-IDF*), seleksi fitur, *support vector machine*, validasi dan pengujian model. Adapun detail dari setiap tahapan metodologi penelitian yang dapat dilihat pada gambar 3.1.



Gambar 3.1 Metodologi Penelitian

3.1.1 Pengumpulan data

Pengumpulan data bertujuan untuk mendapatkan informasi mengenai apa saja yang dibutuhkan untuk membantu proses penelitian ini. Data yang dikumpulkan pada penelitian ini merupakan data yang dapat dipertanggung jawabkan keasliannya. Pada penelitian ini proses pengumpulan data dibagi menjadi dua tahapan yaitu studi literatur dan *web scraping*.

3.1.1.1 Studi literatur

Studi literatur merupakan proses mengumpulkan jurnal, buku, artikel terkait mengenai penelitian ini. Studi literatur juga bertujuan untuk memberikan informasi mengenai metode yang digunakan dalam penelitian ini, sehingga dapat manambah pemahaman dan pengembangan dalam penelitian ini. Studi literatur yang digunakan dalam penelitian ini diambil dari berbagai sumber yang dapat dipercaya dan dipertanggung jawabkan.

3.1.1.2 Web scraping

Web scraping merupakan sebuah proses mengekstraksi data dari *website* dan menyimpannya ke dalam sebuah file atau ke dalam *database* yang dilakukan melalui kode program. *Web scraping* dapat mempermudah pengambilan data karena tidak perlu menyalin data satu persatu secara manual sehingga lebih cepat. *Web scraping* pada penelitian ini dilakukan dengan mengambil data yang terdapat didalam fitur ulasan di *google maps* dengan teknik *web scraper* yang dilakukan menggunakan bahasa pemrograman python dengan *library selenium* dan dengan bantuan *chromedriver* untuk pengambilan data latihnya, sedangkan untuk penambahan data sentimen baru menggunakan bahasan pemrograman php dengan *library curl*. Data latih yang diambil sebanyak 100 komentar untuk setiap pariwisata. Data daftar pariwisata di Kota Yogyakarta yang akan dianalisis sentimen diambil dari situs *visitjogja.com* (situs resmi Dinas Pariwisata Daerah Istimewa Yogyakarta) yaitu sebanyak 27 pariwisata. Data latih yang diambil digunakan untuk pembuatan model dan sekaligus digunakan sebagai data uji untuk menguji model yang telah dibuat. Berikut ini merupakan contoh data ulasan yang telah diambil dapat dilihat pada tabel 3.1.

Tabel 3.1. Contoh Data Ulasan

| Ulasan |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Sebenarnya tempatnya banyak barang,, barang sejarah, tapi sayang seperti tidak terawat banyak sampah daun2 dimana,, mana, tidak ada pemandu juga untuk menjelaskan satu sama lain. |
| Semakin usang, kotor, dan tidak terawat,, pemerintah harus lebih memperhatikan pemeliharaan dan penataan ruangannya,, |
| sayang pas kesana kondisinya kotor banyak daun2 kering.. Terus harga tiketnya juga gak sesuai dengan yang tertulis di depan. MAHAL SEKALI!!!! |
| Tujuan wisata yg tdk boleh dilewatkan saaaaat berwisata ke jogja? disini kita bisa tahu dan belajar sejarah dan adat serta budaya jogjakarta, pemandunya ramah, penjelasannya sangat jelas dan informatif |
| Bersih, Bagus tempat wisatanya. Pelayanan pemandunya sangat Mantap, Ramah2 Orang jualan di Luar |
| Tempatnya bersih,,, ketika datang langsung disambut dan dipandu mengelilingi keraton sehingga dapat menambah ilmu dan wawasan,,, |
| Destinasi wisata wajib kalau anda datang ke Jogja suasana dan bangunan yang klasik membuat anda seakan akan berada di zaman jawa kuno, indah banget pokoknya, pemandunya juga ramah dan sopan |

3.1.2 Data labeling

Data labeling merupakan tahapan memberikan label pada dataset secara manual. Semua dataset yang digunakan akan diberi label berdasarkan kelas sentimen positif atau negatif. Jumlah dataset yang akan diberi label sebanyak 2700 data ulasan. Proses *labeling* dataset ini akan dilakukan oleh 14 orang responden yang terpilih, masing-masing responden akan melabeli minimal sebanyak 100 data ulasan. Sehingga diperoleh sebanyak 2366 data ulasan merupakan kelas positif dan 334 data ulasan merupakan kelas negatif. Berikut ini merupakan contoh data ulasan yang sudah diberi label dapat dilihat pada tabel 3.2.

Tabel 3.2. Contoh Data Ulasan Yang Sudah Diberi Label

| No | Ulasan | Kelas |
|----|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------|
| D1 | Sebenarnya tempatnya banyak barang,, barang sejarah, tapi sayang seperti tidak terawat banyak sampah daun2 dimana,, mana, tidak ada pemandu juga untuk menjelaskan satu sama lain. | negatif |
| D2 | Semakin usang, kotor, dan tidak terawat,, pemerintah harus lebih memperhatikan pemeliharaan dan penataan ruangannya.,, | negatif |
| D3 | sayang pas kesana kondisinya kotor banyak daun2 kering.. Terus harga tiketnya juga gak sesuai dengan yang tertulis di depan. MAHAL SEKALI!!!! | negatif |
| D4 | Tujuan wisata yg tdk boleh dilewatkan saaaat berwisata ke jogja? disini kita bisa tahu dan belajar sejarah dan adat serta budaya jogjakarta, pemandunya ramah, penjelasannya sangat jelas dan informatif | positif |
| D5 | Bersih, Bagus tempat wisatanya. Pelayanan pemandunya sangat Mantap, Ramah2 Orang jualan di Luar | positif |
| D6 | Tempatnya bersih,,, ketika datang langsung disambut dan dipandu mengelilingi keraton sehingga dapat menambah ilmu dan wawasan,,, | positif |
| D7 | Destinasi wisata wajib kalau anda datang ke Jogja suasana dan bangunan yang klasik membuat anda seakan akan berada di zaman jawa kuno, indah banget pokoknya, pemandunya juga ramah dan sopan | positif |

3.1.3 Text preprocessing

Text preprocessing merupakan proses awal dalam *text mining* sebelum masuk ke proses klasifikasi. *Text preprocessing* merupakan proses yang bertujuan untuk mengubah dokumen teks semi-terstruktur atau tidak terstruktur menjadi dokumen teks terstruktur, dengan cara menyamakan struktur serta mengurangi dimensi yang melebar. Pada penelitian ini *text preprocessing* dibagi menjadi delapan tahapan antara lain *case folding*, *remove punctuation*, *remove number*, *remove repeated characters*, *spelling normalization*, *tokenizing*, *filtering* dan *stemming*. Berikut ini merupakan contoh hasil *text preprocessing* yang dapat dilihat pada tabel 3.3.

Tabel 3.3. Contoh Hasil Text Preprocessing

| No | Hasil Text Preprocessing |
|----|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| D1 | ['tempat', 'barang', 'barang', 'sejarah', 'sayang', 'awat', 'sampah', 'daun', 'mana', 'pandu'] |
| D2 | ['usang', 'kotor', 'awat', 'perintah', 'perhati', 'pelihara', 'tata', 'ruang'] |
| D3 | ['sayang', 'pas', 'kesana', 'kondisi', 'kotor', 'daun', 'kering', 'harga', 'tiket', 'sesuai', 'tulis', 'mahal'] |
| D4 | ['tuju', 'wisata', 'lewat', 'wisata', 'yogyakarta', 'ajar', 'sejarah', 'adat', 'budaya', 'yogyakarta', 'pandu', 'ramah', 'jelas', 'informatif'] |
| D5 | ['bersih', 'bagus', 'wisata', 'layan', 'pandu', 'mantap', 'ramah', 'orang', 'jual'] |
| D6 | ['tempat', 'bersih', 'langsung', 'sambut', 'pandu', 'keliling', 'keraton', 'tambah', 'ilmu', 'wawas'] |
| D7 | ['destinasi', 'wisata', 'wajib', 'yogyakarta', 'suasana', 'bangun', 'klasik', 'akan', 'zaman', 'jawa', 'kuno', 'indah', 'banget', 'pokok', 'pandu', 'ramah', 'sopan'] |

3.1.3.1 Case folding

case folding merupakan proses mengubah huruf *uppercase* (huruf kapital) menjadi huruf *lowercase* (huruf kecil). *Case folding* diperlukan karena tidak semua data dalam media sosial konsisten untuk menggunakan huruf kecil. Berikut ini merupakan contoh hasil *case folding* yang dapat dilihat pada tabel 3.4.

Tabel 3.4. Contoh Hasil Case Folding

| No | Hasil Case Folding |
|----|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| D1 | sebenarnya tempatnya banyak barang,, barang sejarah, tapi sayang seperti tidak terawat banyak sampah dedaunan dimana,, mana, tidak ada pemandu juga untuk menjelaskan satu sama lain |
| D2 | semakin usang, kotor, dan tidak terawat,, pemerintah harus lebih memperhatikan pemeliharaan dan penataan ruangannya.,, |
| D3 | sayang pas kesana kondisinya kotor banyak daun2 kering.. terus harga tiketnya juga gak sesuai dengan yang tertulis di depan. mahal sekali!!!! |

Tabel 3.5. Lanjutan Contoh Hasil Case Folding

| No | Hasil Case Folding |
|----|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| D4 | tujuan wisata yg tdk boleh dilewatkanaaaat berwisata ke jogja? disini kita bisa tahu dan belajar sejarah dan adat serta budaya jogjakarta, pemandunya ramah, penjelasannya sangat jelas dan informatif |
| D5 | bersih, bagus tempat wisatanya. pelayanan pemandunya sangat mantap, ramah2 orang jualan di luar |
| D6 | tempatnya bersih,,, ketika datang langsung disambut dan dipandu mengelilingi keraton sehingga dapat menambah ilmu dan wawasan,,. |
| D7 | destinasi wisata wajib kalau anda datang ke jogja suasana dan bangunan yang klasik membuat anda seakan akan berada di zaman jawa kuno, indah banget pokoknya, pemandunya juga ramah dan sopan |

3.1.3.2 Remove punctuation

Remove punctuation merupakan proses untuk menghilangkan tanda baca pada teks dokumen yang bertujuan untuk membersihkan elemen yang tidak berpengaruh terhadap hasil klasifikasi. Elemen tanda baca yang dihapus antara lain tanda seru, titik, titik koma, slash, tanda petik satu, tanda petik dua (kutipan), strip, tanda tanya, tanda kurung, dan sebagainya. Berikut ini merupakan contoh hasil *remove punctuation* yang dapat dilihat pada tabel 3.6.

Tabel 3.6. Contoh Hasil Remove Punctuation

| No | Hasil Remove Punctuation |
|----|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| D1 | sebenarnya tempatnya banyak barang barang sejarah tapi sayang seperti tidak terawat banyak sampah daun2 dimana mana tidak ada pemandu juga untuk menjelaskan satu sama lain |
| D2 | semakin usang kotor dan tidak terawat pemerintah harus lebih memperhatikan pemeliharaan dan penataan ruangannya |
| D3 | sayang pas kesana kondisinya kotor banyak daun2 kering terus harga tiketnya juga gak sesuai dengan yang tertulis di depan mahal sekali |
| D4 | tujuan wisata yg tdk boleh dilewatkanaaaat berwisata ke jogja disini kita bisa tahu dan belajar sejarah dan adat serta budaya jogjakarta pemandunya ramah penjelasannya sangat jelas dan informatif |
| D5 | bersih bagus tempat wisatanya pelayanan pemandunya sangat mantap ramah2 orang jualan di luar |
| D6 | tempatnya bersih ketika datang langsung disambut dan dipandu mengelilingi keraton sehingga dapat menambah ilmu dan wawasan |
| D7 | destinasi wisata wajib kalau anda datang ke jogja suasana dan bangunan yang klasik membuat anda seakan akan berada di zaman jawa kuno indah banget pokoknya pemandunya juga ramah dan sopan |

3.1.3.3 Remove number

Remove number adalah proses penghapusan karakter angka di dalam teks dokumen. *Remove number* dilakukan karena angka dianggap *delimiter* atau tidak berpengaruh terhadap hasil klasifikasi bahkan dapat mengganggu proses klasifikasi. Berikut ini merupakan contoh hasil *remove number* yang dapat dilihat pada tabel 3.7.

Tabel 3.7. Contoh Hasil Remove Number

| No | Hasil Remove Number |
|----|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| D1 | sebenarnya tempatnya banyak barang barang sejarah tapi sayang seperti tidak terawat banyak sampah daun dimana mana tidak ada pemandu juga untuk menjelaskan satu sama lain |
| D2 | semakin usang kotor dan tidak terawat pemerintah harus lebih memperhatikan pemeliharaan dan penataan ruangannya |
| D3 | sayang pas kesana kondisinya kotor banyak daun kering terus harga tiketnya juga gak sesuai dengan yang tertulis di depan mahal sekali |
| D4 | tujuan wisata yg tdk boleh dilewatkanaaaat berwisata ke jogja disini kita bisa tahu dan belajar sejarah dan adat serta budaya jogjakarta pemandunya ramah penjelasannya sangat jelas dan informatif |
| D5 | bersih bagus tempat wisatanya pelayanan pemandunya sangat mantap ramah orang jualan di luar |
| D6 | tempatnya bersih ketika datang langsung disambut dan dipandu mengelilingi keraton sehingga dapat menambah ilmu dan wawasan |

Tabel 3.8. Lanjutan Contoh Hasil Remove Number

| No | Hasil Remove Number |
|----|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| D7 | destinasi wisata wajib kalau anda datang ke jogja suasana dan bangunan yang klasik membuat anda seakan akan berada di zaman jawa kuno indah banget pokoknya pemandunya juga ramah dan sopan |

3.1.3.4 Remove repeated characters

Remove repeated characters adalah proses penghapus karakter yang berulang didalam teks dokumen. Karakter berulang yang dihapus adalah huruf yang memiliki lebih dari dua karakter, dan akan disisakan dua karakter saja. Berikut ini merupakan contoh hasil *remove repeated characters* yang dapat dilihat pada tabel 3.9.

Tabel 3.9. Contoh Hasil Remove Repeated Characters

| No | Hasil Remove Repeated Characters |
|----|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| D1 | sebenarnya tempatnya banyak barang barang sejarah tapi sayang seperti tidak terawat banyak sampah daun dimana mana tidak ada pemandu juga untuk menjelaskan satu sama lain |
| D2 | semakin usang kotor dan tidak terawat pemerintah harus lebih memperhatikan pemeliharaan dan penataan ruangannya |
| D3 | sayang pas kesana kondisinya kotor banyak daun kering terus harga tiketnya juga gak sesuai dengan yang tertulis di depan mahal sekali |
| D4 | tujuan wisata yg tdk boleh dilewatkan saat berwisata ke jogja disini kita bisa tahu dan belajar sejarah dan adat serta budaya jogjakarta pemandunya ramah penjelasannya sangat jelas dan informatif |
| D5 | bersih bagus tempat wisatanya pelayanan pemandunya sangat mantap ramah orang jualan di luar |
| D6 | tempatnya bersih ketika datang langsung disambut dan dipandu mengelilingi keraton sehingga dapat menambah ilmu dan wawasan |
| D7 | destinasi wisata wajib kalau anda datang ke jogja suasana dan bangunan yang klasik membuat anda seakan akan berada di zaman jawa kuno indah banget pokoknya pemandunya juga ramah dan sopan |

3.1.3.5 Spelling normalization

Spelling normalization merupakan proses perbaikan kata yang salah eja maupun singkatan (kata tidak baku) menjadi bentuk kata aslinya (kata baku). *Spelling normalization* dilakukan untuk menghindari dimensi kata yang melebar, karena kesalahan ketik pada penulisan atau singkatan, padahal kata tersebut memiliki arti yang sama. Jika tidak dilakukan *spelling normalization* maka kata tersebut akan dianggap sebuah entitas yang berbeda pada penyusunan matriks. Pada penelitian ini menggunakan daftar kamus *spelling normalization* dari penelitian (Noehilasari, 2014) dan ditambah secara manual dari pengamatan data latih yang digunakan pada penelitian ini. Berikut ini merupakan contoh hasil *spelling normalization* yang dapat dilihat pada tabel 3.10.

Tabel 3.10. Contoh Hasil Spelling Normalization

| No | Hasil Spelling Normalization |
|----|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| D1 | sebenarnya tempatnya banyak barang barang sejarah tapi sayang seperti tidak terawat banyak sampah daun dimana mana tidak ada pemandu juga untuk menjelaskan satu sama lain |
| D2 | semakin usang kotor dan tidak terawat pemerintah harus lebih memperhatikan pemeliharaan dan penataan ruangannya |
| D3 | sayang pas kesana kondisinya kotor banyak daun kering terus harga tiketnya juga tidak sesuai dengan yang tertulis di depan mahal sekali |
| D4 | tujuan wisata yang tidak boleh dilewatkan saat berwisata ke yogyakarta disini kita bisa tahu dan belajar sejarah dan adat serta budaya yogyakarta pemandunya ramah penjelasannya sangat jelas dan informatif |
| D5 | bersih bagus tempat wisatanya pelayanan pemandunya sangat mantap ramah orang jualan di luar |
| D6 | tempatnya bersih ketika datang langsung disambut dan dipandu mengelilingi keraton sehingga dapat menambah ilmu dan wawasan |
| D7 | destinasi wisata wajib kalau anda datang ke yogyakarta suasana dan bangunan yang klasik membuat anda seakan akan berada di zaman jawa kuno indah banget pokoknya pemandunya juga ramah dan sopan |

3.1.3.6 Tokenizing

Tokenizing adalah proses pemotongan *string input* menjadi beberapa kata berdasarkan tiap kata yang menyusunnya. Pada proses ini, data teks dokumen akan dipisah menjadi tiap kata berdasarkan spasi atau karakter lain sehingga menghasilkan *token*. Berikut ini merupakan contoh hasil *tokenizing* yang dapat dilihat pada tabel 3.11.

Tabel 3.11. Contoh Hasil Tokenizing

| No | Hasil Tokenizing |
|----|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| D1 | ['sebenarnya', 'tempatnya', 'banyak', 'barang', 'barang', 'sejarah', 'tapi', 'sayang', 'seperti', 'tidak', 'terawat', 'banyak', 'sampah', 'daun', 'dimana', 'mana', 'tidak', 'ada', 'pemandu', 'juga', 'untuk', 'menjelaskan', 'satu', 'sama', 'lain'] |
| D2 | ['semakin', 'usang', 'kotor', 'dan', 'tidak', 'terawat', 'pemerintah', 'harus', 'lebih', 'memperhatikan', 'pemeliharaan', 'dan', 'penataan', 'ruangannya'] |
| D3 | ['sayang', 'pas', 'kesana', 'kondisinya', 'kotor', 'banyak', 'daun', 'kering', 'terus', 'harga', 'tiketnya', 'juga', 'tidak', 'sesuai', 'dengan', 'yang', 'tertulis', 'di', 'depan', 'mahal', 'sekali'] |
| D4 | ['tujuan', 'wisata', 'yang', 'tidak', 'boleh', 'dilewatkan', 'saat', 'berwisata', 'ke', 'yogyakarta', 'disini', 'kita', 'bisa', 'tahu', 'dan', 'belajar', 'sejarah', 'dan', 'adat', 'serta', 'budaya', 'yogyakarta', 'pemandunya', 'ramah', 'penjelasannya', 'sangat', 'jelas', 'dan', 'informatif'] |
| D5 | ['bersih', 'bagus', 'tempat', 'wisatanya', 'pelayanan', 'pemandunya', 'sangat', 'mantap', 'ramah', 'orang', 'jualan', 'di', 'luar'] |
| D6 | ['tempatnya', 'bersih', 'ketika', 'datang', 'langsung', 'disambut', 'dan', 'dipandu', 'mengelilingi', 'keraton', 'sehingga', 'dapat', 'menambah', 'ilmu', 'dan', 'wawasan'] |
| D7 | ['destinasi', 'wisata', 'wajib', 'kalau', 'anda', 'datang', 'ke', 'yogyakarta', 'suasana', 'dan', 'bangunan', 'yang', 'klasik', 'membuat', 'anda', 'seakan', 'akan', 'berada', 'di', 'zaman', 'jawa', 'kuno', 'indah', 'banget', 'pokoknya', 'pemandunya', 'juga', 'ramah', 'dan', 'sopan'] |

3.1.3.7 Filtering

Filtering adalah proses mengambil kata-kata penting yang dihasilkan oleh *token*. Pada penelitian ini akan menggunakan algoritma *stopword removal* untuk membuang kata-kata yang tidak penting yang didapat dari hasil token seperti “yang”, “di”, “dari”, “untuk”, “ke” yang memiliki frekuensi kemunculan yang tinggi dalam sebuah teks dokumen. Berikut ini merupakan contoh hasil *filtering* yang dapat dilihat pada tabel 3.12.

Tabel 3.12. Contoh Hasil Filtering

| No | Hasil Filtering |
|----|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| D1 | ['tempatnya', 'barang', 'barang', 'sejarah', 'sayang', 'terawat', 'sampah', 'daun', 'dimana', 'pemandu'] |
| D2 | ['usang', 'kotor', 'terawat', 'pemerintah', 'memperhatikan', 'pemeliharaan', 'penataan', 'ruangannya'] |
| D3 | ['sayang', 'pas', 'kesana', 'kondisinya', 'kotor', 'daun', 'kering', 'harga', 'tiketnya', 'sesuai', 'tertulis', 'mahal'] |
| D4 | ['tujuan', 'wisata', 'dilewatkan', 'berwisata', 'yogyakarta', 'belajar', 'sejarah', 'adat', 'budaya', 'yogyakarta', 'pemandunya', 'ramah', 'penjelasannya', 'informatif'] |
| D5 | ['bersih', 'bagus', 'wisatanya', 'pelayanan', 'pemandunya', 'mantap', 'ramah', 'orang', 'jualan'] |
| D6 | ['tempatnya', 'bersih', 'langsung', 'disambut', 'dipandu', 'mengelilingi', 'keraton', 'menambah', 'ilmu', 'wawasan'] |
| D7 | ['destinasi', 'wisata', 'wajib', 'yogyakarta', 'suasana', 'bangunan', 'klasik', 'seakan', 'akan', 'berada', 'zaman', 'jawa', 'kuno', 'indah', 'banget', 'pokoknya', 'pemandunya', 'ramah', 'sopan'] |

3.1.3.8 Stemming

Stemming adalah tahapan untuk mengubah kata atau token hasil *filtering* menjadi kata dasarnya dengan menghilangkan imbuhan pada kata tersebut. Algoritma *stemming* yang digunakan pada penelitian adalah algoritma *stemming* Nazief dan Adriani. Berikut ini merupakan contoh hasil *stemming* yang dapat dilihat pada tabel 3.13.

Tabel 3.13. Contoh Hasil Stemming

| No | Hasil Stemming |
|----|------------------------------------------------------------------------------------------------|
| D1 | ['tempat', 'barang', 'barang', 'sejarah', 'sayang', 'awat', 'sampah', 'daun', 'mana', 'pandu'] |
| D2 | ['usang', 'kotor', 'awat', 'perintah', 'perhati', 'pelihara', 'tata', 'ruang'] |

Tabel 3.14. Lanjutan Contoh Hasil Stemming

| No | Hasil Stemming |
|----|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| D3 | ['sayang', 'pas', 'kesana', 'kondisi', 'kotor', 'daun', 'kering', 'harga', 'tiket', 'sesuai', 'tulis', 'mahal'] |
| D4 | ['tuju', 'wisata', 'lewat', 'wisata', 'yogyakarta', 'ajar', 'sejarah', 'adat', 'budaya', 'yogyakarta', 'pandu', 'ramah', 'jelas', 'informatif'] |
| D5 | ['bersih', 'bagus', 'wisata', 'layan', 'pandu', 'mantap', 'ramah', 'orang', 'jual'] |
| D6 | ['tempat', 'bersih', 'langsung', 'sambut', 'pandu', 'keliling', 'keraton', 'tambah', 'ilmu', 'wawas'] |
| D7 | ['destinasi', 'wisata', 'wajib', 'yogyakarta', 'suasana', 'bangun', 'klasik', 'akan', 'zaman', 'jawa', 'kuno', 'indah', 'banget', 'pokok', 'pandu', 'ramah', 'sopan'] |

3.1.4 Ekstraksi fitur (*term weighting TF-IDF*)

Pada tahapan ini akan dilakukan pembobotan fitur atau kata dengan metode TF-IDF. Tahapan ini dilakukan setelah dilakukannya *text preprocessing*. Pada tahap ini, setiap fitur akan dihitung frekuensi kemunculannya dalam dokumen yang disebut *term frequency* (TF). Kemudian, akan dihitung nilai *inverse document frequency* (IDF), dan yang terakhir akan dihitung nilai *term frequency-inverse document frequency* (TF-IDF). TF-IDF merupakan penggabungan dari perhitungan TF murni dengan IDF. Berikut ini merupakan contoh data ulasan yang sudah melalui proses *text preprocessing* yang akan dihitung nilai TF-IDF untuk setiap fiturnya :

- D1 : tempat barang barang sejarah sayang awat sampah daun mana pandu (kelas sentimen negatif)
- D2 : usang kotor awat perintah perhati pelihara tata ruang (kelas sentimen negatif)
- D3 : sayang pas kesana kondisi kotor daun kering harga tiket sesuai tulis mahal (kelas sentimen negatif)
- D4 : tuju wisata lewat wisata yogyakarta ajar sejarah adat budaya yogyakarta pandu ramah jelas informatif (kelas sentimen positif)
- D5 : bersih bagus wisata layan pandu mantap ramah orang jual (kelas sentimen positif)
- D6 : tempat bersih langsung sambut pandu keliling keraton tambah ilmu wawas (kelas sentimen positif)
- D7 : destinasi wisata wajib yogyakarta suasana bangun klasik akan zaman jawa kuno indah banget pokok pandu ramah sopan (kelas sentimen positif)

Dari data ulasan di atas maka langkah selanjutnya menghitung nilai *term frequency* (TF) setiap fitur, misalkan menghitung TF pada *term “tempat”* sebagai berikut :

$$D1 : tf_{tempat,D1} = 1$$

$$D2 : tf_{tempat,D2} = 0$$

$$D3 : tf_{tempat,D3} = 0$$

$$D4 : tf_{tempat,D4} = 0$$

$$D5 : tf_{tempat,D5} = 0$$

$$D6 : tf_{tempat,D6} = 1$$

$$D7 : tf_{tempat,D7} = 0$$

$$DF_{tempat} = 2$$

Setelah semua nilai TF dan DF fitur “tempat” telah diketahui, maka langkah selanjutnya menghitung nilai IDF setiap fitur menggunakan persamaan 2.3, misalkan menghitung IDF pada *term “tempat”* sebagai berikut :

$$idf_{tempat} = \log(7/2) = 0.544068$$

Selanjutnya hasil perhitungan TF, DF dan IDF untuk semua *term* atau fitur dapat dilihat pada tabel 3.15.

Tabel 3.15. Hasil Perhitungan TF, DF, IDF Semua Fitur

| No | Fitur | TF | | | | | | | DF | IDF |
|----|------------|----|----|----|----|----|----|----|----|----------|
| | | D1 | D2 | D3 | D4 | D5 | D6 | D7 | | |
| 1 | tempat | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 2 | 0.544068 |
| 2 | barang | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 2 | 0.544068 |
| 3 | sejarah | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 2 | 0.544068 |
| 4 | sayang | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 3 | 0.367977 |
| 5 | awat | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0.845098 |
| 6 | sampah | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0.845098 |
| 7 | daun | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 2 | 0.544068 |
| 8 | mana | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0.845098 |
| 9 | pandu | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 5 | 0.146128 |
| 10 | usang | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0.845098 |
| 11 | kotor | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 2 | 0.544068 |
| 12 | perintah | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0.845098 |
| 13 | perhati | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0.845098 |
| 14 | pelihara | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0.845098 |
| 15 | tata | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0.845098 |
| 16 | ruang | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0.845098 |
| 17 | pas | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0.845098 |
| 18 | kesana | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0.845098 |
| 19 | kondisi | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0.845098 |
| 20 | kering | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0.845098 |
| 21 | harga | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0.845098 |
| 22 | tiket | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0.845098 |
| 23 | sesuai | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0.845098 |
| 24 | tulis | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0.845098 |
| 25 | mahal | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0.845098 |
| 26 | tuju | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0.845098 |
| 27 | wisata | 0 | 0 | 0 | 2 | 1 | 0 | 1 | 4 | 0.243038 |
| 28 | lewat | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0.845098 |
| 29 | yogyakarta | 0 | 0 | 0 | 2 | 0 | 0 | 1 | 3 | 0.367977 |
| 30 | ajar | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0.845098 |
| 31 | adat | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0.845098 |
| 32 | budaya | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0.845098 |
| 33 | ramah | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 3 | 0.367977 |
| 34 | jelas | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0.845098 |
| 35 | informatif | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0.845098 |
| 36 | bersih | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 2 | 0.544068 |
| 37 | bagus | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0.845098 |
| 38 | layan | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0.845098 |
| 39 | mantap | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0.845098 |
| 40 | orang | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0.845098 |
| 41 | jual | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0.845098 |
| 42 | langsung | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0.845098 |
| 43 | sambut | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0.845098 |
| 44 | keliling | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0.845098 |
| 45 | keraton | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0.845098 |
| 46 | tambah | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0.845098 |
| 47 | ilmu | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0.845098 |
| 48 | wawas | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0.845098 |
| 49 | destinasi | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0.845098 |
| 50 | wajib | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0.845098 |
| 51 | suasana | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0.845098 |

Tabel 3.16. Lanjutan Hasil Perhitungan TF, DF, IDF Semua Fitur

| No | Fitur | TF | | | | | | | DF | IDF |
|----|--------|----|----|----|----|----|----|----|----|----------|
| | | D1 | D2 | D3 | D4 | D5 | D6 | D7 | | |
| 52 | bangun | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0.845098 |
| 53 | klasik | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0.845098 |
| 54 | akan | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0.845098 |
| 55 | zaman | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0.845098 |
| 56 | jawa | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0.845098 |
| 57 | kuno | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0.845098 |
| 58 | indah | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0.845098 |
| 59 | banget | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0.845098 |
| 60 | pokok | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0.845098 |
| 61 | sopan | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0.845098 |

Setelah nilai IDF setiap fitur telah diketahui, maka langkah selanjutnya menghitung nilai TF-IDF pada setiap fitur menggunakan persamaan 2.4, misalkan menghitung TF-IDF pada *term* “tempat” sebagai berikut :

$$D1 : tf.idf_{tempat,D1} = 1 \times 0.544068 = 0.544068$$

$$D2 : tf.idf_{tempat,D2} = 0 \times 0.544068 = 0$$

$$D3 : tf.idf_{tempat,D3} = 0 \times 0.544068 = 0$$

$$D4 : tf.idf_{tempat,D4} = 0 \times 0.544068 = 0$$

$$D5 : tf.idf_{tempat,D5} = 0 \times 0.544068 = 0$$

$$D6 : tf.idf_{tempat,D6} = 1 \times 0.544068 = 0$$

$$D7 : tf.idf_{tempat,D7} = 0 \times 0.544068 = 0$$

Berikut ini merupakan hasil perhitungan ekstraksi fitur *term weighting* TF-IDF untuk semua fitur atau *term* pada tujuh dokumen di atas :

Tabel 3.17. Hasil Perhitungan TF-IDF Semua Fitur

| No | Fitur | TF-IDF | | | | | | |
|----|----------|----------|----------|----------|----------|----------|----------|----------|
| | | D1 | D2 | D3 | D4 | D5 | D6 | D7 |
| 1 | tempat | 0.544068 | 0 | 0 | 0 | 0 | 0.544068 | 0 |
| 2 | barang | 0.544068 | 0 | 0 | 0 | 0 | 0 | 0 |
| 3 | sejarah | 0.544068 | 0 | 0 | 0.544068 | 0 | 0 | 0 |
| 4 | sayang | 0.367977 | 0.367977 | 0.367977 | 0 | 0 | 0 | 0 |
| 5 | awat | 0.845098 | 0 | 0 | 0 | 0 | 0 | 0 |
| 6 | sampah | 0.845098 | 0 | 0 | 0 | 0 | 0 | 0 |
| 7 | daun | 0.544068 | 0 | 0.544068 | 0 | 0 | 0 | 0 |
| 8 | mana | 0.845098 | 0 | 0 | 0 | 0 | 0 | 0 |
| 9 | pandu | 0.146128 | 0 | 0 | 0.146128 | 0.146128 | 0.146128 | 0.146128 |
| 10 | usang | 0.845098 | 0 | 0 | 0 | 0 | 0 | 0 |
| 11 | kotor | 0 | 0.544068 | 0.544068 | 0 | 0 | 0 | 0 |
| 12 | perintah | 0 | 0.845098 | 0 | 0 | 0 | 0 | 0 |
| 13 | perhati | 0 | 0.845098 | 0 | 0 | 0 | 0 | 0 |
| 14 | pelihara | 0 | 0.845098 | 0 | 0 | 0 | 0 | 0 |
| 15 | tata | 0 | 0.845098 | 0 | 0 | 0 | 0 | 0 |
| 16 | ruang | 0 | 0.845098 | 0 | 0 | 0 | 0 | 0 |
| 17 | pas | 0 | 0 | 0.845098 | 0 | 0 | 0 | 0 |
| 18 | kesana | 0 | 0 | 0.845098 | 0 | 0 | 0 | 0 |
| 19 | kondisi | 0 | 0 | 0.845098 | 0 | 0 | 0 | 0 |
| 20 | kering | 0 | 0 | 0.845098 | 0 | 0 | 0 | 0 |
| 21 | harga | 0 | 0 | 0.845098 | 0 | 0 | 0 | 0 |
| 22 | tiket | 0 | 0 | 0.845098 | 0 | 0 | 0 | 0 |
| 23 | sesuai | 0 | 0 | 0.845098 | 0 | 0 | 0 | 0 |
| 24 | Tulis | 0 | 0 | 0.845098 | 0 | 0 | 0 | 0 |

Tabel 3.18. Lanjutan Hasil Perhitungan TF-IDF Semua Fitur

| No | Fitur | TF-IDF | | | | | |
|----|------------|--------|----|----------|----------|----------|----------|
| | | D1 | D2 | D3 | D4 | D5 | D7 |
| 25 | mahal | 0 | 0 | 0.845098 | 0 | 0 | 0 |
| 26 | tuju | 0 | 0 | 0 | 0.845098 | 0 | 0 |
| 27 | wisata | 0 | 0 | 0 | 0.243038 | 0.243038 | 0 |
| 28 | lewat | 0 | 0 | 0 | 0.845098 | 0 | 0 |
| 29 | yogyakarta | 0 | 0 | 0 | 0.367977 | 0 | 0.367977 |
| 30 | ajar | 0 | 0 | 0 | 0.845098 | 0 | 0 |
| 31 | adat | 0 | 0 | 0 | 0.845098 | 0 | 0 |
| 32 | budaya | 0 | 0 | 0 | 0.845098 | 0 | 0 |
| 33 | ramah | 0 | 0 | 0 | 0.367977 | 0.367977 | 0 |
| 34 | jelas | 0 | 0 | 0 | 0.845098 | 0 | 0 |
| 35 | informatif | 0 | 0 | 0 | 0.845098 | 0 | 0 |
| 36 | bersih | 0 | 0 | 0 | 0 | 0.544068 | 0.544068 |
| 37 | bagus | 0 | 0 | 0 | 0 | 0.845098 | 0 |
| 38 | layan | 0 | 0 | 0 | 0 | 0.845098 | 0 |
| 39 | mantap | 0 | 0 | 0 | 0 | 0.845098 | 0 |
| 40 | orang | 0 | 0 | 0 | 0 | 0.845098 | 0 |
| 41 | jual | 0 | 0 | 0 | 0 | 0.845098 | 0 |
| 42 | langsung | 0 | 0 | 0 | 0 | 0 | 0.845098 |
| 43 | sambut | 0 | 0 | 0 | 0 | 0 | 0.845098 |
| 44 | keliling | 0 | 0 | 0 | 0 | 0 | 0.845098 |
| 45 | keraton | 0 | 0 | 0 | 0 | 0 | 0.845098 |
| 46 | tambah | 0 | 0 | 0 | 0 | 0 | 0.845098 |
| 47 | ilmu | 0 | 0 | 0 | 0 | 0 | 0.845098 |
| 48 | wawas | 0 | 0 | 0 | 0 | 0 | 0.845098 |
| 49 | destinasi | 0 | 0 | 0 | 0 | 0 | 0.845098 |
| 50 | wajib | 0 | 0 | 0 | 0 | 0 | 0.845098 |
| 51 | suasana | 0 | 0 | 0 | 0 | 0 | 0.845098 |
| 52 | bangun | 0 | 0 | 0 | 0 | 0 | 0.845098 |
| 53 | klasik | 0 | 0 | 0 | 0 | 0 | 0.845098 |
| 54 | akan | 0 | 0 | 0 | 0 | 0 | 0.845098 |
| 55 | zaman | 0 | 0 | 0 | 0 | 0 | 0.845098 |
| 56 | jawa | 0 | 0 | 0 | 0 | 0 | 0.845098 |
| 57 | kuno | 0 | 0 | 0 | 0 | 0 | 0.845098 |
| 58 | indah | 0 | 0 | 0 | 0 | 0 | 0.845098 |
| 59 | banget | 0 | 0 | 0 | 0 | 0 | 0.845098 |
| 60 | pokok | 0 | 0 | 0 | 0 | 0 | 0.845098 |
| 61 | sopan | 0 | 0 | 0 | 0 | 0 | 0.845098 |

3.1.5 Seleksi fitur

Seleksi fitur merupakan tahapan pemilihan fitur-fitur yang memiliki pengaruh paling tinggi terhadap pembangunan model klasifikasi metode SVM dan menghilangkan fitur yang kurang berpengaruh atau tidak berpotensi menjadi indeks. Penerapan tahapan seleksi fitur ini dapat mempercepat proses klasifikasi pada metode SVM karena fitur yang digunakan lebih sedikit. Tahapan proses seleksi fitur antara lain menghitung nilai bobot setiap fitur, lalu mengurutkan fitur dari bobot tertinggi ke terendah, kemudian menentukan berapa banyak fitur yang akan digunakan untuk menghilangkan fitur pengganggu dalam klasifikasi. Pada penelitian ini setiap fitur akan dihitung bobot nilainya menggunakan metode seleksi fitur *mutual information* dan *chi square*. Kedua metode tersebut kemudian akan dibandingkan hasil akurasinya, sehingga dapat diketahui metode seleksi fitur mana yang lebih baik dalam meningkatkan hasil akurasi dari klasifikasi metode SVM.

3.1.5.1 Chi square

Chi Square merupakan seleksi fitur yang menguji keterikatan hubungan atau pengaruh sebuah *term* dengan kategorinya. Pada metode seleksi fitur *chi square* pertama menempatkan fitur atau *term* yang akan dihitung nilai jumlah kemunculan katanya menggunakan tabel kontingensi untuk kelas positif dan kelas negatif. Kemudian hasil nilai *chi square* fitur dari kedua kelas tersebut akan dibandingkan, dan akan dipilih nilai *chi square* maksimal dari fitur tersebut. Kemudian setelah semua fitur telah diketahui nilai bobot *chi square*-nya, maka semua fitur akan diurutkan berdasarkan nilai bobot *chi square* tertinggi ke terendah. Kemudian, menentukan banyaknya fitur yang akan digunakan untuk proses klasifikasi. Berikut ini merupakan contoh menghitung nilai bobot *chi square* dari fitur “tempat”, misal terdapat nilai TF-IDF dari fitur “tempat”:

Tabel 3.19. Hasil Perhitungan Term Weighting TF-IDF Untuk Fitur Tempat.

| Dokumen | TF-IDF (tempat) | Kelas Sentimen |
|---------|-----------------|----------------|
| D1 | 0.544068 | negatif |
| D2 | 0 | negatif |
| D3 | 0 | negatif |
| D4 | 0 | positif |
| D5 | 0 | positif |
| D6 | 0.544068 | positif |
| D7 | 0 | positif |

Pertama, menghitung nilai kemunculan fitur “tempat” untuk kelas positif dan negatif, dengan menempatkan nilai jumlah kemunculan *term* atau kata menggunakan tabel kontingensi seperti berikut ini :

Tabel 3.20. Kontingensi Nilai Jumlah Kemunculan Term Tempat Untuk Kelas Positif.

| c = Positif t = “tempat” | ec = 1 | ec = 0 |
|-----------------------------|--------|--------|
| et = 1 | 1 | 1 |
| et = 0 | 3 | 2 |

Tabel 3.21 Kontingensi Nilai Jumlah Kemunculan Term Tempat Untuk Kelas Negatif.

| c = Negatif t = “tempat” | ec = 1 | ec = 0 |
|-----------------------------|--------|--------|
| et = 1 | 1 | 1 |
| et = 0 | 2 | 3 |

Kedua, menghitung nilai bobot *chi square* fitur “tempat” dengan menggunakan persamaan 2.5 pada kelas positif dan negatif sebagai berikut :

$$\begin{aligned}
 X^2(\text{tempat}, \text{positif}) &= \frac{7 \times (1 \times 2 - 3 \times 1)^2}{(1 + 3)(1 + 2)(1 + 1)(3 + 2)} \\
 &= \frac{7 \times (2 - 3)^2}{(4)(3)(2)(5)} \\
 &= \frac{7}{120} \\
 &= 0.058333
 \end{aligned}$$

$$\begin{aligned}
 X^2(\text{tempat}, \text{negatif}) &= \frac{7 \times (1 \times 3 - 2 \times 1)^2}{(1 + 2)(1 + 3)(1 + 1)(2 + 3)} \\
 &= \frac{7 \times (3 - 2)^2}{(3)(4)(2)(5)}
 \end{aligned}$$

$$\begin{aligned}
 &= \frac{7}{120} \\
 &= 0.058333
 \end{aligned}$$

$$X^2(\text{tempat}) = \text{Max}(X^2(\text{tempat, positif}), X^2(\text{tempat, negatif})) = 0.058333$$

Dari perhitungan di atas, diperoleh nilai bobot *chi square* fitur “tempat” adalah 0.058333. Berikut ini merupakan hasil perhitungan nilai bobot *chi square* untuk semua fitur yang dapat dilihat pada tabel 3.22.

Tabel 3.22. Hasil Perhitungan Nilai Bobot *Chi Square* Untuk Semua Fitur

| No | Fitur | Kelas = Positif | | | | Kelas = Negatif | | | | Bobot Chi Square | | Max Bobot Chi Square |
|----|------------|-----------------|---|---|---|-----------------|---|---|---|------------------|----------|----------------------|
| | | A | B | C | D | A | B | C | D | Positif | Negatif | |
| 1 | tempat | 1 | 1 | 3 | 2 | 1 | 1 | 2 | 3 | 0.058333 | 0.058333 | 0.058333333 |
| 2 | barang | 0 | 1 | 4 | 2 | 1 | 0 | 2 | 4 | 1.555556 | 1.555556 | 1.555555556 |
| 3 | sejarah | 1 | 1 | 3 | 2 | 1 | 1 | 2 | 3 | 0.058333 | 0.058333 | 0.058333333 |
| 4 | sayang | 0 | 3 | 4 | 0 | 3 | 0 | 0 | 4 | 7 | 7 | 7 |
| 5 | awat | 0 | 1 | 4 | 2 | 1 | 0 | 2 | 4 | 1.555556 | 1.555556 | 1.555555556 |
| 6 | sampah | 0 | 1 | 4 | 2 | 1 | 0 | 2 | 4 | 1.555556 | 1.555556 | 1.555555556 |
| 7 | daun | 0 | 2 | 4 | 1 | 2 | 0 | 1 | 4 | 3.733333 | 3.733333 | 3.733333333 |
| 8 | mana | 0 | 1 | 4 | 2 | 1 | 0 | 2 | 4 | 1.555556 | 1.555556 | 1.555555556 |
| 9 | pandu | 4 | 1 | 0 | 2 | 1 | 4 | 2 | 0 | 3.733333 | 3.733333 | 3.733333333 |
| 10 | usang | 0 | 1 | 4 | 2 | 1 | 0 | 2 | 4 | 1.555556 | 1.555556 | 1.555555556 |
| 11 | kotor | 0 | 2 | 4 | 1 | 2 | 0 | 1 | 4 | 3.733333 | 3.733333 | 3.733333333 |
| 12 | perintah | 0 | 1 | 4 | 2 | 1 | 0 | 2 | 4 | 1.555556 | 1.555556 | 1.555555556 |
| 13 | perhati | 0 | 1 | 4 | 2 | 1 | 0 | 2 | 4 | 1.555556 | 1.555556 | 1.555555556 |
| 14 | pelihara | 0 | 1 | 4 | 2 | 1 | 0 | 2 | 4 | 1.555556 | 1.555556 | 1.555555556 |
| 15 | tata | 0 | 1 | 4 | 2 | 1 | 0 | 2 | 4 | 1.555556 | 1.555556 | 1.555555556 |
| 16 | ruang | 0 | 1 | 4 | 2 | 1 | 0 | 2 | 4 | 1.555556 | 1.555556 | 1.555555556 |
| 17 | pas | 0 | 1 | 4 | 2 | 1 | 0 | 2 | 4 | 1.555556 | 1.555556 | 1.555555556 |
| 18 | kesana | 0 | 1 | 4 | 2 | 1 | 0 | 2 | 4 | 1.555556 | 1.555556 | 1.555555556 |
| 19 | kondisi | 0 | 1 | 4 | 2 | 1 | 0 | 2 | 4 | 1.555556 | 1.555556 | 1.555555556 |
| 20 | kering | 0 | 1 | 4 | 2 | 1 | 0 | 2 | 4 | 1.555556 | 1.555556 | 1.555555556 |
| 21 | harga | 0 | 1 | 4 | 2 | 1 | 0 | 2 | 4 | 1.555556 | 1.555556 | 1.555555556 |
| 22 | tiket | 0 | 1 | 4 | 2 | 1 | 0 | 2 | 4 | 1.555556 | 1.555556 | 1.555555556 |
| 23 | sesuai | 0 | 1 | 4 | 2 | 1 | 0 | 2 | 4 | 1.555556 | 1.555556 | 1.555555556 |
| 24 | tulis | 0 | 1 | 4 | 2 | 1 | 0 | 2 | 4 | 1.555556 | 1.555556 | 1.555555556 |
| 25 | mahal | 0 | 1 | 4 | 2 | 1 | 0 | 2 | 4 | 1.555556 | 1.555556 | 1.555555556 |
| 26 | tuju | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 | 0.875 | 0.875 | 0.875 |
| 27 | wisata | 3 | 0 | 1 | 3 | 0 | 3 | 3 | 1 | 3.9375 | 3.9375 | 3.9375 |
| 28 | lewat | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 | 0.875 | 0.875 | 0.875 |
| 29 | yogyakarta | 2 | 0 | 2 | 3 | 0 | 2 | 3 | 2 | 2.1 | 2.1 | 2.1 |
| 30 | ajar | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 | 0.875 | 0.875 | 0.875 |
| 31 | adat | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 | 0.875 | 0.875 | 0.875 |
| 32 | budaya | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 | 0.875 | 0.875 | 0.875 |
| 33 | ramah | 3 | 0 | 1 | 3 | 0 | 3 | 3 | 1 | 3.9375 | 3.9375 | 3.9375 |
| 34 | jelas | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 | 0.875 | 0.875 | 0.875 |
| 35 | informatif | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 | 0.875 | 0.875 | 0.875 |
| 36 | bersih | 2 | 0 | 2 | 3 | 0 | 2 | 3 | 2 | 2.1 | 2.1 | 2.1 |
| 37 | bagus | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 | 0.875 | 0.875 | 0.875 |
| 38 | layan | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 | 0.875 | 0.875 | 0.875 |
| 39 | mantap | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 | 0.875 | 0.875 | 0.875 |
| 40 | orang | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 | 0.875 | 0.875 | 0.875 |
| 41 | jual | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 | 0.875 | 0.875 | 0.875 |
| 42 | langsung | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 | 0.875 | 0.875 | 0.875 |
| 43 | sambut | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 | 0.875 | 0.875 | 0.875 |
| 44 | keliling | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 | 0.875 | 0.875 | 0.875 |

Tabel 3.23. Lanjutan Hasil Perhitungan Nilai Bobot Chi Square Untuk Semua Fitur

| No | Fitur | Kelas = Positif | | | | Kelas = Negatif | | | | Bobot Chi Square | | Max Bobot Chi Square |
|----|-----------|-----------------|---|---|---|-----------------|---|---|---|------------------|---------|----------------------|
| | | A | B | C | D | A | B | C | D | Positif | Negatif | |
| 45 | keraton | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 | 0.875 | 0.875 | 0.875 |
| 46 | tambah | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 | 0.875 | 0.875 | 0.875 |
| 47 | ilmu | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 | 0.875 | 0.875 | 0.875 |
| 48 | wawas | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 | 0.875 | 0.875 | 0.875 |
| 49 | destinasi | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 | 0.875 | 0.875 | 0.875 |
| 50 | wajib | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 | 0.875 | 0.875 | 0.875 |
| 51 | suasana | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 | 0.875 | 0.875 | 0.875 |
| 52 | bangun | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 | 0.875 | 0.875 | 0.875 |
| 53 | klasik | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 | 0.875 | 0.875 | 0.875 |
| 54 | akan | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 | 0.875 | 0.875 | 0.875 |
| 55 | zaman | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 | 0.875 | 0.875 | 0.875 |
| 56 | jawa | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 | 0.875 | 0.875 | 0.875 |
| 57 | kuno | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 | 0.875 | 0.875 | 0.875 |
| 58 | indah | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 | 0.875 | 0.875 | 0.875 |
| 59 | banget | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 | 0.875 | 0.875 | 0.875 |
| 60 | pokok | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 | 0.875 | 0.875 | 0.875 |
| 61 | sopan | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 | 0.875 | 0.875 | 0.875 |

Setelah semua fitur diketahui hasil bobot *chi square*-nya, langkah selanjutnya mengurutkan fitur berdasarkan bobot *chi square* tertinggi ke terendah. Berikut ini merupakan hasil mengurutkan fitur berdasarkan bobot *chi square* tertinggi ke terendah yang dapat dilihat pada tabel 3.24.

Tabel 3.24. Hasil Mengurutkan Bobot Chi Square Tertinggi Ke Terendah

| No | Fitur | Max Bobot Chi Square |
|----|------------|----------------------|
| 1 | sayang | 7 |
| 2 | wisata | 3.9375 |
| 3 | ramah | 3.9375 |
| 4 | daun | 3.733333333 |
| 5 | pandu | 3.733333333 |
| 6 | kotor | 3.733333333 |
| 7 | yogyakarta | 2.1 |
| 8 | bersih | 2.1 |
| 9 | barang | 1.555555556 |
| 10 | awat | 1.555555556 |
| 11 | sampah | 1.555555556 |
| 12 | mana | 1.555555556 |
| 13 | usang | 1.555555556 |
| 14 | perintah | 1.555555556 |
| 15 | perhati | 1.555555556 |
| 16 | pelihara | 1.555555556 |
| 17 | Tata | 1.555555556 |
| 18 | ruang | 1.555555556 |
| 19 | pas | 1.555555556 |
| 20 | kesana | 1.555555556 |
| 21 | kondisi | 1.555555556 |
| 22 | kering | 1.555555556 |
| 23 | harga | 1.555555556 |
| 24 | tiket | 1.555555556 |
| 25 | sesuai | 1.555555556 |
| 26 | tulis | 1.555555556 |
| 27 | mahal | 1.555555556 |
| 28 | tuju | 0.875 |
| 29 | lewat | 0.875 |

Tabel 3.25. Lanjutan Hasil Mengurutkan Bobot *Chi Square* Tertinggi Ke Terendah

| No | Fitur | Max Bobot Chi Square |
|----|------------|----------------------|
| 30 | ajar | 0.875 |
| 31 | adat | 0.875 |
| 32 | budaya | 0.875 |
| 33 | jelas | 0.875 |
| 34 | informatif | 0.875 |
| 35 | bagus | 0.875 |
| 36 | layan | 0.875 |
| 37 | mantap | 0.875 |
| 38 | orang | 0.875 |
| 39 | jual | 0.875 |
| 40 | langsung | 0.875 |
| 41 | sambut | 0.875 |
| 42 | keliling | 0.875 |
| 43 | keraton | 0.875 |
| 44 | tambah | 0.875 |
| 45 | ilmu | 0.875 |
| 46 | wawas | 0.875 |
| 47 | destinasi | 0.875 |
| 48 | wajib | 0.875 |
| 49 | suasana | 0.875 |
| 50 | bangun | 0.875 |
| 51 | klasik | 0.875 |
| 52 | akan | 0.875 |
| 53 | zaman | 0.875 |
| 54 | jawa | 0.875 |
| 55 | kuno | 0.875 |
| 56 | indah | 0.875 |
| 57 | banget | 0.875 |
| 58 | pokok | 0.875 |
| 59 | sopan | 0.875 |
| 60 | tempat | 0.058333333 |
| 61 | sejarah | 0.058333333 |

Setelah semua fitur diurutkan berdasarkan bobot *chi square*-nya, langkah selanjutnya menentukan banyaknya fitur yang akan digunakan. Dimisalkan hanya menggunakan 40 fitur dengan bobot *chi square* tertinggi, maka perubahan tabel TF-IDF setelah melalui proses seleksi fitur *chi square* dapat dilihat pada tabel 3.26.

Tabel 3.26. Hasil Perubahan Tabel TF-IDF Setelah Seleksi Fitur *Chi Square*

| No | Fitur | TF-IDF | | | | | | |
|----|------------|----------|----------|----------|----------|----------|----------|----------|
| | | D1 | D2 | D3 | D4 | D5 | D6 | D7 |
| 1 | sayang | 0.367977 | 0.367977 | 0.367977 | 0 | 0 | 0 | 0 |
| 2 | wisata | 0 | 0 | 0 | 0.243038 | 0.243038 | 0 | 0.243038 |
| 3 | ramah | 0 | 0 | 0 | 0.367977 | 0.367977 | 0 | 0.367977 |
| 4 | daun | 0.544068 | 0 | 0.544068 | 0 | 0 | 0 | 0 |
| 5 | pandu | 0.146128 | 0 | 0 | 0.146128 | 0.146128 | 0.146128 | 0.146128 |
| 6 | kotor | 0 | 0.544068 | 0.544068 | 0 | 0 | 0 | 0 |
| 7 | yogyakarta | 0 | 0 | 0 | 0.367977 | 0 | 0 | 0.367977 |
| 8 | bersih | 0 | 0 | 0 | 0 | 0.544068 | 0.544068 | 0 |
| 9 | barang | 0.544068 | 0 | 0 | 0 | 0 | 0 | 0 |
| 10 | awat | 0.845098 | 0 | 0 | 0 | 0 | 0 | 0 |
| 11 | sampah | 0.845098 | 0 | 0 | 0 | 0 | 0 | 0 |
| 12 | mana | 0.845098 | 0 | 0 | 0 | 0 | 0 | 0 |
| 13 | usang | 0.845098 | 0 | 0 | 0 | 0 | 0 | 0 |
| 14 | perintah | 0 | 0.845098 | 0 | 0 | 0 | 0 | 0 |

Tabel 3.27. Lanjutan Hasil Perubahan Tabel TF-IDF Setelah Seleksi Fitur Chi Square

| No | Fitur | TF-IDF | | | | | | |
|----|------------|--------|----------|----------|----------|----------|----------|----|
| | | D1 | D2 | D3 | D4 | D5 | D6 | D7 |
| 15 | perhati | 0 | 0.845098 | 0 | 0 | 0 | 0 | 0 |
| 16 | pelihara | 0 | 0.845098 | 0 | 0 | 0 | 0 | 0 |
| 17 | tata | 0 | 0.845098 | 0 | 0 | 0 | 0 | 0 |
| 18 | ruang | 0 | 0.845098 | 0 | 0 | 0 | 0 | 0 |
| 19 | pas | 0 | 0 | 0.845098 | 0 | 0 | 0 | 0 |
| 20 | kesana | 0 | 0 | 0.845098 | 0 | 0 | 0 | 0 |
| 21 | kondisi | 0 | 0 | 0.845098 | 0 | 0 | 0 | 0 |
| 22 | kering | 0 | 0 | 0.845098 | 0 | 0 | 0 | 0 |
| 23 | harga | 0 | 0 | 0.845098 | 0 | 0 | 0 | 0 |
| 24 | tiket | 0 | 0 | 0.845098 | 0 | 0 | 0 | 0 |
| 25 | sesuai | 0 | 0 | 0.845098 | 0 | 0 | 0 | 0 |
| 26 | tulis | 0 | 0 | 0.845098 | 0 | 0 | 0 | 0 |
| 27 | mahal | 0 | 0 | 0.845098 | 0 | 0 | 0 | 0 |
| 28 | tuju | 0 | 0 | 0 | 0.845098 | 0 | 0 | 0 |
| 29 | lewat | 0 | 0 | 0 | 0.845098 | 0 | 0 | 0 |
| 30 | ajar | 0 | 0 | 0 | 0.845098 | 0 | 0 | 0 |
| 31 | adat | 0 | 0 | 0 | 0.845098 | 0 | 0 | 0 |
| 32 | budaya | 0 | 0 | 0 | 0.845098 | 0 | 0 | 0 |
| 33 | jelas | 0 | 0 | 0 | 0.845098 | 0 | 0 | 0 |
| 34 | informatif | 0 | 0 | 0 | 0.845098 | 0 | 0 | 0 |
| 35 | bagus | 0 | 0 | 0 | 0 | 0.845098 | 0 | 0 |
| 36 | layan | 0 | 0 | 0 | 0 | 0.845098 | 0 | 0 |
| 37 | mantap | 0 | 0 | 0 | 0 | 0.845098 | 0 | 0 |
| 38 | orang | 0 | 0 | 0 | 0 | 0.845098 | 0 | 0 |
| 39 | jual | 0 | 0 | 0 | 0 | 0.845098 | 0 | 0 |
| 40 | langsung | 0 | 0 | 0 | 0 | 0 | 0.845098 | 0 |

3.1.5.2 Mutual information

Mutual Information merupakan seleksi fitur yang mengukur atau menghitung jumlah ketergantungan (informasi) antara dua variabel acak. Pada metode seleksi fitur *mutual information* pertama menempatkan fitur atau *term* yang akan dihitung nilai jumlah kemunculan katanya menggunakan tabel *confusion matrix* untuk kelas positif dan kelas negatif. Kemudian hasil nilai *mutual information* fitur dari kedua kelas tersebut akan dibandingkan, dan akan dipilih nilai *mutual information* terbesar dari fitur tersebut. Kemudian setelah semua fitur telah diketahui nilai bobot *mutual information*-nya, maka semua fitur akan diurutkan berdasarkan nilai bobot *mutual Information* yang paling tinggi ke yang paling rendah. Kemudian, menentukan banyaknya fitur yang akan digunakan untuk proses klasifikasi. Berikut ini merupakan contoh menghitung nilai bobot *mutual information* (MI) dari fitur “tempat”, misal terdapat nilai TF-IDF dari fitur “tempat”:

Tabel 3.28. Hasil Perhitungan Term Weighting TF-IDF Untuk Fitur Tempat.

| Dokumen | TF-IDF (tempat) | Kelas Sentimen |
|---------|-----------------|----------------|
| D1 | 0.544068 | negatif |
| D2 | 0 | negatif |
| D3 | 0 | negatif |
| D4 | 0 | positif |
| D5 | 0 | positif |
| D6 | 0.544068 | positif |
| D7 | 0 | positif |

Pertama, menghitung nilai kemunculan fitur “tempat” untuk kelas positif dan negatif, dengan menempatkan nilai jumlah kemunculan *term* atau kata menggunakan tabel *confusion matrix* seperti berikut ini :

Tabel 3.29. Confusion Matrix Nilai Jumlah Kemunculan Term Tempat Untuk Kelas Positif.

| c = Positif t = “tempat” | ec = 1 | ec = 0 |
|-----------------------------|--------|--------|
| et = 1 | 1 | 1 |
| et = 0 | 3 | 2 |

Tabel 3.30. Confusion Matrix Nilai Jumlah Kemunculan Term Tempat Untuk Kelas Negatif.

| c = Negatif t = “tempat” | ec = 1 | ec = 0 |
|-----------------------------|--------|--------|
| et = 1 | 1 | 1 |
| et = 0 | 2 | 3 |

Kedua, menghitung nilai bobot MI fitur “tempat” dengan menggunakan persamaan 2.7 pada kelas positif dan negatif sebagai berikut :

- Fitur “tempat” kelas positif

$$\begin{aligned}
 I(\text{tempat}, \text{positif}) &= \frac{1}{7} \log_2 \frac{7 \times 1}{(1+1) \times (3+1)} + \frac{3}{7} \log_2 \frac{7 \times 3}{(3+2) \times (3+1)} + \frac{1}{7} \log_2 \frac{7 \times 1}{(1+1) \times (1+2)} + \frac{2}{7} \log_2 \frac{7 \times 2}{(3+2) \times (1+2)} \\
 &= \frac{1}{7} \log_2 \frac{7}{2 \times 4} + \frac{3}{7} \log_2 \frac{21}{5 \times 4} + \frac{1}{7} \log_2 \frac{7}{2 \times 3} + \frac{2}{7} \log_2 \frac{14}{5 \times 3} \\
 &= \frac{1}{7} \log_2 \frac{7}{8} + \frac{3}{7} \log_2 \frac{21}{20} + \frac{1}{7} \log_2 \frac{7}{6} + \frac{2}{7} \log_2 \frac{14}{15} \\
 &= 0.005978
 \end{aligned}$$

- Fitur “tempat” kelas negatif

$$\begin{aligned}
 I(\text{tempat}, \text{negatif}) &= \frac{1}{7} \log_2 \frac{7 \times 1}{(1+1) \times (2+1)} + \frac{2}{7} \log_2 \frac{7 \times 2}{(2+3) \times (2+1)} + \frac{1}{7} \log_2 \frac{7 \times 1}{(1+1) \times (1+3)} + \frac{3}{7} \log_2 \frac{7 \times 3}{(2+3) \times (1+3)} \\
 &= \frac{1}{7} \log_2 \frac{7}{2 \times 3} + \frac{2}{7} \log_2 \frac{14}{5 \times 3} + \frac{1}{7} \log_2 \frac{7}{2 \times 4} + \frac{3}{7} \log_2 \frac{21}{5 \times 4} \\
 &= \frac{1}{7} \log_2 \frac{7}{6} + \frac{2}{7} \log_2 \frac{14}{15} + \frac{1}{7} \log_2 \frac{7}{8} + \frac{3}{7} \log_2 \frac{21}{20} \\
 &= 0.005978
 \end{aligned}$$

Dari perhitungan di atas, diperoleh nilai bobot *mutual information* fitur “tempat” pada kelas positif 0.005978 dan pada kelas negatif 0.005978. Maka hasil nilai bobot *mutual information* terbesar dari fitur “tempat” kelas positif dan negatif adalah 0.005978. Berikut ini merupakan hasil perhitungan nilai bobot *mutual information* untuk semua fitur pada kelas positif dan negatif yang dapat dilihat pada tabel 3.31.

Tabel 3.31. Hasil Nilai Jumlah Kemunculan Fitur Kelas Positif dan Negatif

| No | Fitur | Kelas = positif | | | | Kelas = negatif | | | |
|----|---------|-----------------|-----|-----|-----|-----------------|-----|-----|-----|
| | | N11 | N10 | N01 | N00 | N11 | N10 | N01 | N00 |
| 1 | tempat | 1 | 1 | 3 | 2 | 1 | 1 | 2 | 3 |
| 2 | barang | 0 | 1 | 4 | 2 | 1 | 0 | 2 | 4 |
| 3 | sejarah | 1 | 1 | 3 | 2 | 1 | 1 | 2 | 3 |
| 4 | sayang | 0 | 3 | 4 | 0 | 3 | 0 | 0 | 4 |
| 5 | awat | 0 | 1 | 4 | 2 | 1 | 0 | 2 | 4 |
| 6 | sampah | 0 | 1 | 4 | 2 | 1 | 0 | 2 | 4 |
| 7 | daun | 0 | 2 | 4 | 1 | 2 | 0 | 1 | 4 |
| 8 | mana | 0 | 1 | 4 | 2 | 1 | 0 | 2 | 4 |

Tabel 3.32. Lanjutan Hasil Nilai Jumlah Kemunculan Fitur Kelas Positif dan Negatif

| No | Fitur | Kelas = positif | | | | Kelas = negatif | | | |
|----|------------|-----------------|-----|-----|-----|-----------------|-----|-----|-----|
| | | N11 | N10 | N01 | N00 | N11 | N10 | N01 | N00 |
| 9 | pandu | 4 | 1 | 0 | 2 | 1 | 4 | 2 | 0 |
| 10 | usang | 0 | 1 | 4 | 2 | 1 | 0 | 2 | 4 |
| 11 | kotor | 0 | 2 | 4 | 1 | 2 | 0 | 1 | 4 |
| 12 | perintah | 0 | 1 | 4 | 2 | 1 | 0 | 2 | 4 |
| 13 | perhati | 0 | 1 | 4 | 2 | 1 | 0 | 2 | 4 |
| 14 | pelihara | 0 | 1 | 4 | 2 | 1 | 0 | 2 | 4 |
| 15 | tata | 0 | 1 | 4 | 2 | 1 | 0 | 2 | 4 |
| 16 | ruang | 0 | 1 | 4 | 2 | 1 | 0 | 2 | 4 |
| 17 | pas | 0 | 1 | 4 | 2 | 1 | 0 | 2 | 4 |
| 18 | kesana | 0 | 1 | 4 | 2 | 1 | 0 | 2 | 4 |
| 19 | kondisi | 0 | 1 | 4 | 2 | 1 | 0 | 2 | 4 |
| 20 | kering | 0 | 1 | 4 | 2 | 1 | 0 | 2 | 4 |
| 21 | harga | 0 | 1 | 4 | 2 | 1 | 0 | 2 | 4 |
| 22 | tiket | 0 | 1 | 4 | 2 | 1 | 0 | 2 | 4 |
| 23 | sesuai | 0 | 1 | 4 | 2 | 1 | 0 | 2 | 4 |
| 24 | tulis | 0 | 1 | 4 | 2 | 1 | 0 | 2 | 4 |
| 25 | mahal | 0 | 1 | 4 | 2 | 1 | 0 | 2 | 4 |
| 26 | tuju | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 |
| 27 | wisata | 3 | 0 | 1 | 3 | 0 | 3 | 3 | 1 |
| 28 | lewat | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 |
| 29 | yogyakarta | 2 | 0 | 2 | 3 | 0 | 2 | 3 | 2 |
| 30 | ajar | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 |
| 31 | adat | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 |
| 32 | budaya | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 |
| 33 | ramah | 3 | 0 | 1 | 3 | 0 | 3 | 3 | 1 |
| 34 | jelas | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 |
| 35 | informatif | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 |
| 36 | bersih | 3 | 0 | 1 | 3 | 0 | 3 | 3 | 1 |
| 37 | bagus | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 |
| 38 | layan | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 |
| 39 | mantap | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 |
| 40 | orang | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 |
| 41 | jual | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 |
| 42 | langsung | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 |
| 43 | sambut | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 |
| 44 | keliling | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 |
| 45 | keraton | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 |
| 46 | tambah | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 |
| 47 | ilmu | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 |
| 48 | wawas | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 |
| 49 | destinasi | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 |
| 50 | wajib | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 |
| 51 | suasana | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 |
| 52 | bangun | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 |
| 53 | klasik | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 |
| 54 | akan | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 |
| 55 | zaman | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 |
| 56 | jawa | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 |
| 57 | kuno | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 |
| 58 | indah | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 |
| 59 | banget | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 |
| 60 | pokok | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 |
| 61 | sopan | 1 | 0 | 3 | 3 | 0 | 1 | 3 | 3 |

Berikut ini merupakan hasil perhitungan nilai bobot *mutual information* untuk semua fitur pada kelas positif, negatif dan nilai maksimal bobot (nilai akhir bobot *mutual information*) yang dapat dilihat pada tabel 3.33.

Tabel 3.33. Hasil Perhitungan Nilai Bobot *Mutual Information*

| No | Fitur | Bobot MI | | Max Bobot MI |
|----|------------|----------|----------|--------------|
| | | Positif | Negatif | |
| 1 | tempat | 0.005978 | 0.005978 | 0.005978 |
| 2 | barang | 0.198117 | 0.198117 | 0.198117 |
| 3 | sejarah | 0.005978 | 0.005978 | 0.005978 |
| 4 | sayang | 0.985228 | 0.985228 | 0.985228 |
| 5 | awat | 0.198117 | 0.198117 | 0.198117 |
| 6 | sampah | 0.198117 | 0.198117 | 0.198117 |
| 7 | daun | 0.469565 | 0.469565 | 0.469565 |
| 8 | mana | 0.198117 | 0.198117 | 0.198117 |
| 9 | pandu | 0.469565 | 0.469565 | 0.469565 |
| 10 | usang | 0.198117 | 0.198117 | 0.198117 |
| 11 | kotor | 0.469565 | 0.469565 | 0.469565 |
| 12 | perintah | 0.198117 | 0.198117 | 0.198117 |
| 13 | perhati | 0.198117 | 0.198117 | 0.198117 |
| 14 | pelihara | 0.198117 | 0.198117 | 0.198117 |
| 15 | tata | 0.198117 | 0.198117 | 0.198117 |
| 16 | ruang | 0.198117 | 0.198117 | 0.198117 |
| 17 | pas | 0.198117 | 0.198117 | 0.198117 |
| 18 | kesana | 0.198117 | 0.198117 | 0.198117 |
| 19 | kondisi | 0.198117 | 0.198117 | 0.198117 |
| 20 | kering | 0.198117 | 0.198117 | 0.198117 |
| 21 | harga | 0.198117 | 0.198117 | 0.198117 |
| 22 | tiket | 0.198117 | 0.198117 | 0.198117 |
| 23 | sesuai | 0.198117 | 0.198117 | 0.198117 |
| 24 | tulis | 0.198117 | 0.198117 | 0.198117 |
| 25 | mahal | 0.198117 | 0.198117 | 0.198117 |
| 26 | tuju | 0.128085 | 0.128085 | 0.128085 |
| 27 | wisata | 0.521641 | 0.521641 | 0.521641 |
| 28 | lewat | 0.128085 | 0.128085 | 0.128085 |
| 29 | yogyakarta | 0.291692 | 0.291692 | 0.291692 |
| 30 | ajar | 0.128085 | 0.128085 | 0.128085 |
| 31 | adat | 0.128085 | 0.128085 | 0.128085 |
| 32 | budaya | 0.128085 | 0.128085 | 0.128085 |
| 33 | ramah | 0.521641 | 0.521641 | 0.521641 |
| 34 | jelas | 0.128085 | 0.128085 | 0.128085 |
| 35 | informatif | 0.128085 | 0.128085 | 0.128085 |
| 36 | bersih | 0.521641 | 0.521641 | 0.521641 |
| 37 | bagus | 0.128085 | 0.128085 | 0.128085 |
| 38 | layan | 0.128085 | 0.128085 | 0.128085 |
| 39 | mantap | 0.128085 | 0.128085 | 0.128085 |
| 40 | orang | 0.128085 | 0.128085 | 0.128085 |
| 41 | jual | 0.128085 | 0.128085 | 0.128085 |
| 42 | langsung | 0.128085 | 0.128085 | 0.128085 |
| 43 | sambut | 0.128085 | 0.128085 | 0.128085 |
| 44 | keliling | 0.128085 | 0.128085 | 0.128085 |
| 45 | keraton | 0.128085 | 0.128085 | 0.128085 |
| 46 | tambah | 0.128085 | 0.128085 | 0.128085 |
| 47 | ilmu | 0.128085 | 0.128085 | 0.128085 |
| 48 | wawas | 0.128085 | 0.128085 | 0.128085 |
| 49 | destinasi | 0.128085 | 0.128085 | 0.128085 |

Tabel 3.34. Lanjutan Hasil Perhitungan Nilai Bobot Mutual Information

| No | Fitur | Bobot MI | | Max Bobot MI |
|----|---------|----------|----------|--------------|
| | | Positif | Negatif | |
| 50 | wajib | 0.128085 | 0.128085 | 0.128085 |
| 51 | suasana | 0.128085 | 0.128085 | 0.128085 |
| 52 | bangun | 0.128085 | 0.128085 | 0.128085 |
| 53 | klasik | 0.128085 | 0.128085 | 0.128085 |
| 54 | akan | 0.128085 | 0.128085 | 0.128085 |
| 55 | zaman | 0.128085 | 0.128085 | 0.128085 |
| 56 | jawa | 0.128085 | 0.128085 | 0.128085 |
| 57 | kuno | 0.128085 | 0.128085 | 0.128085 |
| 58 | indah | 0.128085 | 0.128085 | 0.128085 |
| 59 | banget | 0.128085 | 0.128085 | 0.128085 |
| 60 | pokok | 0.128085 | 0.128085 | 0.128085 |
| 61 | sopan | 0.128085 | 0.128085 | 0.128085 |

Setelah semua fitur diketahui hasil bobot *mutual information*-nya, langkah selanjutnya mengurutkan fitur berdasarkan bobot *mutual information* tertinggi ke terendah. Berikut ini merupakan hasil mengurutkan fitur berdasarkan bobot *mutual information* tertinggi ke terendah yang dapat dilihat pada tabel 3.35.

Tabel 3.35. Hasil Mengurutkan Bobot Mutual Information Tertinggi Ke Terendah

| No | Fitur | Bobot MI |
|----|------------|----------|
| 1 | sayang | 0.985228 |
| 2 | wisata | 0.521641 |
| 3 | ramah | 0.521641 |
| 4 | bersih | 0.521641 |
| 5 | daun | 0.469565 |
| 6 | pandu | 0.469565 |
| 7 | kotor | 0.469565 |
| 8 | yogyakarta | 0.291692 |
| 9 | barang | 0.198117 |
| 10 | awat | 0.198117 |
| 11 | sampah | 0.198117 |
| 12 | mana | 0.198117 |
| 13 | usang | 0.198117 |
| 14 | perintah | 0.198117 |
| 15 | perhati | 0.198117 |
| 16 | pelihara | 0.198117 |
| 17 | tata | 0.198117 |
| 18 | ruang | 0.198117 |
| 19 | pas | 0.198117 |
| 20 | kesana | 0.198117 |
| 21 | kondisi | 0.198117 |
| 22 | kering | 0.198117 |
| 23 | harga | 0.198117 |
| 24 | tiket | 0.198117 |
| 25 | sesuai | 0.198117 |
| 26 | tulis | 0.198117 |
| 27 | mahal | 0.198117 |
| 28 | tuju | 0.128085 |
| 29 | lewat | 0.128085 |
| 30 | ajar | 0.128085 |
| 31 | adat | 0.128085 |
| 32 | budaya | 0.128085 |
| 33 | jelas | 0.128085 |
| 34 | informatif | 0.128085 |

Tabel 3.36. Lanjutan Hasil Mengurutkan Bobot *Mutual Information* Tertinggi Ke Terendah

| No | Fitur | Bobot MI |
|----|-----------|----------|
| 35 | bagus | 0.128085 |
| 36 | layan | 0.128085 |
| 37 | mantap | 0.128085 |
| 38 | orang | 0.128085 |
| 39 | jual | 0.128085 |
| 40 | Langsung | 0.128085 |
| 41 | sambut | 0.128085 |
| 42 | keliling | 0.128085 |
| 43 | keraton | 0.128085 |
| 44 | tambah | 0.128085 |
| 45 | ilmu | 0.128085 |
| 46 | wawas | 0.128085 |
| 47 | destinasi | 0.128085 |
| 48 | wajib | 0.128085 |
| 49 | suasana | 0.128085 |
| 50 | bangun | 0.128085 |
| 51 | klasik | 0.128085 |
| 52 | akan | 0.128085 |
| 53 | zaman | 0.128085 |
| 54 | jawa | 0.128085 |
| 55 | kuno | 0.128085 |
| 56 | indah | 0.128085 |
| 57 | banget | 0.128085 |
| 58 | pokok | 0.128085 |
| 59 | sopan | 0.128085 |
| 60 | tempat | 0.005978 |
| 61 | sejarah | 0.005978 |

Setelah semua fitur diurutkan berdasarkan bobot *mutual information*-nya, langkah selanjutnya menentukan banyaknya fitur yang akan digunakan. Dimisalkan hanya menggunakan 40 fitur dengan bobot *Mutual Information* tertinggi, maka perubahan tabel TF-IDF setelah melalui proses seleksi fitur *mutual information* dapat dilihat pada tabel 3.37.

Tabel 3.37. Hasil Perubahan Tabel TF-IDF Setelah Seleksi Fitur *Mutual Information*

| No | Fitur | TF-IDF | | | | | | |
|----|------------|----------|----------|----------|----------|----------|----------|----------|
| | | D1 | D2 | D3 | D4 | D5 | D6 | D7 |
| 1 | sayang | 0.367977 | 0.367977 | 0.367977 | 0 | 0 | 0 | 0 |
| 2 | wisata | 0 | 0 | 0 | 0.243038 | 0.243038 | 0 | 0.243038 |
| 3 | ramah | 0 | 0 | 0 | 0.367977 | 0.367977 | 0 | 0.367977 |
| 4 | bersih | 0 | 0 | 0 | 0 | 0.544068 | 0.544068 | 0 |
| 5 | daun | 0.544068 | 0 | 0.544068 | 0 | 0 | 0 | 0 |
| 6 | pandu | 0.146128 | 0 | 0 | 0.146128 | 0.146128 | 0.146128 | 0.146128 |
| 7 | kotor | 0 | 0.544068 | 0.544068 | 0 | 0 | 0 | 0 |
| 8 | yogyakarta | 0 | 0 | 0 | 0.367977 | 0 | 0 | 0.367977 |
| 9 | barang | 0.544068 | 0 | 0 | 0 | 0 | 0 | 0 |
| 10 | awat | 0.845098 | 0 | 0 | 0 | 0 | 0 | 0 |
| 11 | sampah | 0.845098 | 0 | 0 | 0 | 0 | 0 | 0 |
| 12 | mana | 0.845098 | 0 | 0 | 0 | 0 | 0 | 0 |
| 13 | usang | 0.845098 | 0 | 0 | 0 | 0 | 0 | 0 |
| 14 | perintah | 0 | 0.845098 | 0 | 0 | 0 | 0 | 0 |
| 15 | perhati | 0 | 0.845098 | 0 | 0 | 0 | 0 | 0 |
| 16 | pelihara | 0 | 0.845098 | 0 | 0 | 0 | 0 | 0 |
| 17 | tata | 0 | 0.845098 | 0 | 0 | 0 | 0 | 0 |
| 18 | ruang | 0 | 0.845098 | 0 | 0 | 0 | 0 | 0 |
| 19 | pas | 0 | 0 | 0.845098 | 0 | 0 | 0 | 0 |

Tabel 3.38. Lanjutan Hasil Perubahan Tabel TF-IDF Setelah Seleksi Fitur Mutual Information

| No | Fitur | TF-IDF | | | | | | |
|----|------------|--------|----|----------|----------|----------|----------|----|
| | | D1 | D2 | D3 | D4 | D5 | D6 | D7 |
| 20 | kesana | 0 | 0 | 0.845098 | 0 | 0 | 0 | 0 |
| 21 | kondisi | 0 | 0 | 0.845098 | 0 | 0 | 0 | 0 |
| 22 | kering | 0 | 0 | 0.845098 | 0 | 0 | 0 | 0 |
| 23 | harga | 0 | 0 | 0.845098 | 0 | 0 | 0 | 0 |
| 24 | tiket | 0 | 0 | 0.845098 | 0 | 0 | 0 | 0 |
| 25 | sesuai | 0 | 0 | 0.845098 | 0 | 0 | 0 | 0 |
| 26 | tulis | 0 | 0 | 0.845098 | 0 | 0 | 0 | 0 |
| 27 | mahal | 0 | 0 | 0.845098 | 0 | 0 | 0 | 0 |
| 28 | tuju | 0 | 0 | 0 | 0.845098 | 0 | 0 | 0 |
| 29 | lewat | 0 | 0 | 0 | 0.845098 | 0 | 0 | 0 |
| 30 | ajar | 0 | 0 | 0 | 0.845098 | 0 | 0 | 0 |
| 31 | adat | 0 | 0 | 0 | 0.845098 | 0 | 0 | 0 |
| 32 | budaya | 0 | 0 | 0 | 0.845098 | 0 | 0 | 0 |
| 33 | jelas | 0 | 0 | 0 | 0.845098 | 0 | 0 | 0 |
| 34 | informatif | 0 | 0 | 0 | 0.845098 | 0 | 0 | 0 |
| 35 | bagus | 0 | 0 | 0 | 0 | 0.845098 | 0 | 0 |
| 36 | layan | 0 | 0 | 0 | 0 | 0.845098 | 0 | 0 |
| 37 | mantap | 0 | 0 | 0 | 0 | 0.845098 | 0 | 0 |
| 38 | orang | 0 | 0 | 0 | 0 | 0.845098 | 0 | 0 |
| 39 | jual | 0 | 0 | 0 | 0 | 0.845098 | 0 | 0 |
| 40 | langsung | 0 | 0 | 0 | 0 | 0 | 0.845098 | 0 |

3.1.6 *Support vector machine*

Pada tahap klasifikasi dengan metode *support vector machine* akan dibagi menjadi dua bagian yaitu klasifikasi metode *support vector machine* dengan seleksi fitur *chi square* dan klasifikasi metode *support vector machine* dengan seleksi fitur *mutual information*. Pada setiap bagian akan dibagi lagi menjadi dua tahapan, yaitu tahap pelatihan metode *support vector machine* dan tahap pengujian metode *support vector machine*.

3.1.6.1 *Support vector machine* dengan seleksi fitur *chi square*

Pada proses klasifikasi menggunakan metode *support vector machine* dengan seleksi fitur *chi square* akan dibagi menjadi 2 tahapan yaitu tahap pelatihan dan tahap pengujian.

- a) Tahap pelatihan *support vector machine* dengan seleksi fitur *chi square*

Tahap pelatihan *support vector machine* dengan seleksi fitur *chi square* merupakan tahapan pembentukan model klasifikasi yang akan dibangun. Tahapan pelatihan *support vector machine* dengan seleksi fitur *chi square* antara lain :

1. Menentukan bidang pembatas dengan menggunakan persamaan 2.9 dan 2.10 sehingga didapatkan hasil seperti berikut :

$$D1 : -0.367977w_1 - 0.544068w_4 - 0.146128w_5 - 0.544068w_9 - 0.845098w_{10} - 0.845098w_{11} - 0.845098w_{12} - 0.845098w_{13} - b \geq 1$$

$$D2 : -0.367977w_1 - 0.544068w_6 - 0.845098w_{14} - 0.845098w_{15} - 0.845098w_{16} - 0.845098w_{17} - 0.845098w_{18} - b \geq 1$$

$$D3 : -0.367977w_1 - 0.544068w_4 - 0.544068w_6 - 0.845098w_{19} - 0.845098w_{20} - 0.845098w_{21} - 0.845098w_{22} - 0.845098w_{23} - 0.845098w_{24} - 0.845098w_{25} - 0.845098w_{26} - 0.845098w_{27} - b \geq 1$$

$$D4 : 0.243038w2 + 0.367977w3 + 0.146128w5 + 0.367977w7 + 0.845098w28 + 0.845098w29 + 0.845098w30 + 0.845098w31 + 0.845098w32 + 0.845098w33 + 0.845098w34 + b \geq 1$$

$$D5 : 0.243038w2 + 0.367977w3 + 0.146128w5 + 0.544068w8 + 0.845098w35 + 0.845098w36 + 0.845098w37 + 0.845098w38 + 0.845098w39 + b \geq 1$$

$$D6 : 0.146128w5 + 0.544068w8 + 0.845098w40 + b \geq 1$$

$$D7 : 0.243038w2 + 0.367977w3 + 0.146128w5 + 0.367977w7 + b \geq 1$$

2. Ubah fitur data latih ke dalam format *support vector*

Ubah semua fitur yang ada di data latih pada tabel 3.26 TF-IDF setelah seleksi fitur *chi square* menjadi format *support vector*. Kemudian data berlabel positif diberi kelas 1 sedangkan data berlabel negatif diberi kelas -1 seperti pada tabel 3.39.

Tabel 3.39. Format Support Vector

| No | Fitur | TF-IDF | | | | | | |
|-----------|--------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|
| | | D1(x1) | D2(x2) | D3(x3) | D4(x4) | D5(x5) | D6(x6) | D7(x7) |
| 1 | sayang | 0.367977 | 0.367977 | 0.367977 | 0 | 0 | 0 | 0 |
| 2 | wisata | 0 | 0 | 0 | 0.243038 | 0.243038 | 0 | 0.243038 |
| 3 | ramah | 0 | 0 | 0 | 0.367977 | 0.367977 | 0 | 0.367977 |
| 4 | daun | 0.544068 | 0 | 0.544068 | 0 | 0 | 0 | 0 |
| 5 | pandu | 0.146128 | 0 | 0 | 0.146128 | 0.146128 | 0.146128 | 0.146128 |
| 6 | kotor | 0 | 0.544068 | 0.544068 | 0 | 0 | 0 | 0 |
| 7 | yogyakarta | 0 | 0 | 0 | 0.367977 | 0 | 0 | 0.367977 |
| 8 | bersih | 0 | 0 | 0 | 0 | 0.544068 | 0.544068 | 0 |
| 9 | barang | 0.544068 | 0 | 0 | 0 | 0 | 0 | 0 |
| 10 | awat | 0.845098 | 0 | 0 | 0 | 0 | 0 | 0 |
| 11 | sampah | 0.845098 | 0 | 0 | 0 | 0 | 0 | 0 |
| 12 | mana | 0.845098 | 0 | 0 | 0 | 0 | 0 | 0 |
| 13 | usang | 0.845098 | 0 | 0 | 0 | 0 | 0 | 0 |
| 14 | perintah | 0 | 0.845098 | 0 | 0 | 0 | 0 | 0 |
| 15 | perhati | 0 | 0.845098 | 0 | 0 | 0 | 0 | 0 |
| 16 | pelihara | 0 | 0.845098 | 0 | 0 | 0 | 0 | 0 |
| 17 | tata | 0 | 0.845098 | 0 | 0 | 0 | 0 | 0 |
| 18 | ruang | 0 | 0.845098 | 0 | 0 | 0 | 0 | 0 |
| 19 | pas | 0 | 0 | 0.845098 | 0 | 0 | 0 | 0 |
| 20 | kesana | 0 | 0 | 0.845098 | 0 | 0 | 0 | 0 |
| 21 | kondisi | 0 | 0 | 0.845098 | 0 | 0 | 0 | 0 |
| 22 | kering | 0 | 0 | 0.845098 | 0 | 0 | 0 | 0 |
| 23 | harga | 0 | 0 | 0.845098 | 0 | 0 | 0 | 0 |
| 24 | tiket | 0 | 0 | 0.845098 | 0 | 0 | 0 | 0 |
| 25 | sesuai | 0 | 0 | 0.845098 | 0 | 0 | 0 | 0 |
| 26 | tulis | 0 | 0 | 0.845098 | 0 | 0 | 0 | 0 |
| 27 | mahal | 0 | 0 | 0.845098 | 0 | 0 | 0 | 0 |
| 28 | tuju | 0 | 0 | 0 | 0.845098 | 0 | 0 | 0 |
| 29 | lewat | 0 | 0 | 0 | 0.845098 | 0 | 0 | 0 |
| 30 | ajar | 0 | 0 | 0 | 0.845098 | 0 | 0 | 0 |
| 31 | adat | 0 | 0 | 0 | 0.845098 | 0 | 0 | 0 |
| 32 | budaya | 0 | 0 | 0 | 0.845098 | 0 | 0 | 0 |
| 33 | jelas | 0 | 0 | 0 | 0.845098 | 0 | 0 | 0 |
| 34 | informatif | 0 | 0 | 0 | 0.845098 | 0 | 0 | 0 |
| 35 | bagus | 0 | 0 | 0 | 0 | 0.845098 | 0 | 0 |
| 36 | layan | 0 | 0 | 0 | 0 | 0.845098 | 0 | 0 |
| 37 | mantap | 0 | 0 | 0 | 0 | 0.845098 | 0 | 0 |
| 38 | orang | 0 | 0 | 0 | 0 | 0.845098 | 0 | 0 |
| 39 | jual | 0 | 0 | 0 | 0 | 0.845098 | 0 | 0 |

Tabel 3.40. Lanjutan Format *Support Vector*

| No | Fitur | TF-IDF | | | | | | |
|-----------|--------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|
| | | D1(x1) | D2(x2) | D3(x3) | D4(x4) | D5(x5) | D6(x6) | D7(x7) |
| 40 | langsung | 0 | 0 | 0 | 0 | 0 | 0.845098 | 0 |
| | y (kelas) | -1 | -1 | -1 | 1 | 1 | 1 | 1 |

3. Hitung kernelisasi data latih

Data latih yang sudah diubah menjadi format *support vector* kemudian dilakukan perhitungan kernelisasi menggunakan kernel linear pada persamaan 2.15. Perhitungan matriks kernel linear ini dihitung menggunakan dimensi 1x1 dimana 1 adalah banyaknya data latih, seperti berikut ini :

$$x_1 = [0.367977, 0, 0, 0.544068, 0.146128, 0, 0, 0, 0.544068, 0.845098, 0.845098, 0.845098, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]$$

$$x_2 = [0.367977, 0, 0, 0, 0.544068, 0, 0, 0, 0, 0, 0, 0.845098, 0.845098, 0.845098, 0.845098, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]$$

$$x_3 = [0.367977, 0, 0, 0.544068, 0, 0.544068, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0.845098, 0.845098, 0.845098, 0.845098, 0.845098, 0.845098, 0.845098, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]$$

$$x_4 = [0, 0.243038, 0.367977, 0, 0.146128, 0, 0.367977, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]$$

$$x_5 = [0, 0.243038, 0.367977, 0, 0.146128, 0, 0, 0.544068, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]$$

$$x_6 = [0, 0, 0, 0, 0.146128, 0, 0, 0.544068, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]$$

$$x_7 = [0, 0.243038, 0.367977, 0, 0.146128, 0, 0.367977, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]$$

Dimisalkan menghitung hasil $x_i x_j^T$ kernel linear pada $x_i = 1$ dan $x_j^T = 1$ sampai 7, seperti berikut ini :

$$\begin{aligned} x_1 x_1^T &= [0.367977, 0, 0, 0.544068, 0.146128, 0, 0, 0, 0.544068, 0.845098, 0.845098, 0.845098, 0.845098, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0] \\ &\quad [0.367977, 0, 0, 0.544068, 0.146128, 0, 0, 0, 0.544068, 0.845098, 0.845098, 0.845098, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]^T \end{aligned}$$

$$\begin{aligned} &= 0.1354071 + 0.29601 + 0.021353 + 0.29601 + 0.714191 + 0.714191 + 0.714191 \\ &= 3.605543 \end{aligned}$$

$$x_1 x_2^T = [0.367977, 0, 0, 0.544068, 0.146128, 0, 0, 0, 0.544068, 0.845098, 0.845098, 0.845098, 0.845098, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]$$

$$\begin{aligned}
& [0.367977, 0, 0, 0, 0, 0.544068, 0, 0, 0, 0, 0, 0, 0, 0, 0.845098, 0.845098, \\
& 0.845098, 0.845098, 0.845098, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, \\
& 0, 0, 0]^T \\
& = 0.1354071 \\
x_1 x_3^T & = [0.367977, 0, 0, 0.544068, 0.146128, 0, 0, 0, 0.544068, 0.845098, 0.845098, \\
& 0.845098, 0.845098, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, \\
& 0, 0, 0] \\
& [0.367977, 0, 0, 0.544068, 0, 0.544068, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, \\
& 0.845098, 0.845098, 0.845098, 0.845098, 0.845098, 0.845098, 0.845098, 0.845098, \\
& 0.845098, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, \\
& 0, 0, 0, 0]^T \\
& = 0.1354071 + 0.29601 \\
& = 0.431417 \\
x_1 x_4^T & = [0.367977, 0, 0, 0.544068, 0.146128, 0, 0, 0, 0.544068, 0.845098, 0.845098, \\
& 0.845098, 0.845098, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, \\
& 0, 0, 0, 0] \\
& [0, 0.243038, 0.367977, 0, 0.146128, 0, 0.367977, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, \\
& 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0.845098, 0.845098, 0.845098, 0.845098, 0.845098, \\
& 0.845098, 0.845098, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, \\
& 0, 0, 0, 0]^T \\
& = 0.021353 \\
x_1 x_5^T & = [0.367977, 0, 0, 0.544068, 0.146128, 0, 0, 0, 0.544068, 0.845098, 0.845098, \\
& 0.845098, 0.845098, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, \\
& 0, 0, 0, 0] \\
& [0, 0.243038, 0.367977, 0, 0.146128, 0, 0, 0.544068, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, \\
& 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, \\
& 0.845098, 0.845098, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, \\
& 0, 0, 0, 0]^T \\
& = 0.021353 \\
x_1 x_6^T & = [0.367977, 0, 0, 0.544068, 0.146128, 0, 0, 0, 0.544068, 0.845098, 0.845098, \\
& 0.845098, 0.845098, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, \\
& 0, 0, 0, 0] \\
& [0, 0, 0, 0, 0.146128, 0, 0, 0.544068, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, \\
& 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, \\
& 0, 0, 0, 0, 0.845098]^T \\
& = 0.021353 \\
x_1 x_7^T & = [0.367977, 0, 0, 0.544068, 0.146128, 0, 0, 0, 0.544068, 0.845098, 0.845098, \\
& 0.845098, 0.845098, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, \\
& 0, 0, 0, 0] \\
& [0, 0.243038, 0.367977, 0, 0.146128, 0, 0.367977, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, \\
& 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, \\
& 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, \\
& 0, 0, 0, 0]^T \\
& = 0.021353
\end{aligned}$$

Lakukan perhitungan diatas sampai iterasi $x_i = 1$ sampai 7 dan $x_j^T = 1$ sampai 7 sehingga diperoleh matriks hasil $x_i x_j^T$ seperti berikut ini :

$$x_i x_j^T = \begin{bmatrix} 3.605543 & 0.135407 & 0.431417 & 0.021353 & 0.021353 & 0.021353 & 0.021353 \\ 0.135407 & 4.00237 & 0.431417 & 0 & 0 & 0 & 0 \\ 0.431417 & 0.431417 & 7.155143 & 0 & 0 & 0 & 0 \\ 0.021353 & 0 & 0 & 5.350569 & 0.215828 & 0.021353 & 0.351235 \\ 0.021353 & 0 & 0 & 0.215828 & 4.082791 & 0.317363 & 0.215828 \\ 0.021353 & 0 & 0 & 0.021353 & 0.317363 & 1.031554 & 0.021353 \\ 0.021353 & 0 & 0 & 0.351235 & 0.215828 & 0.021353 & 0.351235 \end{bmatrix}$$

4. Hitung nilai Lagrange Multiplier

Nilai Lagrange Multiplier ini untuk memperoleh nilai α_i yang menghasilkan nilai Lagrange Multiplier yang maksimal dengan menggunakan persamaan 2.11. seperti berikut ini :

$$\begin{aligned} \max_{\alpha} L_D &= [(\alpha_1 + \alpha_2 + \alpha_3 + \alpha_4 + \alpha_5 + \alpha_6 + \alpha_7) - \frac{1}{2}(3.605543\alpha_1\alpha_1 + \\ &\quad 0.135407\alpha_1\alpha_2 + 0.431417\alpha_1\alpha_3 - 0.021353\alpha_1\alpha_4 - 0.021353\alpha_1\alpha_5 - \\ &\quad 0.021353\alpha_1\alpha_6 - 0.021353\alpha_1\alpha_7 + 0.135407\alpha_2\alpha_1 + 4.00237\alpha_2\alpha_2 + \\ &\quad 0.431417\alpha_2\alpha_3 - 0\alpha_2\alpha_4 - 0\alpha_2\alpha_5 - 0\alpha_2\alpha_6 - 0\alpha_2\alpha_7 + \\ &\quad 0.431417\alpha_3\alpha_1 + 0.431417\alpha_3\alpha_2 + 7.155143\alpha_3\alpha_3 - 0\alpha_3\alpha_4 - 0\alpha_3\alpha_5 - \\ &\quad 0\alpha_3\alpha_6 - 0\alpha_3\alpha_7 - 0.021353\alpha_4\alpha_1 - 0\alpha_4\alpha_2 - 0\alpha_4\alpha_3 + \\ &\quad 5.350569\alpha_4\alpha_4 + 0.215828\alpha_4\alpha_5 + 0.021353\alpha_4\alpha_6 + 0.351235\alpha_4\alpha_7 - \\ &\quad 0.021353\alpha_5\alpha_1 - 0\alpha_5\alpha_2 - 0\alpha_5\alpha_3 + 0.215828\alpha_5\alpha_4 + 4.082791\alpha_5\alpha_5 + \\ &\quad 0.317363\alpha_5\alpha_6 + 0.215828\alpha_5\alpha_7 - 0.021353\alpha_6\alpha_1 - 0\alpha_6\alpha_2 - 0\alpha_6\alpha_3 + \\ &\quad 0.021353\alpha_6\alpha_4 + 0.317363\alpha_6\alpha_5 + 1.031554\alpha_6\alpha_6 + 0.021353\alpha_6\alpha_7 - \\ &\quad 0.021353\alpha_7\alpha_1 - 0\alpha_7\alpha_2 - 0\alpha_7\alpha_3 + 0.351235\alpha_7\alpha_4 + 0.215828\alpha_7\alpha_5 + \\ &\quad 0.021353\alpha_7\alpha_6 + 0.351235\alpha_7\alpha_7) \\ &= [(\alpha_1 + \alpha_2 + \alpha_3 + \alpha_4 + \alpha_5 + \alpha_6 + \alpha_7) - \frac{1}{2}(3.605543\alpha_1^2 + \\ &\quad 0.270814\alpha_1\alpha_2 + 0.862834\alpha_1\alpha_3 - 0.04271\alpha_1\alpha_4 - 0.04271\alpha_1\alpha_5 - \\ &\quad 0.04271\alpha_1\alpha_6 - 0.04271\alpha_1\alpha_7 + 4.00237\alpha_2^2 + 0.862834\alpha_2\alpha_3 + \\ &\quad 7.155143\alpha_3^2 + 5.350569\alpha_4^2 + 0.431656\alpha_4\alpha_5 + 0.042706\alpha_4\alpha_6 + \\ &\quad 0.70247\alpha_4\alpha_7 + 4.082791\alpha_5^2 + 0.634726\alpha_5\alpha_6 + 0.431656\alpha_5\alpha_7 + \\ &\quad 1.031554\alpha_6^2 + 0.042706\alpha_6\alpha_7 + 0.351235\alpha_7^2)] \end{aligned}$$

Dengan syarat : $\alpha_i \geq 0$ ($i = 1, 2, \dots, n$) dan $\sum_{i=1}^n \alpha_i y_i = 0$ maka didapatkan nilai $\max_{\alpha} L_D = -12.735118$ dengan nilai $\alpha_1 = 2, \alpha_2 = 1, \alpha_3 = 1, \alpha_4 = 1, \alpha_5 = 1, \alpha_6 = 1, \alpha_7 = 1$

5. Menentukan nilai w (bobot) dan b (bias) dengan menggunakan persamaan 2.12 dan 2.13 sebagai berikut :

$$\begin{aligned} w &= 2(-1)x_1 + 1.(-1).x_2 + 1(-1)x_3 + 1(1)x_4 + 1(1)x_5 + 1(1)x_6 + 1(1)x_7 \\ &= -2x_1 - x_2 - x_3 + x_4 + x_5 + x_6 + x_7 \\ &= [-1.47191, 0.729114, 1.103931, -1.6322, 0.292256, -1.08814, 0.735954, \\ &\quad 1.088136, -1.08814, -1.6902, -1.6902, -1.6902, -1.6902, -0.8451, -0.8451, \\ &\quad -0.8451, -0.845098, -0.8451, -0.8451, 0.845098, 0.845098, 0.845098, \\ &\quad 0.845098, 0.845098, 0.845098, 0.845098, 0.845098, 0.845098, 0.845098, \\ &\quad 0.845098] \end{aligned}$$

$$\begin{aligned}
b &= -\frac{1}{2}(x_1^- \cdot w + x_2^- \cdot w + x_3^- \cdot w + x_4^+ \cdot w + x_5^+ \cdot w + x_6^+ \cdot w + x_7^+ \cdot w) \\
&= -\frac{1}{2}(-7.692510743 - 4.704612779 - 8.449408157 + 5.896278963 + \\
&\quad 4.789103536 + 1.348917392 + 0.896944556) \\
&= -\frac{1}{2}(-7.915287232) \\
&= 3.957643616
\end{aligned}$$

b) Tahap pengujian *support vector machine* dengan seleksi fitur *chi square*

Dimisalkan terdapat data uji x = “Wisatanya bersih sekali,, terus pemandunya juga ramah... MANTAB!!!!”. Sebelum data uji tersebut masuk ke proses klasifikasi, terlebih dahulu melalui *text preprocessing* sebagai berikut :

Tabel 3.41. Proses Text Preprocessing Data Uji

| Tahap Text Preprocessing | Hasil Text Preprocessing |
|---------------------------|-------------------------------------------------------------------------------------|
| case folding | wisatanya bersih sekali,, terus pemandunya juga ramah... mantab!!!! |
| remove punctuation | wisatanya bersih sekali terus pemandunya juga ramah mantab |
| remove number | wisatanya bersih sekali terus pemandunya juga ramah mantab |
| remove repeated character | wisatanya bersih sekali terus pemandunya juga ramah mantab |
| spelling normalization | wisatanya bersih sekali terus pemandunya juga ramah mantap |
| tokenizing | ['wisatanya', 'bersih', 'sekali', 'terus', 'pemandunya', 'juga', 'ramah', 'mantap'] |
| filtering | ['wisatanya', 'bersih', 'pemandunya', 'ramah', 'mantap'] |
| stemming | ['wisata', 'bersih', 'pandu', 'ramah', 'mantap'] |

Selanjutnya masuk ke proses tahapan pengujian *support vector machine* dengan seleksi fitur *chi square* antara lain :

1. Hitung TF data uji terhadap data latih

Setelah dilakukan *text preprocessing* langkah selanjutnya menghitung TF data uji terhadap data latih dengan seleksi fitur *mutual information* pada tabel 3.28 sehingga didapatkan hasil yang dapat dilihat pada tabel 3.42, 3.43 dan 3.44.

Tabel 3.42. Hitung TF data uji terhadap data latih

| | t1 | t2 | t3 | t4 | t5 | t6 | t7 | t8 | t9 | t10 | t11 | t12 | t13 | t14 | t15 |
|---|----|----|----|----|----|----|----|----|----|-----|-----|-----|-----|-----|-----|
| x | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Tabel 3.43. Lanjutan Hitung TF data uji terhadap data latih

| | t16 | t17 | t18 | t19 | t20 | t21 | t22 | t23 | t24 | t25 | t26 | t27 |
|---|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| x | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Tabel 3.44. Lanjutan Hitung TF data uji terhadap data latih

| | t28 | t29 | t30 | t31 | t32 | t33 | t34 | t35 | t36 | t37 | t38 | t39 | t40 |
|---|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| x | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |

Pada tabel 3.42, 3.43 dan 3.44 diperoleh bahwa fitur atau *term* data uji x terdapat pada t2, t3, t5, t8 dan t37.

2. Hitung kernelisasi data uji terhadap data latih

Pada tahap ini menghitung kernelisasi data uji terhadap data latih menggunakan kernel linear dengan mengalikan matriks fitur data uji x dengan fitur yang terdapat pada data latih (Dokumen 1 sampai 7), sehingga diperoleh hasil yang dapat dilihat pada tabel 3.45.

Tabel 3.45. Hasil Hitung kernelisasi data uji terhadap data latih

| | x1 | x2 | x3 | x4 | x5 | x6 | x7 |
|----------|-----------|----|----|----------|----------|----------|----------|
| k(x,x_i) | 0.0097828 | 0 | 0 | 0.158283 | 0.818278 | 0.209987 | 0.158283 |

3. Penentuan kelas dengan fungsi keputusan klasifikasi

Penentuan kelas data uji dengan fungsi keputusan menggunakan persamaan 2.14 yaitu fungsi keputusan klasifikasi $sign(f(x))$ dimana nilai α dan b diambil dari proses pelatihan metode SVM dengan seleksi fitur *chi square* sedangkan nilai $k(x, x_i)$ diperoleh dari table 3.46. Fungsi $sign(f(x))$ merupakan fungsi normalisasi, jika nilai x di dalam fungsi $sign(f(x))$ lebih dari 0 atau > 0 maka fungsi tersebut memberikan nilai 1 (kelas positif). Jika nilai x dalam fungsi $sign(f(x))$ kurang dari 0 atau < 0 maka fungsi tersebut memberikan nilai -1 (kelas negatif). Berikut ini merupakan perhitungan penentuan kelas data uji dengan fungsi keputusan klasifikasi $sign(f(x))$:

$$\begin{aligned} f(x) &= sign[(2(-1)(0.0097828) + 1(-1)(0) + 1(-1)(0) + 1(1)(0.158283) + \\ &\quad 1(1)(0.818278) + 1(1)(0.209987) + 1(1)(0.158283)) + 3.9576436] \\ &= sign[(-0.0195656 + 0 + 0 + 0.158282694 + 0.818277904 + \\ &\quad 0.209987194 + 0.158282694) + 3.9576436] \\ &= sign[1.325264886 + 3.9576436] \\ &= sign[5.282908486] \\ &= 1 \end{aligned}$$

Berdasarkan hasil perhitungan fungsi keputusan klasifikasi $sign(f(x))$ maka data uji x = “Wisatanya bersih sekali,, terus pemandunya juga ramah... MANTAB!!!!” masuk ke dalam kelas positif (sentimen positif).

3.1.6.2 Support vector machine dengan seleksi fitur mutual information

a) Tahap pelatihan support vector machine dengan seleksi fitur mutual information

Tahap pelatihan *support vector machine* dengan seleksi fitur *mutual information* merupakan tahapan pembentukan model klasifikasi yang akan dibangun. Tahapan pelatihan *support vector machine* dengan seleksi fitur *mutual information* antara lain :

1. Menentukan bidang pembatas dengan menggunakan persamaan 2.9 dan 2.10 sehingga didapatkan hasil seperti berikut :

$$\begin{aligned} D1 &: -0.367977w_1 - 0.544068w_5 - 0.146128w_6 - 0.544068w_9 - 0.845098w_{10} - \\ &\quad 0.845098w_{11} - 0.845098w_{12} - 0.845098w_{13} - b \geq 1 \\ D2 &: -0.367977w_1 - 0.544068w_7 - 0.845098w_{14} - 0.845098w_{15} - 0.845098w_{16} - \\ &\quad 0.845098w_{17} - 0.845098w_{18} - b \geq 1 \\ D3 &: -0.367977w_1 - 0.544068w_5 - 0.544068w_7 - 0.845098w_{19} - 0.845098w_{20} - \\ &\quad 0.845098w_{21} - 0.845098w_{22} - 0.845098w_{23} - 0.845098w_{24} - 0.845098w_{25} - \\ &\quad 0.845098w_{26} - 0.845098w_{27} - b \geq 1 \\ D4 &: 0.243038w_2 + 0.367977w_3 + 0.146128w_6 + 0.367977w_8 + 0.845098w_{28} + \\ &\quad 0.845098w_{29} + 0.845098w_{30} + 0.845098w_{31} + 0.845098w_{32} + 0.845098w_{33} + \\ &\quad 0.845098w_{34} + b \geq 1 \\ D5 &: 0.243038w_2 + 0.367977w_3 + 0.544068w_4 + 0.146128w_6 + 0.845098w_{35} + \\ &\quad 0.845098w_{36} + 0.845098w_{37} + 0.845098w_{38} + 0.845098w_{39} + b \geq 1 \\ D6 &: 0.544068w_4 + 0.146128w_6 + 0.845098w_{40} + b \geq 1 \\ D7 &: 0.243038w_2 + 0.367977w_3 + 0.146128w_6 + 0.367977w_8 + b \geq 1 \end{aligned}$$

2. Ubah fitur ke dalam format *support vector*

Ubah semua fitur yang ada di data latih pada tabel 3.28 TF-IDF setelah seleksi fitur *mutual information* menjadi format *support vector*. Kemudian data berlabel positif diberi kelas 1 sedangkan data berlabel negatif diberi kelas -1 seperti pada tabel 3.46.

Tabel 3.46. Format Support Vector

| No | Fitur | TF-IDF | | | | | | |
|-----------|------------|----------|----------|----------|----------|----------|----------|----------|
| | | D1(x1) | D2(x2) | D3(x3) | D4(x4) | D5(x5) | D6(x6) | D7(x7) |
| 1 | sayang | 0.367977 | 0.367977 | 0.367977 | 0 | 0 | 0 | 0 |
| 2 | wisata | 0 | 0 | 0 | 0.243038 | 0.243038 | 0 | 0.243038 |
| 3 | ramah | 0 | 0 | 0 | 0.367977 | 0.367977 | 0 | 0.367977 |
| 4 | bersih | 0 | 0 | 0 | 0 | 0.544068 | 0.544068 | 0 |
| 5 | daun | 0.544068 | 0 | 0.544068 | 0 | 0 | 0 | 0 |
| 6 | pandu | 0.146128 | 0 | 0 | 0.146128 | 0.146128 | 0.146128 | 0.146128 |
| 7 | kotor | 0 | 0.544068 | 0.544068 | 0 | 0 | 0 | 0 |
| 8 | yogyakarta | 0 | 0 | 0 | 0.367977 | 0 | 0 | 0.367977 |
| 9 | barang | 0.544068 | 0 | 0 | 0 | 0 | 0 | 0 |
| 10 | awat | 0.845098 | 0 | 0 | 0 | 0 | 0 | 0 |
| 11 | sampah | 0.845098 | 0 | 0 | 0 | 0 | 0 | 0 |
| 12 | mana | 0.845098 | 0 | 0 | 0 | 0 | 0 | 0 |
| 13 | usang | 0.845098 | 0 | 0 | 0 | 0 | 0 | 0 |
| 14 | perintah | 0 | 0.845098 | 0 | 0 | 0 | 0 | 0 |
| 15 | perhati | 0 | 0.845098 | 0 | 0 | 0 | 0 | 0 |
| 16 | pelihara | 0 | 0.845098 | 0 | 0 | 0 | 0 | 0 |
| 17 | tata | 0 | 0.845098 | 0 | 0 | 0 | 0 | 0 |
| 18 | ruang | 0 | 0.845098 | 0 | 0 | 0 | 0 | 0 |
| 19 | pas | 0 | 0 | 0.845098 | 0 | 0 | 0 | 0 |
| 20 | kesana | 0 | 0 | 0.845098 | 0 | 0 | 0 | 0 |
| 21 | kondisi | 0 | 0 | 0.845098 | 0 | 0 | 0 | 0 |
| 22 | kering | 0 | 0 | 0.845098 | 0 | 0 | 0 | 0 |
| 23 | harga | 0 | 0 | 0.845098 | 0 | 0 | 0 | 0 |
| 24 | tiket | 0 | 0 | 0.845098 | 0 | 0 | 0 | 0 |
| 25 | sesuai | 0 | 0 | 0.845098 | 0 | 0 | 0 | 0 |
| 26 | tulis | 0 | 0 | 0.845098 | 0 | 0 | 0 | 0 |
| 27 | mahal | 0 | 0 | 0.845098 | 0 | 0 | 0 | 0 |
| 28 | tuju | 0 | 0 | 0 | 0.845098 | 0 | 0 | 0 |
| 29 | lewat | 0 | 0 | 0 | 0.845098 | 0 | 0 | 0 |
| 30 | ajar | 0 | 0 | 0 | 0.845098 | 0 | 0 | 0 |
| 31 | adat | 0 | 0 | 0 | 0.845098 | 0 | 0 | 0 |
| 32 | budaya | 0 | 0 | 0 | 0.845098 | 0 | 0 | 0 |
| 33 | jelas | 0 | 0 | 0 | 0.845098 | 0 | 0 | 0 |
| 34 | informatif | 0 | 0 | 0 | 0.845098 | 0 | 0 | 0 |
| 35 | bagus | 0 | 0 | 0 | 0 | 0.845098 | 0 | 0 |
| 36 | layan | 0 | 0 | 0 | 0 | 0.845098 | 0 | 0 |
| 37 | mantap | 0 | 0 | 0 | 0 | 0.845098 | 0 | 0 |
| 38 | orang | 0 | 0 | 0 | 0 | 0.845098 | 0 | 0 |
| 39 | jual | 0 | 0 | 0 | 0 | 0.845098 | 0 | 0 |
| 40 | langsung | 0 | 0 | 0 | 0 | 0 | 0.845098 | 0 |
| y (kelas) | | -1 | -1 | -1 | 1 | 1 | 1 | 1 |

3. Hitung kernelisasi

Data latih yang sudah diubah menjadi format *support vector* kemudian dilakukan perhitungan kernelisasi menggunakan kernel linear pada persamaan 2.15. Perhitungan matriks kernel linear ini dihitung menggunakan dimensi 1x1 dimana 1 adalah banyaknya data latih, seperti berikut ini :

$$\begin{aligned}
x_1 &= [0.367977, 0, 0, 0, 0.544068, 0.146128, 0, 0, 0.544068, 0.845098, 0.845098, \\
&\quad 0.845098, 0.845098, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0] \\
x_2 &= [0.367977, 0, 0, 0, 0, 0.544068, 0, 0, 0, 0, 0, 0, 0.845098, 0.845098, 0.845098, \\
&\quad 0.845098, 0.845098, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0] \\
x_3 &= [0.367977, 0, 0, 0, 0.544068, 0, 0.544068, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0] \\
x_4 &= [0, 0.243038, 0.367977, 0, 0, 0.146128, 0, 0.367977, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0] \\
x_5 &= [0, 0.243038, 0.367977, 0.544068, 0, 0.146128, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0] \\
x_6 &= [0, 0, 0, 0.544068, 0, 0.146128, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0] \\
x_7 &= [0, 0.243038, 0.367977, 0, 0, 0.146128, 0, 0.367977, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
\end{aligned}$$

4. Dimisalkan menghitung hasil $x_i x_j^T$ kernel linear pada $x_i = 1$ dan $x_j^T = 1$ sampai 7, seperti berikut ini :

$$\begin{aligned}
x_1 x_1^T &= [0.367977, 0, 0, 0, 0.544068, 0.146128, 0, 0, 0.544068, 0.845098, 0.845098, \\
&\quad 0.845098, 0.845098, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0] \\
&\quad [0.367977, 0, 0, 0, 0.544068, 0.146128, 0, 0, 0.544068, 0.845098, 0.845098, \\
&\quad 0.845098, 0.845098, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0] \\
&= 0.1354071+0.29601+0.021353+0.29601+0.714191+0.714191+0.714191+0.71 \\
&\quad 4191 \\
&= 3.605543 \\
x_1 x_2^T &= [0.367977, 0, 0, 0, 0.544068, 0.146128, 0, 0, 0.544068, 0.845098, 0.845098, \\
&\quad 0.845098, 0.845098, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0] \\
&\quad [0.367977, 0, 0, 0, 0, 0.544068, 0, 0, 0, 0, 0, 0, 0.845098, 0.845098, 0.845098, \\
&\quad 0.845098, 0.845098, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0] \\
&= 0.135407 \\
x_1 x_3^T &= [0.367977, 0, 0, 0, 0.544068, 0.146128, 0, 0, 0.544068, 0.845098, 0.845098, \\
&\quad 0.845098, 0.845098, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0] \\
&\quad [0.367977, 0, 0, 0, 0.544068, 0, 0.544068, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0] \\
&= 0.1354071+0.29601 \\
&= 0.431417
\end{aligned}$$

$$\begin{aligned}
x_1 x_4^T &= [0.367977, 0, 0, 0, 0.544068, 0.146128, 0, 0, 0.544068, 0.845098, 0.845098, \\
&\quad 0.845098, 0.845098, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, \\
&\quad 0, 0, 0] \\
&[0, 0.243038, 0.367977, 0, 0, 0.146128, 0, 0.367977, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, \\
&\quad 0, 0, 0, 0, 0, 0, 0, 0, 0.845098, 0.845098, 0.845098, 0.845098, 0.845098, 0.845098, \\
&\quad 0.845098, 0.845098, 0, 0, 0, 0, 0]^T \\
&= 0.021353 \\
x_1 x_5^T &= [0.367977, 0, 0, 0, 0.544068, 0.146128, 0, 0, 0.544068, 0.845098, 0.845098, \\
&\quad 0.845098, 0.845098, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, \\
&\quad 0, 0, 0] \\
&[0, 0.243038, 0.367977, 0.544068, 0, 0.146128, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, \\
&\quad 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0.845098, 0.845098, 0.845098, 0.845098, \\
&\quad 0.845098, 0.845098, 0]^T \\
&= 0.021353 \\
x_1 x_6^T &= [0.367977, 0, 0, 0, 0.544068, 0.146128, 0, 0, 0.544068, 0.845098, 0.845098, \\
&\quad 0.845098, 0.845098, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, \\
&\quad 0, 0, 0] \\
&[0, 0, 0, 0.544068, 0, 0.146128, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, \\
&\quad 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0.845098]^T \\
&= 0.021353 \\
x_1 x_7^T &= [0.367977, 0, 0, 0, 0.544068, 0.146128, 0, 0, 0.544068, 0.845098, 0.845098, \\
&\quad 0.845098, 0.845098, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, \\
&\quad 0, 0, 0] \\
&[0, 0.243038, 0.367977, 0, 0, 0.146128, 0, 0.367977, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, \\
&\quad 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]^T \\
&= 0.021353
\end{aligned}$$

Lakukan perhitungan diatas sampai iterasi $x_i = 1$ sampai 7 dan $x_j^T = 1$ sampai 7 sehingga diperoleh matriks hasil $x_i x_j^T$ seperti berikut ini :

$$x_i x_j^T = \begin{bmatrix} 3.605543 & 0.135407 & 0.431417 & 0.021353 & 0.021353 & 0.021353 & 0.021353 \\ 0.135407 & 4.00237 & 0.431417 & 0 & 0 & 0 & 0 \\ 0.431417 & 0.431417 & 7.155143 & 0 & 0 & 0 & 0 \\ 0.021353 & 0 & 0 & 5.350569 & 0.215828 & 0.021353 & 0.351235 \\ 0.021353 & 0 & 0 & 0.215828 & 4.082791 & 0.317363 & 0.215828 \\ 0.021353 & 0 & 0 & 0.021353 & 0.317363 & 1.031554 & 0.021353 \\ 0.021353 & 0 & 0 & 0.351235 & 0.215828 & 0.021353 & 0.351235 \end{bmatrix}$$

5. Hitung nilai Lagrange Multiplier

Nilai Lagrange Multiplier ini untuk memperoleh nilai α_i yang menghasilkan nilai Lagrange Multiplier yang maksimal dengan menggunakan persamaan 2.11. seperti berikut ini :

$$\begin{aligned}
\max_{\alpha} L_D &= [(\alpha_1 + \alpha_2 + \alpha_3 + \alpha_4 + \alpha_5 + \alpha_6 + \alpha_7) - \frac{1}{2}(3.605543\alpha_1\alpha_1 + \\
&\quad 0.135407\alpha_1\alpha_2 + 0.431417\alpha_1\alpha_3 - 0.021353\alpha_1\alpha_4 - 0.021353\alpha_1\alpha_5 - \\
&\quad 0.021353\alpha_1\alpha_6 - 0.021353\alpha_1\alpha_7 + 0.135407\alpha_2\alpha_1 + 4.00237\alpha_2\alpha_2 + \\
&\quad 0.431417\alpha_2\alpha_3 - 0\alpha_2\alpha_4 - 0\alpha_2\alpha_5 - 0\alpha_2\alpha_6 - 0\alpha_2\alpha_7 + \\
&\quad 0.021353\alpha_3\alpha_1 + 0.021353\alpha_3\alpha_2 - 0.021353\alpha_3\alpha_4 - 0.021353\alpha_3\alpha_5 - \\
&\quad 0.021353\alpha_3\alpha_6 - 0.021353\alpha_3\alpha_7 + 0.135407\alpha_4\alpha_1 + 0.431417\alpha_4\alpha_2 + \\
&\quad 0.021353\alpha_4\alpha_3 - 0.021353\alpha_4\alpha_5 - 0.021353\alpha_4\alpha_6 - 0.021353\alpha_4\alpha_7 + \\
&\quad 0.021353\alpha_5\alpha_1 + 0.021353\alpha_5\alpha_2 - 0.021353\alpha_5\alpha_3 - 0.021353\alpha_5\alpha_4 - \\
&\quad 0.021353\alpha_5\alpha_6 - 0.021353\alpha_5\alpha_7 + 0.135407\alpha_6\alpha_1 + 0.431417\alpha_6\alpha_2 + \\
&\quad 0.021353\alpha_6\alpha_3 - 0.021353\alpha_6\alpha_4 - 0.021353\alpha_6\alpha_5 - 0.021353\alpha_6\alpha_7 + \\
&\quad 0.021353\alpha_7\alpha_1 + 0.021353\alpha_7\alpha_2 - 0.021353\alpha_7\alpha_3 - 0.021353\alpha_7\alpha_4 - \\
&\quad 0.021353\alpha_7\alpha_5 - 0.021353\alpha_7\alpha_6 + 0.135407\alpha_2\alpha_7 + 4.00237\alpha_3\alpha_7 + \\
&\quad 0.431417\alpha_4\alpha_7 + 0.021353\alpha_5\alpha_7 + 0.021353\alpha_6\alpha_7]
\end{aligned}$$

$$\begin{aligned}
& 0.431417\alpha_3\alpha_1 + 0.431417\alpha_3\alpha_2 + 7.155143\alpha_3\alpha_3 - 0\alpha_3\alpha_4 - 0\alpha_3\alpha_5 - \\
& 0\alpha_3\alpha_6 - 0\alpha_3\alpha_7 - 0.021353\alpha_4\alpha_1 - 0\alpha_4\alpha_2 - 0\alpha_4\alpha_3 + \\
& 5.350569\alpha_4\alpha_4 + 0.215828\alpha_4\alpha_5 + 0.021353\alpha_4\alpha_6 + 0.351235\alpha_4\alpha_7 - \\
& 0.021353\alpha_5\alpha_1 - 0\alpha_5\alpha_2 - 0\alpha_5\alpha_3 + 0.215828\alpha_5\alpha_4 + 4.082791\alpha_5\alpha_5 + \\
& 0.317363\alpha_5\alpha_6 + 0.215828\alpha_5\alpha_7 - 0.021353\alpha_6\alpha_1 - 0\alpha_6\alpha_2 - 0\alpha_6\alpha_3 + \\
& 0.021353\alpha_6\alpha_4 + 0.317363\alpha_6\alpha_5 + 1.031554\alpha_6\alpha_6 + 0.021353\alpha_6\alpha_7 - \\
& 0.021353\alpha_7\alpha_1 - 0\alpha_7\alpha_2 - 0\alpha_7\alpha_3 + 0.351235\alpha_7\alpha_4 + 0.215828\alpha_7\alpha_5 + \\
& 0.021353\alpha_7\alpha_6 + 0.351235\alpha_7\alpha_7) \\
= & [(\alpha_1 + \alpha_2 + \alpha_3 + \alpha_4 + \alpha_5 + \alpha_6 + \alpha_7) - \frac{1}{2}(3.605543\alpha_1^2 + \\
& 0.270814\alpha_1\alpha_2 + 0.862834\alpha_1\alpha_3 - 0.04271\alpha_1\alpha_4 - 0.04271\alpha_1\alpha_5 - \\
& 0.04271\alpha_1\alpha_6 - 0.04271\alpha_1\alpha_7 + 4.00237\alpha_2^2 + 0.862834\alpha_2\alpha_3 + \\
& 7.155143\alpha_3^2 + 5.350569\alpha_4^2 + 0.431656\alpha_4\alpha_5 + 0.042706\alpha_4\alpha_6 + \\
& 0.70247\alpha_4\alpha_7 + 4.082791\alpha_5^2 + 0.634726\alpha_5\alpha_6 + 0.431656\alpha_5\alpha_7 + \\
& 1.031554\alpha_6^2 + 0.042706\alpha_6\alpha_7 + 0.351235\alpha_7^2)
\end{aligned}$$

Dengan syarat : $\alpha_i \geq 0$ ($i = 1, 2, \dots, n$) dan $\sum_{i=1}^n \alpha_i y_i = 0$ maka didapatkan nilai $\max_{\alpha} L_D = -12.735118$ dengan nilai $\alpha_1 = 2, \alpha_2 = 1, \alpha_3 = 1, \alpha_4 = 1, \alpha_5 = 1, \alpha_6 = 1, \alpha_7 = 1$

6. Menentukan nilai w (bobot) dan b (bias) dengan menggunakan persamaan 2.12 dan 2.13 sebagai berikut :

$$\begin{aligned}
w &= 2(-1)x_1 + 1(-1)x_2 + 1(-1)x_3 + 1(1)x_4 + 1(1)x_5 + 1(1)x_6 + 1(1)x_7 \\
&= -2x_1 - x_2 - x_3 + x_4 + x_5 + x_6 + x_7 \\
&= [-1.47191, 0.729114, 1.103931, 1.088136, 1.6322, 0.292256, -1.08814, \\
&\quad 0.735954, -1.08814, -1.6902, -1.6902, -1.6902, -0.8451, -0.8451, \\
&\quad -0.8451, -0.8451, -0.8451, -0.8451, -0.8451, -0.8451, -0.8451, \\
&\quad -0.8451, -0.845098, -0.8451, -0.8451, 0.845098, 0.845098, 0.845098, \\
&\quad 0.845098, 0.845098, 0.845098, 0.845098, 0.845098, 0.845098, 0.845098, \\
&\quad 0.845098, 0.845098, 0.845098] \\
b &= -\frac{1}{2}(x_1^- \cdot w + x_2^- \cdot w + x_3^- \cdot w + x_4^+ \cdot w + x_5^+ \cdot w + x_6^+ \cdot w + x_7^+ \cdot w) \\
&= -\frac{1}{2}(-7.692510743 - 4.704612779 - 8.449408157 + 5.896278963 + \\
&\quad 4.789103536 + 1.348917392 + 0.896944556) \\
&= -\frac{1}{2}(-7.915287232) \\
&= 3.957643616
\end{aligned}$$

- b) Tahap pengujian *support vector machine* dengan seleksi fitur *mutual information*

Dimisalkan terdapat data uji x = “Wisatanya bersih sekali,, terus pemandunya juga ramah... MANTAB!!!!”. Sebelum data uji tersebut masuk ke proses klasifikasi, terlebih dahulu melalui *text preprocessing* sebagai berikut :

Tabel 3.47. Proses Text Preprocessing Data Uji

| Tahap Text Preprocessing | Hasil Text Preprocessing |
|---------------------------|---------------------------------------------------------------------|
| case folding | wisatanya bersih sekali,, terus pemandunya juga ramah... mantab!!!! |
| remove punctuation | wisatanya bersih sekali terus pemandunya juga ramah mantab |
| remove number | wisatanya bersih sekali terus pemandunya juga ramah mantab |
| remove repeated character | wisatanya bersih sekali terus pemandunya juga ramah mantab |

Tabel 3.48. Lanjutan Proses Text Preprocessing Data Uji

| Tahap Text Preprocessing | Hasil Text Preprocessing |
|--------------------------|-------------------------------------------------------------------------------------|
| spelling normalization | wisatanya bersih sekali terus pemandunya juga ramah mantap |
| tokenizing | ['wisatanya', 'bersih', 'sekali', 'terus', 'pemandunya', 'juga', 'ramah', 'mantap'] |
| filtering | ['wisatanya', 'bersih', 'pemandunya', 'ramah', 'mantap'] |
| stemming | ['wisata', 'bersih', 'pandu', 'ramah', 'mantap'] |

Selanjutnya masuk ke proses tahapan pengujian *support vector machine* dengan seleksi fitur *mutual information* antara lain :

1. Hitung TF data uji terhadap data latih

Setelah dilakukan *text preprocessing* langkah selanjutnya menghitung TF data uji terhadap data latih dengan seleksi fitur *mutual information* pada tabel 3.40 sehingga didapatkan hasil yang dapat dilihat pada tabel 3.49, 3.50 dan 3.51.

Tabel 3.49. Hitung TF data uji terhadap data latih

| x | t1 | t2 | t3 | t4 | t5 | t6 | t7 | t8 | t9 | t10 | t11 | t12 | t13 | t14 | t15 |
|---|----|----|----|----|----|----|----|----|----|-----|-----|-----|-----|-----|-----|
| x | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Tabel 3.50. Lanjutan Hitung TF data uji terhadap data latih

| x | t16 | t17 | t18 | t19 | t20 | t21 | t22 | t23 | t24 | t25 | t26 | t27 |
|---|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| x | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Tabel 3.51. Lanjutan Hitung TF data uji terhadap data latih

| x | t28 | t29 | t30 | t31 | t32 | t33 | t34 | t35 | t36 | t37 | t38 | t39 | t40 |
|---|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| x | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |

Pada tabel 3.49, 3.50 dan 3.51 diperoleh bahwa fitur atau *term* data uji x terdapat pada t2, t3, t4, t6 dan t37.

2. Hitung kernelisasi data uji terhadap data latih

Pada tahap ini menghitung kernelisasi data uji terhadap data latih menggunakan kernel linear dengan mengalikan matriks fitur data uji x dengan fitur yang terdapat pada data latih (Dokumen 1 sampai 7), sehingga diperoleh hasil yang dapat dilihat pada tabel 3.52.

Tabel 3.52. Hasil Hitung kernelisasi data uji terhadap data latih

| | x1 | x2 | x3 | x4 | x5 | x6 | x7 |
|---------|-----------|----|----|----------|----------|----------|----------|
| k(x,xi) | 0.0097828 | 0 | 0 | 0.158283 | 0.818278 | 0.209987 | 0.158283 |

3. Penentuan kelas dengan fungsi keputusan klasifikasi

Penentuan kelas data uji dengan fungsi keputusan menggunakan persamaan 2.14 yaitu fungsi keputusan klasifikasi $sign(f(x))$ dimana nilai a dan b diambil dari proses pelatihan metode SVM dengan seleksi fitur *mutual information* sedangkan nilai $k(x,xi)$ diperoleh dari table 3.46. Fungsi $sign(f(x))$ merupakan fungsi normalisasi, jika nilai x di dalam fungsi $sign(f(x))$ lebih dari 0 atau > 0 maka fungsi tersebut memberikan nilai 1 (kelas positif). Jika nilai x dalam fungsi $sign(f(x))$ kurang dari 0 atau < 0 maka fungsi tersebut memberikan nilai -1 (kelas negatif). Berikut ini merupakan perhitungan penentuan kelas data uji dengan fungsi keputusan klasifikasi $sign(f(x))$:

$$\begin{aligned}
 f(x) &= sign[(2(-1)(0.0097828) + 1(-1)(0) + 1(-1)(0) + 1(1)(0.158283) + \\
 &\quad 1(1)(0.818278) + 1(1)(0.209987) + 1(1)(0.158283)) + 3.9576436] \\
 &= sign[(-0.0195656 + 0 + 0 + 0.158282694 + 0.818277904 + \\
 &\quad 0.209987194 + 0.158282694) + 3.9576436] \\
 &= sign[1.325264886 + 3.9576436]
 \end{aligned}$$

$$\begin{aligned}
 &= \text{sign}[5.282908486] \\
 &= 1
 \end{aligned}$$

Berdasarkan hasil perhitungan fungsi keputusan klasifikasi $\text{sign}(f(x))$ maka data uji $x = \text{"Wisatanya bersih sekali,, terus pemandunya juga ramah... MANTAB!!!!"}$ masuk ke dalam kelas positif (sentimen positif).

3.1.7 Validasi dan pengujian model

Pada bagian ini merupakan rancangan pengujian model yang akan untuk menguji model yang dibuat oleh sistem. Metode pengujian model yang akan dibuat pada penelitian ini antara lain *confusion matrix* dan *k-fold cross validation*.

3.1.7.1 Confusion matrix

Confusion matrix merupakan pengujian model untuk mengukur akurasi, presisi dan *recall* dari model yang dibangun. *Confusion matrix* memiliki empat kemungkinan yang merepresentasikan hasil dari proses klasifikasi, diantaranya *True Positive* (TP), *True Negative* (TN), *False Positive* (FP), *False Negative* (FN). Rancangan empat kemungkinan *confusion matrix* dapat dilihat pada tabel 3.53 dan 3.54.

Tabel 3.53. Confusion Matrix Metode SVM dengan Chi Square

| K-Fold | Confusion Matrix | | | |
|--------|------------------|----|----|----|
| | TP | FN | FP | TN |
| 1 | | | | |
| 2 | | | | |
| 3 | | | | |
| 4 | | | | |
| 5 | | | | |
| 6 | | | | |
| 7 | | | | |

Tabel 3.54. Confusion Matrix Metode SVM dengan Mutual Information

| K-Fold | Confusion Matrix | | | |
|--------|------------------|----|----|----|
| | TP | FN | FP | TN |
| 1 | | | | |
| 2 | | | | |
| 3 | | | | |
| 4 | | | | |
| 5 | | | | |
| 6 | | | | |
| 7 | | | | |

3.1.7.2 K-fold cross validation

K-fold cross validation digunakan untuk memvalidasi proses pengujian model yang dibangun. Pada penelitian ini akan menggunakan iterasi $k = 7$ dimana sebanyak 2700 dataset yang digunakan akan dibagi menjadi 7 bagian. Pada iterasi pertama dataset ke-1 sampai ke-385 akan digunakan sebagai data uji sedangkan dataset ke-386 sampai 2700 akan digunakan sebagai data latih. Rancangan *k-fold cross validation* yang dibuat dapat dilihat pada tabel 3.55, 3.57 dan 3.58.

Tabel 3.55. Pengujian K-Fold Cross Validation Metode SVM dengan Chi Square

| K-Fold | Akurasi | Presisi | Recall |
|--------|---------|---------|--------|
| 1 | | | |
| 2 | | | |
| 3 | | | |
| 4 | | | |
| 5 | | | |

Tabel 3.56. Lanjutan Pengujian K-Fold Cross Validation Metode SVM dengan Chi Square

| | | | |
|------------------|--|--|--|
| 6 | | | |
| 7 | | | |
| Rata-Rata | | | |

Tabel 3.57. Pengujian K-Fold Cross Validation Metode SVM dengan Mutual Information

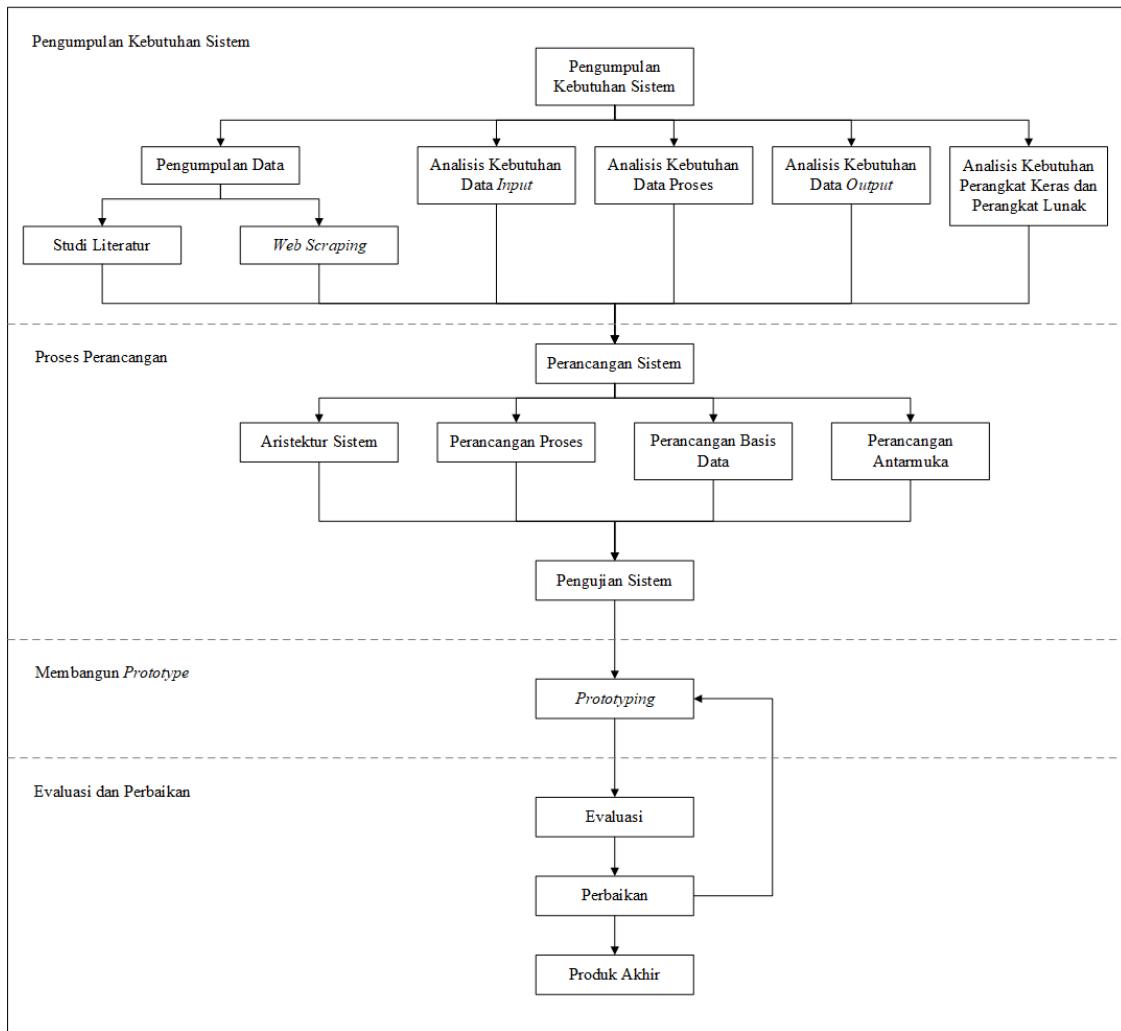
| K-Fold | Akurasi | Presisi | Recall |
|------------------|---------|---------|--------|
| 1 | | | |
| 2 | | | |
| 3 | | | |
| 4 | | | |
| 5 | | | |
| 6 | | | |
| 7 | | | |
| Rata-Rata | | | |

Tabel 3.58. Perbandingan Hasil Rata-Rata Pengujian K-Fold Cross Validation

| Akurasi | | Presisi | | Recall | |
|-------------------|--------|-------------------|--------|-------------------|--------|
| SVM+Chi Square | SVM+MI | SVM+Chi Square | SVM+MI | SVM+Chi Square | SVM+MI |
| | | | | | |

3.2 Metodologi pengembangan sistem

Metodologi pengembangan sistem yang digunakan pada penelitian ini adalah metodologi pengembangan sistem *prototype*. Tahapan-tahapan dalam metodelogi pengembangan sistem *prototype* antara lain pengumpulan kebutuhan sistem, proses perancangan yang cepat, membangun *prototype*, evaluasi dan perbaikan (Ogedebe & Jacob, 2012). Pada tahapan pengumpulan kebutuhan sistem dibagi menjadi pengumpulan data, analisis kebutuhan data *input*, analisis kebutuhan proses, analisis kebutuhan *output*, analisis kebutuhan perangkat keras dan perangkat lunak. Pada bagian pengumpulan data dibagi menjadi mencari studi literatur terkait dengan penelitian dan melakukan *web scraping* pada ulasan di *google maps*. Pada tahapan proses perancangan dibagi menjadi arsitektur sistem, perancangan proses, perancangan basis data, perancangan antarmuka, dan pengujian sistem menggunakan pengujian *black box*. Pada tahapan membangun *prototype*, dilakukan dengan mengimplementasikan perancangan sistem yang dibuat ke dalam kode program. Pada tahapan evaluasi dan perbaikan dibagi menjadi evaluasi, perbaikan dan produk akhir. *Prototype* yang telah dibuat lalu dievaluasi, jika terdapat kekurangan maka akan dilakukan perbaikan dan pembuatan ulang *protoype*, setelah selesai *prototype* akan dievaluasi lagi, jika tidak ada perbaikan maka produk akhir pada sistem ini telah selesai. Berikut ini merupakan tahapan-tahapan dari metodologi pengembangan sistem yang digunakan dapat dilihat pada gambar 3.2.



Gambar 3.2. Metodologi Pengembangan Sistem

3.2.1 Pengumpulan kebutuhan sistem

Pengumpulan kebutuhan sistem merupakan proses mengidentifikasi kebutuhan yang diperlukan oleh sistem dalam penelitian ini. Pengumpulan kebutuhan sistem bertujuan untuk mempermudah pada perancangan sistem. Pengumpulan kebutuhan sistem yang akan dikembangkan pada penelitian ini sebagai berikut :

3.2.1.1 Pengumpulan data

Pengumpulan data merupakan proses mengumpulkan data-data yang diperlukan dalam proses penelitian ini. Proses pengumpulan data pada penelitian ini adalah studi literatur dan *web scraping*. Studi literatur merupakan proses mengumpulkan jurnal, buku, artikel terkait mengenai penelitian ini. Studi literatur ini bertujuan untuk memberikan informasi mengenai metode yang digunakan dalam penelitian ini, sehingga dapat manambah pemahaman dan pengembangan dalam penelitian ini. Sedangkan *web scraping* merupakan proses mengambil data yang terdapat didalam fitur ulasan di *google maps* dengan teknik *scraping*. Data yang diambil sebanyak 100 komentar untuk setiap pariwisata. Data daftar pariwisata di Kota Yogyakarta yang akan dianalisis sentimen diambil dari situs *visitingjogja.com* (situs resmi Dinas Pariwisata Daerah Istimewa Yogyakarta) yaitu sebanyak 27 pariwisata. Kemudian, data ulasan di *google maps* ini akan melalui proses

labeling oleh 14 orang responden yang terpilih. Masing-masing responden akan melabeli minimal sebanyak 100 ulasan.

3.2.1.2 Analisis kebutuhan data *input*

Kebutuhan data *input* merupakan data yang digunakan sebagai data masukan pada sistem yang akan dibangun. Berikut merupakan kebutuhan data *input* yang digunakan pada penelitian ini :

1. Data ulasan di *Google Maps Review*

Data ulasan pariwisata di Kota Yogyakarta diperoleh dari *Google Maps Review* dengan cara *scrapping*. Jumlah daftar pariwisata di Kota Yogyakarta yang digunakan pada penelitian ini sebanyak 27 pariwisata, masing-masing pariwisata akan diambil ulasan sebanyak 100 komentar. Kemudian, data tersebut akan dibagi menjadi dua bagian yaitu data *training* (data latih) dan data uji. Data *training* digunakan untuk membangun model klasifikasi sedangkan data uji digunakan untuk menguji akurasi, presisi, *recall* pada algoritma klasifikasi yang digunakan pada penelitian ini.

2. Data *Spelling Normalization*

Spelling Normalization merupakan proses untuk memperbaiki *term* (kata) yang salah eja maupun disingkat dengan bentuk tertentu (kata yang tidak baku) menjadi kata baku. Data kamus ini memuat *term* seperti “bgt”, “bnget”, “bnngt” yang memiliki arti “banget”.

3. Data *Filtering (Stopword Removal)*

Filtering atau *Stopword Removal* merupakan proses membuang kata-kata yang tidak penting yang didapat dari hasil token seperti “yang”, “di”, “dari”, “untuk”, “ke”. Data kamus *Filtering* atau *Stopword* diambil dari penelitian Tala pada tahun 2013 dan ditambah kamus buatan yang dibuat berdasarkan data latih yang digunakan.

4. Data *Stemming*

Proses *Stemming* adalah tahapan untuk mengubah kata atau token hasil *filtering* menjadi kata dasarnya dengan menghilangkan imbuhan pada kata tersebut. Proses *stemming* pada penelitian ini, menggunakan data kamus dari pustaka Sastrawi.

3.2.1.3 Analisis kebutuhan data proses

Kebutuhan data proses merupakan data yang digunakan pada saat sistem sedang mengelola data *input* menjadi data *output*. Berikut merupakan kebutuhan data proses yang digunakan pada penelitian ini :

1. Proses *Text Preprocessing*

Pada proses *text preprocessing* bertujuan untuk mengurangi *noise* atau membersihkan data pada data latih atau data uji sebelum digunakan, dengan cara menyamakan struktur dan mengurangi dimensi yang melebar. Tahapan *text preprocessing* pada penelitian ini yaitu *case folding*, *remove punctuation*, *remove number*, *remove repeated characters*, *spelling normalization*, *tokenizing*, *filtering*, *stemming*.

2. Proses Ekstraksi Fitur (*Term Weighting TF-IDF*)

Proses ekstraksi fitur (*Term Weighting TF-IDF*) merupakan proses pembobotan setiap fitur atau *term* setelah proses *text preprocessing*. *Term Weighting TF-IDF* digunakan karena menyertakan frekuensi suatu *term* yang berupa kata, frasa atau unit hasil indeks lainnya dalam dokumen yang dapat digunakan untuk mengetahui konteks dari dokumen tersebut. Maka, untuk setiap kata tersebut diberikan indikator yang disebut *term weight*.

3. Proses Seleksi Fitur

Proses seleksi fitur merupakan proses memilih fitur atau *term* yang relevan yang mempengaruhi hasil klasifikasi, sehingga dapat mengurangi dimensi data, meningkatkan efektifitas dan efisiensi kinerja dari metode klasifikasi svm. Pada penelitian ini, akan membandingkan seleksi fitur *mutual information* dan *chi square*.

4. Proses Data Latih dan Data Uji

Proses data latih bertujuan untuk membentuk sebuah model dengan algoritma SVM dengan seleksi fitur *chi square* dan SVM dengan seleksi fitur *Mutual Information*. Proses data uji merupakan proses untuk menguji model yang sudah dibentuk. Proses klasifikasi pada penelitian ini dibagi menjadi kelas sentimen positif dan kelas sentimen negatif.

3.2.1.4 Analisis kebutuhan data output

Kebutuhan data *output* merupakan data keluaran yang dihasilkan dari proses yang dilakukan oleh sistem. Berikut merupakan kebutuhan data *output* yang digunakan pada penelitian ini :

1. Hasil *text preprocessing*, ekstraksi fitur (*Term Weighting TF-IDF*), seleksi fitur dari data ulasan di *google maps* yang siap diolah ke proses klasifikasi.
2. Statistik hasil klasifikasi kelas sentimen (positif dan negatif) yang berupa grafik.
3. Statistik hasil pengujian akurasi, presisi, *recall* dari sistem yang telah dibangun dalam melakukan klasifikasi berupa grafik.

3.2.1.5 Analisis kebutuhan perangkat keras dan perangkat lunak

Pada bagian ini akan dianalisis kebutuhan perangkat keras dan perangkat lunak yang diperlukan dalam membangun sistem. Berikut merupakan kebutuhan perangkat keras dan perangkat lunak yang digunakan pada penelitian ini :

1. Perangkat Keras (*Hardware*)

Tabel 3.59. Kebutuhan Perangkat Keras

| No | Perangkat Keras | Keterangan |
|----|------------------------------------------|------------------------------------------------------------------|
| 1 | <i>Processor</i> | AMD A10-9600P RADEON R5, 10 COMPUTE CORES 4C+6G (4CPUs), ~2.4GHz |
| 2 | <i>Storage</i> | 1 TB |
| 3 | RAM | 8 GB DDR4 |
| 4 | <i>Graphic</i> | AMD RADEON R5 dan R7 |
| 5 | Koneksi Internet | Wifi atau Kuota Internet |
| 6 | Perangkat <i>Input</i> dan <i>Output</i> | <i>Monitor, Mouse, Keyboard</i> |

2. Perangkat Lunak (*Software*)

Tabel 3.60. Kebutuhan Perangkat Lunak

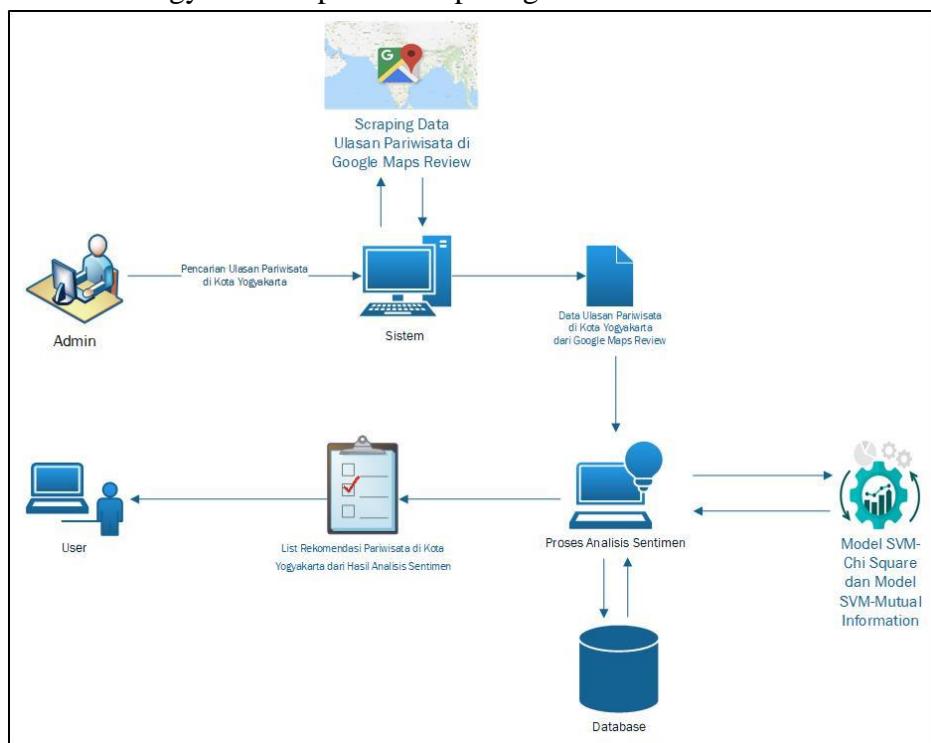
| No | Perangkat Lunak | Keterangan |
|----|----------------------|------------------------------------------------|
| 1 | Sistem Operasi | Windows 10 dengan 64-bit |
| 2 | <i>Sublime Text</i> | <i>Text Code Editor</i> |
| 3 | PHP | Bahasa Pemrograman |
| 4 | Python | Bahasa Pemrograman |
| 5 | MySQL | Tempat Menyimpan <i>Database</i> |
| 6 | XAMPP | Sebagai <i>Server Lokal</i> |
| 7 | <i>Corel Draw</i> | Membuat Rancangan <i>Interface</i> |
| 8 | <i>Visio</i> | Membuat Desain Diagram |
| 9 | <i>Colab</i> | <i>Text Code Editor</i> berbasis <i>online</i> |
| 10 | <i>Google Chrome</i> | <i>Web Browser</i> |
| 11 | <i>Draw.io</i> | Membuat Desain Diagram |

3.2.2 Perancangan sistem

Perancangan sistem merupakan tahapan merancangan sistem analisis sentimen ulasan di *google maps* menggunakan metode *support vector machine* dengan seleksi fitur *mutual information* dan *chi square* untuk rekomendasi destinasi tempat pariwisata di Kota Yogyakarta yang akan dikembangkan. Pada tahapan perancangan sistem ini dibagi menjadi beberapa tahapan antara lain arsitektur sistem, perancangan proses, perancangan basis data, dan perancangan antarmuka.

3.2.2.1 Arsitektur sistem

Perancangan arsitektur sistem yang akan dibangun pada pembuatan sistem analisis sentimen ulasan di *google maps* menggunakan metode *support vector machine* dengan seleksi fitur *mutual information* dan *chi square* untuk rekomendasi destinasi tempat pariwisata di Kota Yogyakarta dapat dilihat pada gambar 3.



Gambar 3.3. Arsitektur Sistem

Pada gambar 3.3. arsitektur sistem yang akan dibangun terdiri dari aktor admin dan *user* atau pengguna. Aktor admin dapat melakukan pencarian ulasan berdasarkan nama wisata dengan proses *scraping* data ulasan dari *google maps*. Data ulasan wisata yang telah didapatkan dari proses *scraping* kemudian akan masuk ke dalam proses analisis sentimen. Proses analisis sentimen menghasilkan *text preprocessing* dan hasil klasifikasi sentimen positif dan negatif. Klasifikasi sentimen positif dan negatif data ulasan ini, menggunakan model *svm* dengan seleksi fitur *chi square* dan model *svm* dengan seleksi fitur *mutual information* yang telah dibuat sebelumnya. Hasil dari proses analisis sentimen tersebut kemudian disimpan ke dalam *database* yang disebut data sentimen. Data sentimen inilah yang digunakan untuk pembobotan masing-masing wisata, hasil pembobotan tersebut kemudian diurutkan dari wisata yang memiliki bobot tertinggi ke terendah sehingga menghasilkan *list* rekomendasi pariwisata di Kota Yogyakarta yang menjadi referensi bagi

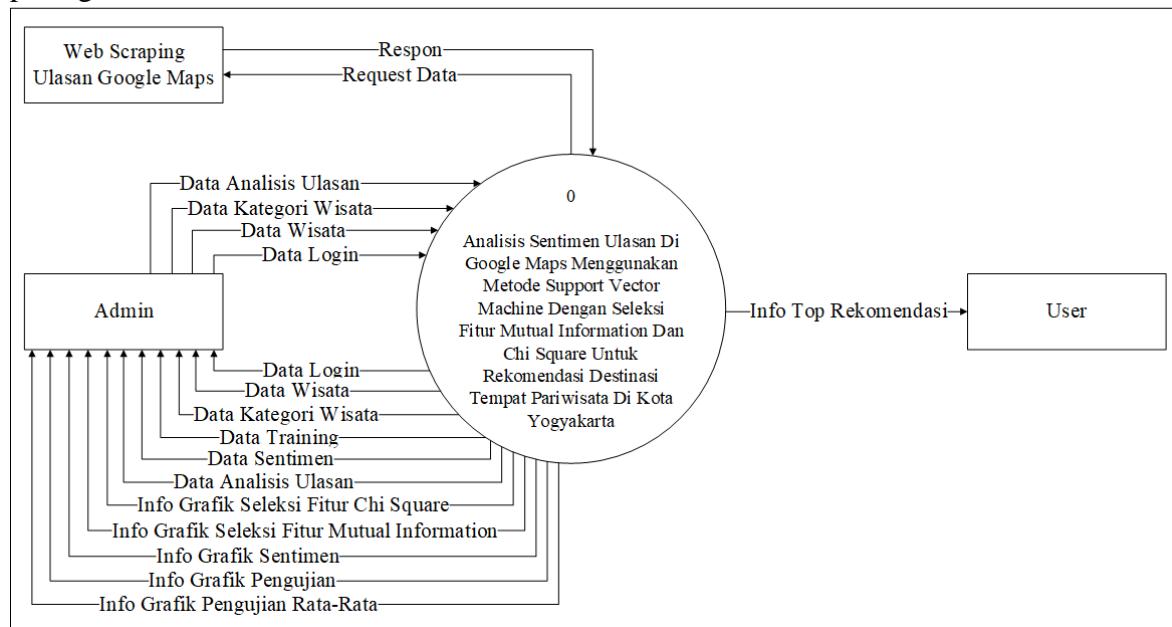
user. Aktor *user* pada sistem ini, dapat melihat *list* rekomendasi wisata untuk semua kategori, berdasarkan kategori, atau berdasarkan kata pencarian yang dimasukkan *user*, dan dapat melihat informasi mengenai destinasi wisata yang dicari.

3.2.2.2 Perancangan proses

Perancangan proses pada sistem ini terdiri dari *data flow diagram level 0*, *data flow diagram level 1*, dan *data flow diagram level 2*. *Data flow diagram* (DFD) merupakan suatu diagram yang menggambarkan alur informasi untuk setiap proses dari segi masukan maupun keluaran data.

a) Data flow diagram level 0

Data flow diagram (DFD) *level 0* atau yang disebut juga diagram konteks merupakan diagram yang memiliki tingkatan tertinggi dalam DFD yang menggambarkan keseluruhan proses dalam sistem, dan biasa diberi nomor nol. DFD *level 0* pada sistem ini dapat dilihat pada gambar 3.4.

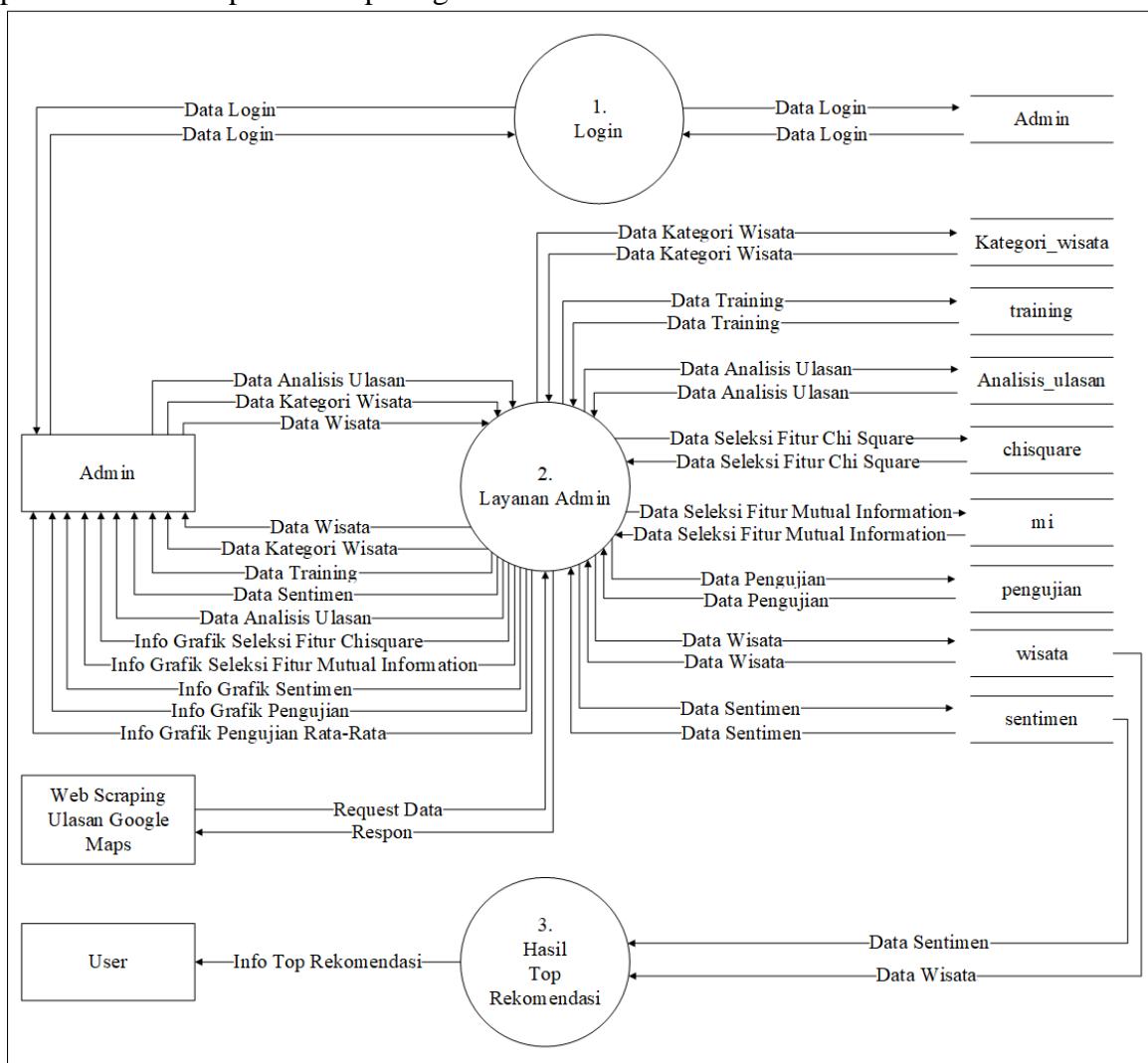


Gambar 3.4. Data Flow Diagram Level 0

Pada penelitian ini DFD *level 0* memiliki satu buah proses yaitu analisis sentimen ulasan di *google maps* menggunakan metode *support vector machine* dengan seleksi fitur *mutual information* dan *chi square* untuk rekomendasi destinasi tempat pariwisata di Kota Yogyakarta, dan tiga entitas antara lain admin, user, *web scraping* ulasan *google maps*. Pertama, pada entitas admin terdiri dari empat aktivitas *input* yaitu memberikan data analisis ulasan, data kategori wisata, data wisata, data login. Admin juga memiliki sebelas aktivitas *output* yaitu menerima konfirmasi data *login*, data wisata, data kategori wisata, data *training*, data sentiment, data analisis ulasan, info grafik seleksi fitur *chi square*, info grafik seleksi fitur *mutual information*, info grafik sentimen, info grafik pengujian, dan grafik pengujian rata-rata. Kedua, pada entitas user hanya memiliki satu aktivitas *output* yaitu menerima info top rekomendasi. Ketiga, entitas *web scraping* ulasan di *google maps* satu aktivitas *input* (meminta *request data*) dan satu aktivitas *ouput* (respon).

b) Data flow diagram level 1

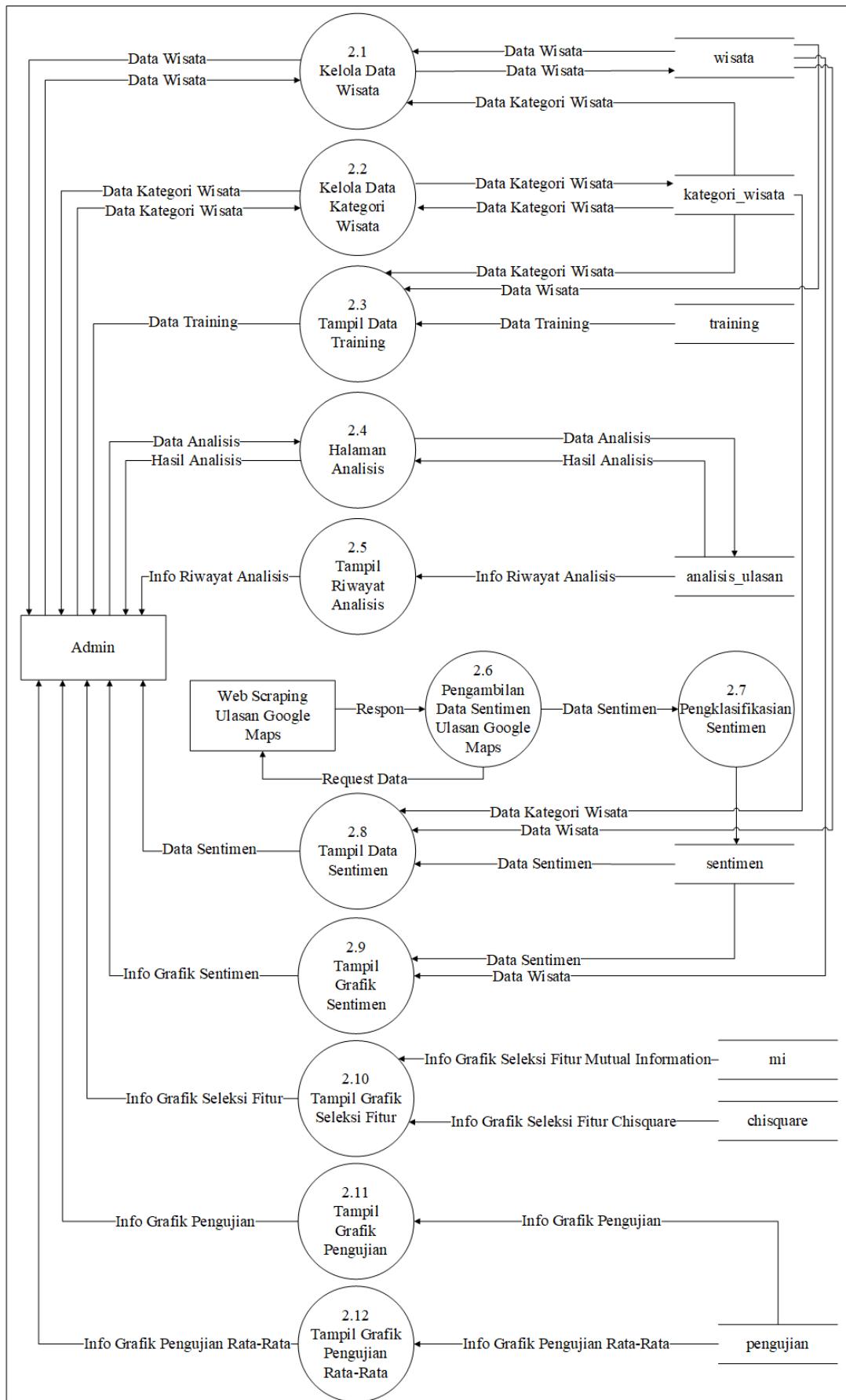
Data flow diagram (DFD) level 1 merupakan diagram penjabaran atau pemecahan semua proses pada DFD level 0 yang memuat penyimpanan data. Pada DFD level pada penelitian ini, terdapat tiga buah entitas, tiga buah proses dan tiga buah penyimpanan. Setiap proses pada DFD level 1 ini menggambarkan arus aliran data serta interaksi antara entitas (*user*, *admin*, *web scraping ulasan google maps*) dan penyimpanan (*database*). Pertama pada proses *login*, *admin* memasukkan data *admin* untuk *login* ke sistem, kemudian sistem akan mengecek data tersebut dengan yang ada didalam *database*, jika data ditemukan atau benar maka *admin* akan mendapatkan konfirmasi untuk data *login*-nya. Kedua pada proses layanan *admin*, *admin* dapat melakukan aktivitas *input* data-data ke dalam sistem, kemudian sistem ini menghubungkan dan memproses data-data tadi ke dalam *database* sehingga *admin* dapat melakukan pencarian, tambah data, edit data, hapus data, atau menampilkan data. Pada proses layanan *admin* juga dapat melakukan *request* data dari entitas *web scraping ulasan di google maps* dan mendapatkan respon hasil *web scraping* data ulasan. Ketiga, pada proses hasil top rekomendasi entitas *user* dapat menerima info top rekomendasi dari data sentimen dan data wisata yang diambil dari *database* sentimen dan wisata. Detail dari DFD level 1 pada sistem ini dapat dilihat pada gambar 3.5.



Gambar 3.5. Data Flow Diagram Level 1

c) ***Data flow diagram level 2***

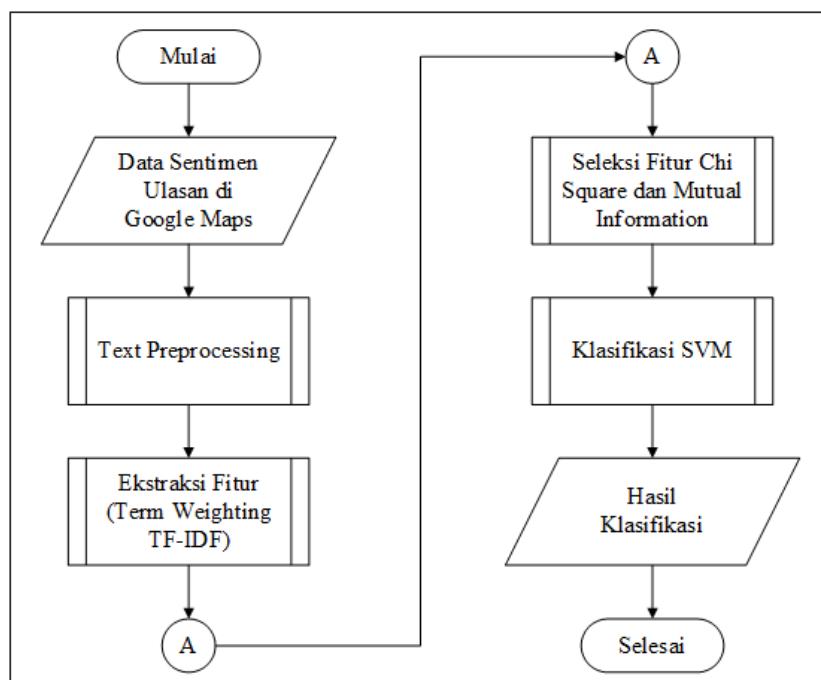
Data flow diagram (DFD) *level 2* pada sistem ini merupakan uraian proses layanan admin yang ada didalam DFD *level 1*. DFD *level 2* pada sistem ini, terdapat dua entitas(admin, *web scraping* ulasan *google maps*), dua belas proses, dan delapan penyimpanan (*database*). DFD *level 2* pada sistem ini menjelaskan arus aliran data yang masuk dari entitas ke setiap proses yang ada serta penyimpanan. Proses pada DFD *level 2* ini yaitu kelola data wisata, Kelola data kategori wisata, tampil data training, halaman analisis, tampil riwayat analisis, pengambilan data sentimen ulasan *google maps*, pengklasifikasian sentimen, tampil data sentimen, tampil grafik sentiment, tampil grafik seleksi fitur, tampil grafik pengujian dan tampil grafik pengujian rata-rata. Penyimpanan pada DFD *level 2* ini antara lain wisata, kategori_wisata, *training*, analisis_ulasan, sentimen, mi, *chisquare* dan pengujian. Pada proses pengambilan data sentimen ulasan *google maps* memerlukan entitas *web scraping* ulasan *google maps* untuk memperoleh data sentimen baru. Kemudian data sentimen tersebut masuk ke dalam proses pengklasifikasian sentimen, lalu data hasil klasifikasi masuk ke dalam penyimpanan sentimen. Pada proses tampil data sentimen mengambil data dari penyimpanan sentimen, wisata, dan kategori wisata agar dapat dilihat oleh entitas admin. Detail dari DFD *level 2* pada sistem ini dapat dilihat pada gambar 3.6.



Gambar 3.6. Data Flow Diagram Level 2

d) *Flowchart* analisis sentimen

Flowchart analisis sentimen merupakan penjelasan secara garis besar mengenai alur kerja sistem analisis sentimen yang dibangun. Tahap pertama, data sentimen ulasan di *google maps* yang sudah dilabeli secara manual akan masuk ke dalam tahap *text preprocessing*. Tahap *text preprocessing* ini bertujuan untuk mengubah data tidak terstruktur atau semi terstruktur menjadi data terstruktur sehingga data siap untuk diolah pada tahapan ekstraksi fitur (*term weighting* TF-IDF). Ekstraksi fitur (*term weighting* TF-IDF) merupakan tahapan pembobotan semua fitur atau *term* yang didapat dari data hasil proses *text preprocessing*. Setelah, semua fitur diberi bobot langkah selanjutnya melakukan pemilihan fitur yang memiliki pengaruh paling tinggi dan menghilangkan fitur yang kurang berpengaruh atau tidak berpotensi menjadi indeks dengan menggunakan metode seleksi fitur *chi square* dan metode *mutual information*. Fitur-fitur hasil pemilihan fitur dari proses seleksi fitur ini kemudian digunakan pada tahap klasifikasi menggunakan metode SVM. Pada tahap ini dibagi menjadi dua yaitu tahap pelatihan SVM dan pengujian SVM. Tahap pelatihan SVM merupakan tahapan pembuatan model klasifikasi yang akan dibangun yang nantinya digunakan untuk mengklasifikasikan data uji, sedangkan tahap pengujian SVM merupakan tahap menguji model klasifikasi yang dibangun dengan membandingkan fitur atau token dari data uji dengan data latih. Jika fitur pada data uji terdapat pada data latih maka akan diberi bobot, dan dihitung kernelisasi data uji terhadap data latih. Kemudian, data uji yang sudah di kernelisasi akan masuk ke proses penentuan kelas dengan fungsi keputusan yang akan menghasilkan hasil klasifikasi dari data uji apakah masuk ke dalam sentimen kelas positif atau negatif. Berikut ini merupakan *flowchart* analisis sentimen yang dapat dilihat pada gambar 3.7.

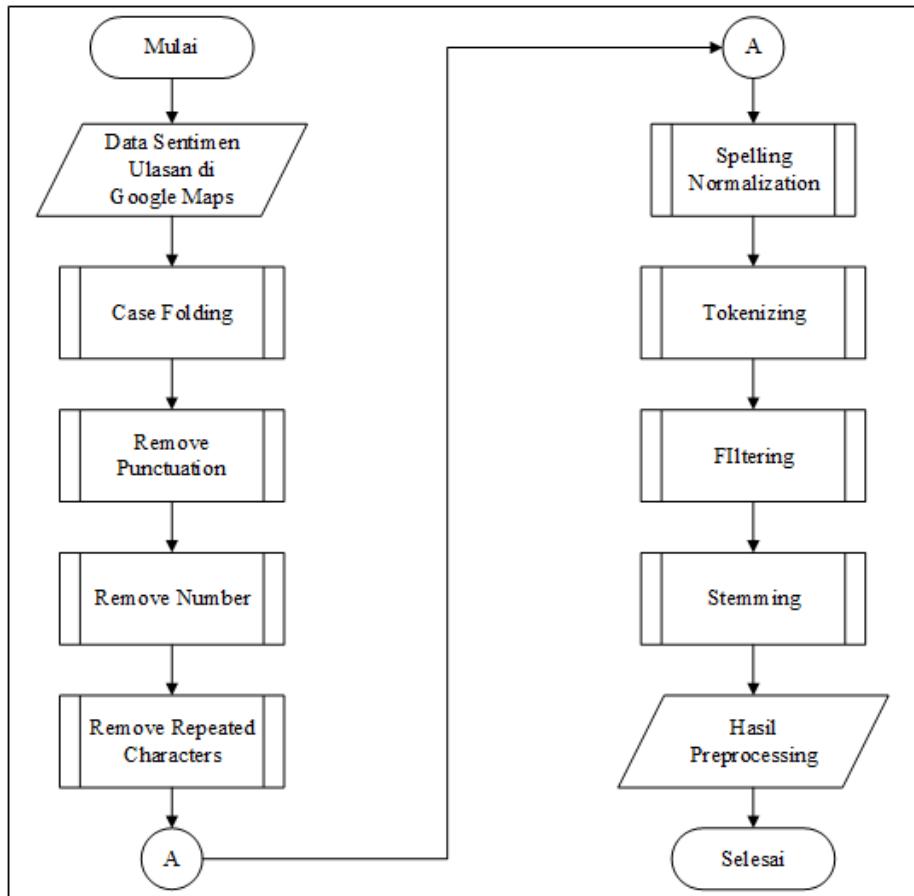


Gambar 3.7. *Flowchart* Analisis Sentimen

e) *Flowchart* *text preprocessing*

Flowchart *text preprocessing* merupakan penjelasan alur kerja dari proses *text preprocessing*. *Text preprocessing* merupakan tahapan mengubah data sentimen ulasan di

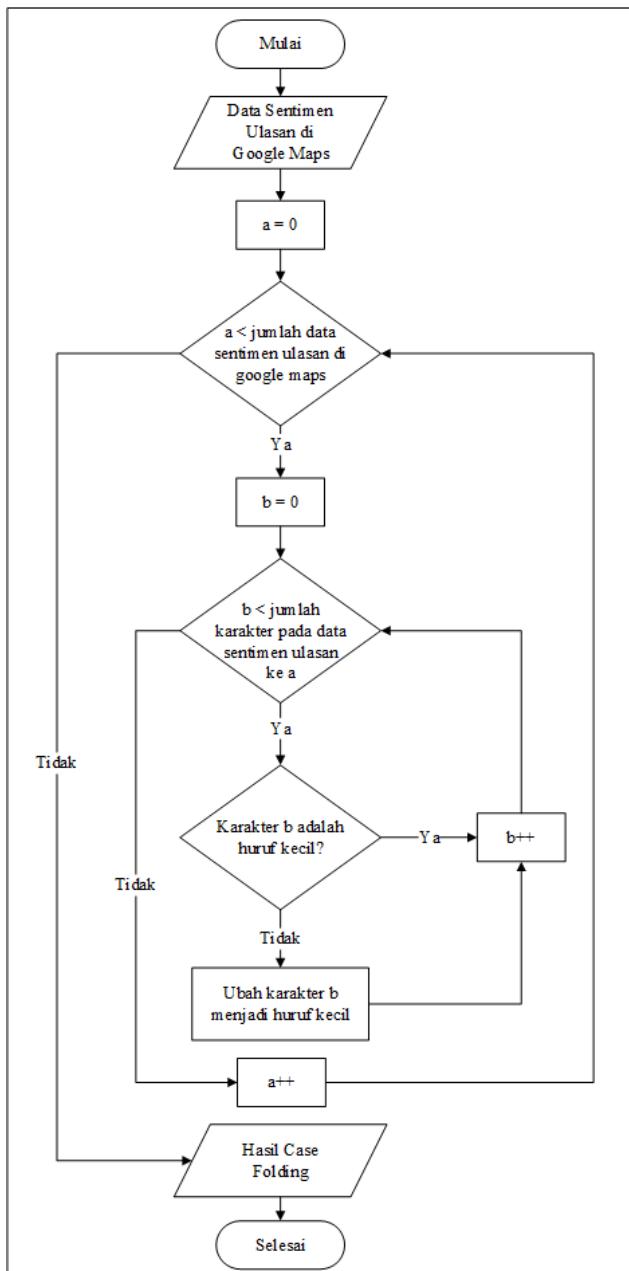
google maps yang tidak terstruktur atau semi terstruktur menjadi data yang terstruktur. *Text preprocessing* ini bertujuan untuk menyiapkan dataset untuk dapat diolah pada proses selanjutnya. Proses *text preprocessing* pada sistem ini antara lain *case folding*, *remove punctuation*, *remove number*, *remove repeated characters*, *spelling normalization*, *tokenizing*, *filtering* dan *stemming*. Berikut ini merupakan *flowchart text preprocessing* yang dapat dilihat pada gambar 3.8.



Gambar 3.8. *Flowchart Text Preprocessing*

f) *Flowchart case folding*

Flowchart case folding merupakan penjelasan alur kerja dari proses *case folding*. *Case folding* merupakan proses mengubah huruf *uppercase* (huruf kapital) menjadi huruf *lowercase* (huruf kecil). *Case folding* diperlukan karena tidak semua data dalam ulasan di *google maps* konsisten untuk menggunakan huruf kecil. Alur kerja dari proses *case folding* pertama, *case folding* akan menerima masukan berupa data sentimen ulasan di *google maps*. Kemudian, *case folding* akan menginisialisasi a sebagai jumlah data sentimen ulasan di *google maps* dan b sebagai jumlah karakter pada data sentimen ulasan ke-a. Kemudian akan melakukan pengecekan apakah karakter b merupakan huruf kecil, jika bukan huruf kecil maka karakter huruf b diubah menjadi huruf kecil dengan fungsi *.lower()*. Jika karakter b merupakan huruf kecil maka karakter tersebut disimpan dan berlanjut ke karakter b selanjutnya. Jika semua karakter pada ulasan ke a sudah diubah jadi huruf kecil maka lanjut ke ulasan a selanjutnya. *Case folding* akan melakukan proses ini secara berulang sampai semua data diubah menjadi huruf kecil. Berikut ini merupakan *flowchart case folding* yang dapat dilihat pada gambar 3.9.

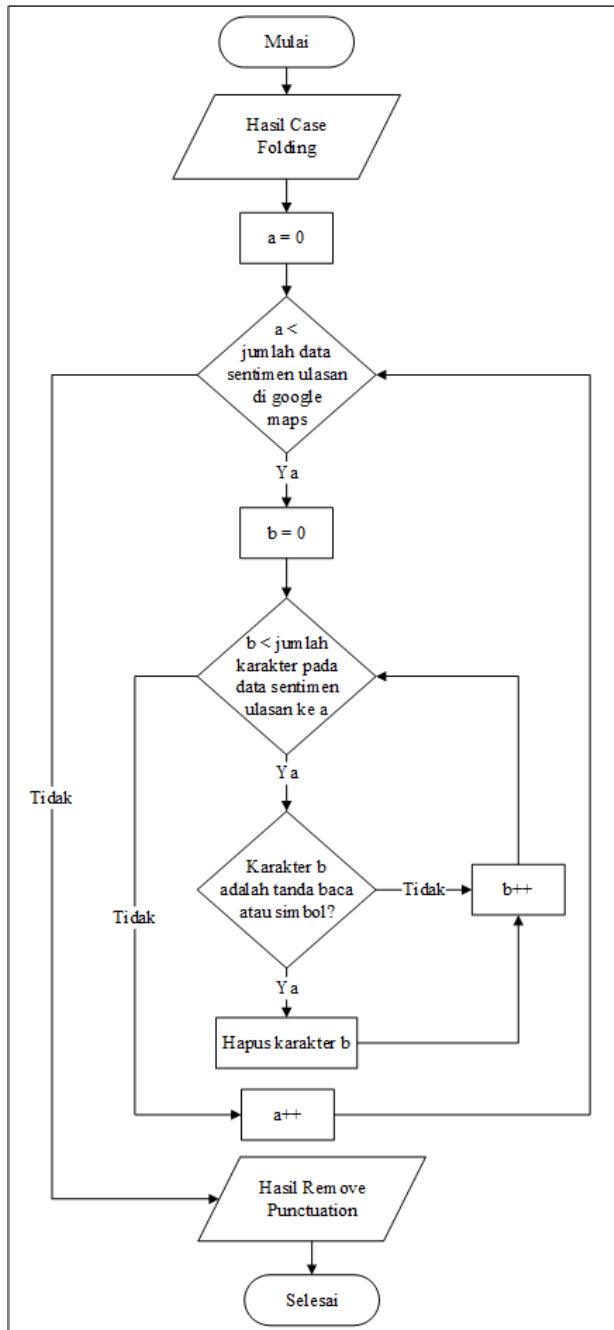


Gambar 3.9. Flowchart Case Folding

g) Flowchart remove punctuation

Flowchart *remove punctuation* merupakan penjelasan alur kerja dari proses *remove punctuation*. *Remove punctuation* merupakan proses untuk menghilangkan tanda baca dan simbol pada teks dokumen yang bertujuan untuk membersihkan elemen yang tidak berpengaruh terhadap hasil klasifikasi. Alur kerja dari proses *remove punctuation* yaitu akan menerima masukan berupa data hasil *case folding*. Kemudian, *remove punctuation* akan menginisialisasi a sebagai jumlah data sentimen ulasan di *google maps* dan b sebagai jumlah karakter pada data sentimen ulasan ke-a. Kemudian akan melakukan pengecekan apakah karakter b merupakan tanda baca atau simbol, jika ya maka karakter b akan dihapus. Jika karakter b bukan simbol atau tanda baca maka karakter tersebut disimpan dan berlanjut ke karakter b selanjutnya. Jika semua karakter pada ulasan ke a yang berupa tanda baca atau simbol sudah dihapus, maka lanjut ke ulasan a selanjutnya. *Remove punctuation* akan

melakukan proses ini secara berulang sampai semua data berupa tanda baca dan simbol dihapus. Berikut ini merupakan *flowchart case folding* yang dapat dilihat pada gambar 3.10.

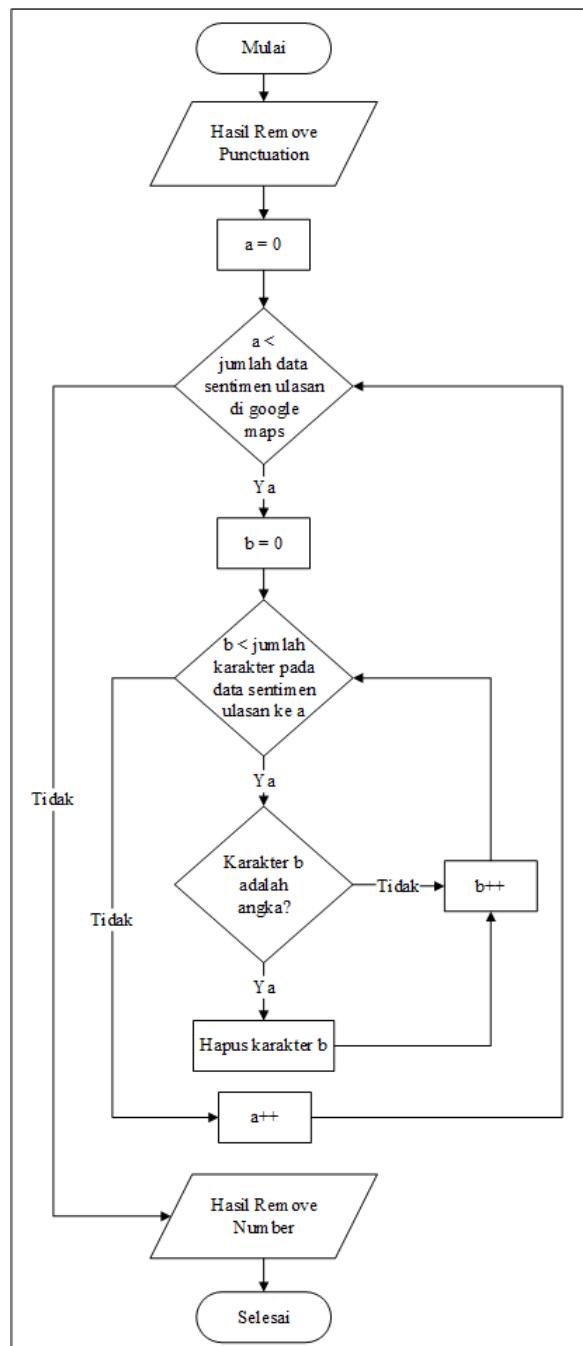


Gambar 3.10. Flowchart Remove Punctuation

h) Flowchart remove number

Flowchart remove number merupakan penjelasan alur kerja dari proses *remove number*. *Remove number* merupakan proses untuk menghilangkan angka pada teks dokumen, karena angka dianggap *delimiter* atau tidak berpengaruh terhadap hasil klasifikasi bahkan dapat mengganggu proses klasifikasi. Alur kerja dari proses *remove number* yaitu akan menerima masukan berupa data hasil *remove punctuation*. Kemudian, *remove number* akan menginisialisasi a sebagai jumlah data sentimen ulasan di *google maps* dan b sebagai jumlah karakter pada data sentimen ulasan ke-a. Kemudian akan melakukan pengecekan

apakah karakter b merupakan angka, jika ya maka karakter b akan dihapus. Jika karakter b bukan angka maka karakter tersebut disimpan dan berlanjut ke karakter b selanjutnya. Jika semua karakter pada ulasan ke a yang berupa angka sudah dihapus, maka lanjut ke ulasan a selanjutnya. *Remove number* akan melakukan proses ini secara berulang sampai semua data berupa angka dihapus. Berikut ini merupakan *flowchart remove number* yang dapat dilihat pada gambar 3.11.

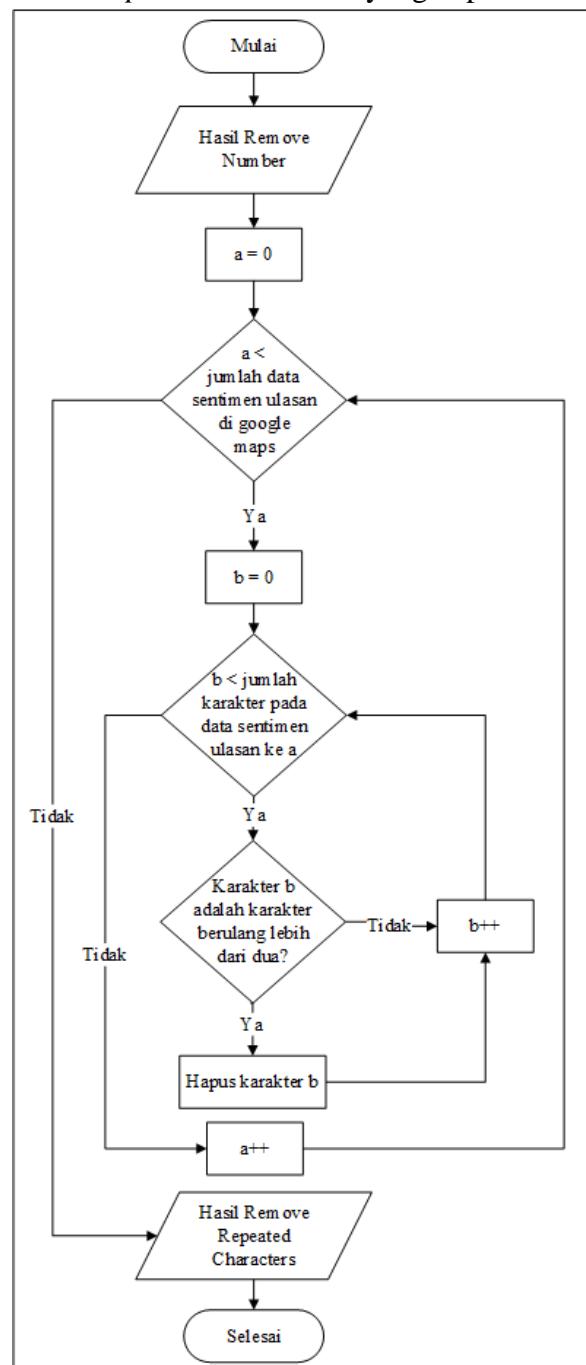


Gambar 3.11. Flowchart Remove Number

i) Flowchart remove repeated characters

Flowchart *remove repeated characters* merupakan penjelasan alur kerja dari proses *remove repeated characters*. *Remove repeated characters* merupakan proses untuk menghilangkan karakter yang memiliki jumlah lebih dari dua karena dapat mengganggu

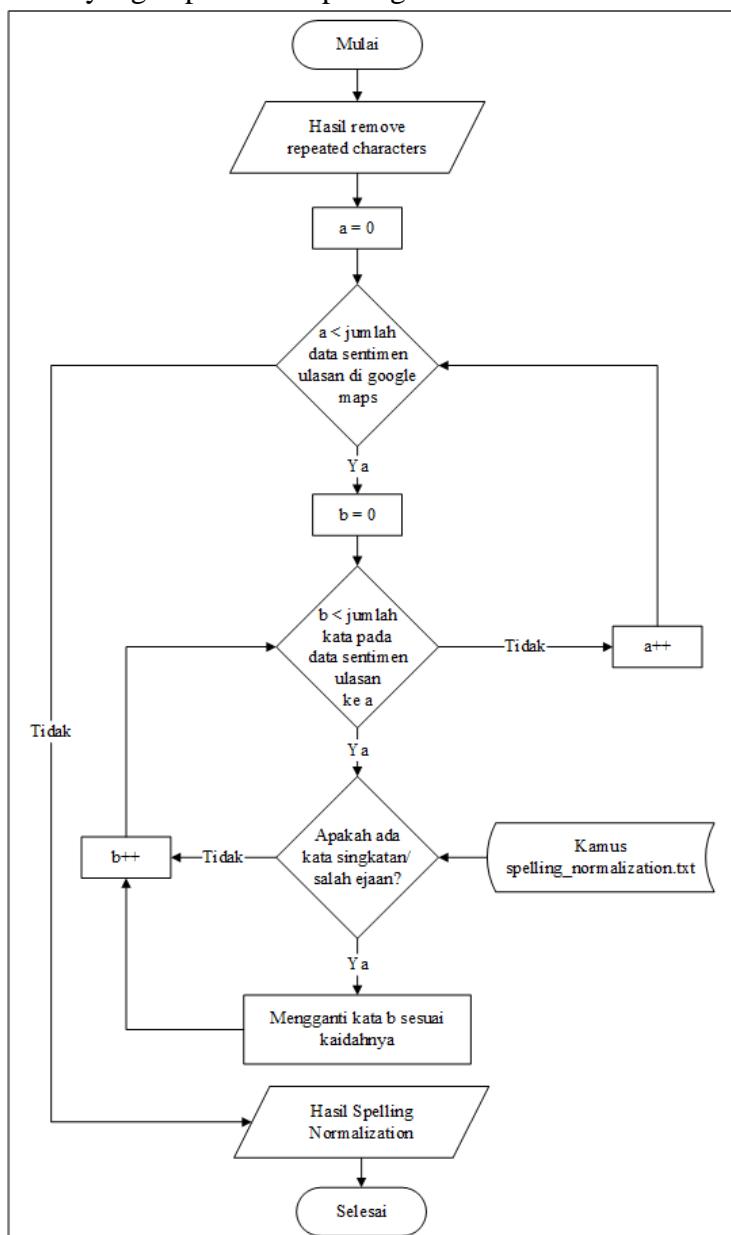
proses klasifikasi. Alur kerja dari proses *remove repeated characters* yaitu akan menerima masukan berupa data hasil *remove number*. Kemudian, *remove repeated characters* akan menginisialisasi a sebagai jumlah data sentimen ulasan di *google maps* dan b sebagai jumlah karakter pada data sentimen ulasan ke-a. Kemudian akan melakukan pengecekan apakah karakter b lebih dari dua, jika ya maka karakter b akan dihapus. Jika karakter b tidak lebih dari dua maka karakter tersebut disimpan dan berlanjut ke karakter b selanjutnya. Jika semua karakter pada ulasan ke a yang memiliki karakter lebih dari dua dihapus semua, maka lanjut ke ulasan a selanjutnya. *Remove repeated characters* akan melakukan proses ini secara berulang sampai semua data yang memiliki karakter lebih dari dua dihapus. Berikut ini merupakan *flowchart remove repeated characters* yang dapat dilihat pada gambar 3.12.



Gambar 3.12. Flowchart Remove Repeated Characters

j) Flowchart spelling normalization

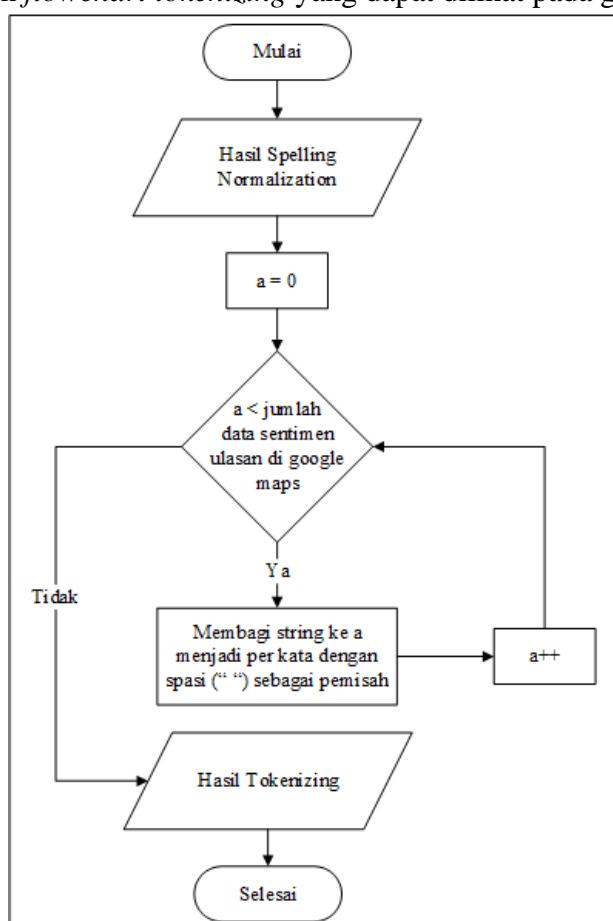
Flowchart *spelling normalization* merupakan penjelasan alur kerja dari proses *spelling normalization*. *Spelling normalization* merupakan proses perbaikan kata yang salah eja maupun singkatan menjadi bentuk kata aslinya. Pada penelitian ini menggunakan daftar kamus *spelling normalization* dari penelitian (Noehilasari, 2014) dan ditambah secara manual dari pengamatan data latih yang digunakan pada penelitian ini yang terdapat pada file *spelling_normalization.txt*. Alur kerja dari proses *spelling normalization* yaitu akan menerima masukan berupa data hasil *remove repeated characters* dan mengecek setiap kata pada data apakah merupakan singkatan atau salah ejaan. Jika kata merupakan singkatan atau salah ejaan maka akan diperbaiki sesuai kaidahnya, jika tidak maka akan disimpan. *Spelling normalization* akan melakukan proses ini secara berulang sampai semua data yang berupa singkatan atau salah ejaan diganti sesuai kaidahnya. Berikut ini merupakan *flowchart spelling normalization* yang dapat dilihat pada gambar 3.13.



Gambar 3.13. *Flowchart Spelling Normalization*

k) Flowchart tokenizing

Flowchart tokenizing merupakan penjelasan alur kerja dari proses *tokenizing*. *Tokenizing* adalah proses pemotongan *string input* menjadi beberapa kata berdasarkan tiap kata yang menyusunnya. Pada proses ini, data teks dokumen akan dipisah menjadi tiap kata berdasarkan spasi atau karakter lain sehingga menghasilkan *token*. Alur kerja dari proses *tokenizing* yaitu akan menerima masukan berupa data hasil *spelling normalization* dan menginisialisai *a* sebagai jumlah data sentimen ulasan di *google maps*. Kemudian tokenizing akan membagi string ke *a* menjadi per kata dengan spasi sebagai pemisah. Proses tokenizing ini akan dilakukan secara berulang sampai semua data sudah berupa token. Berikut ini merupakan flowchart *tokenizing* yang dapat dilihat pada gambar 3.14.

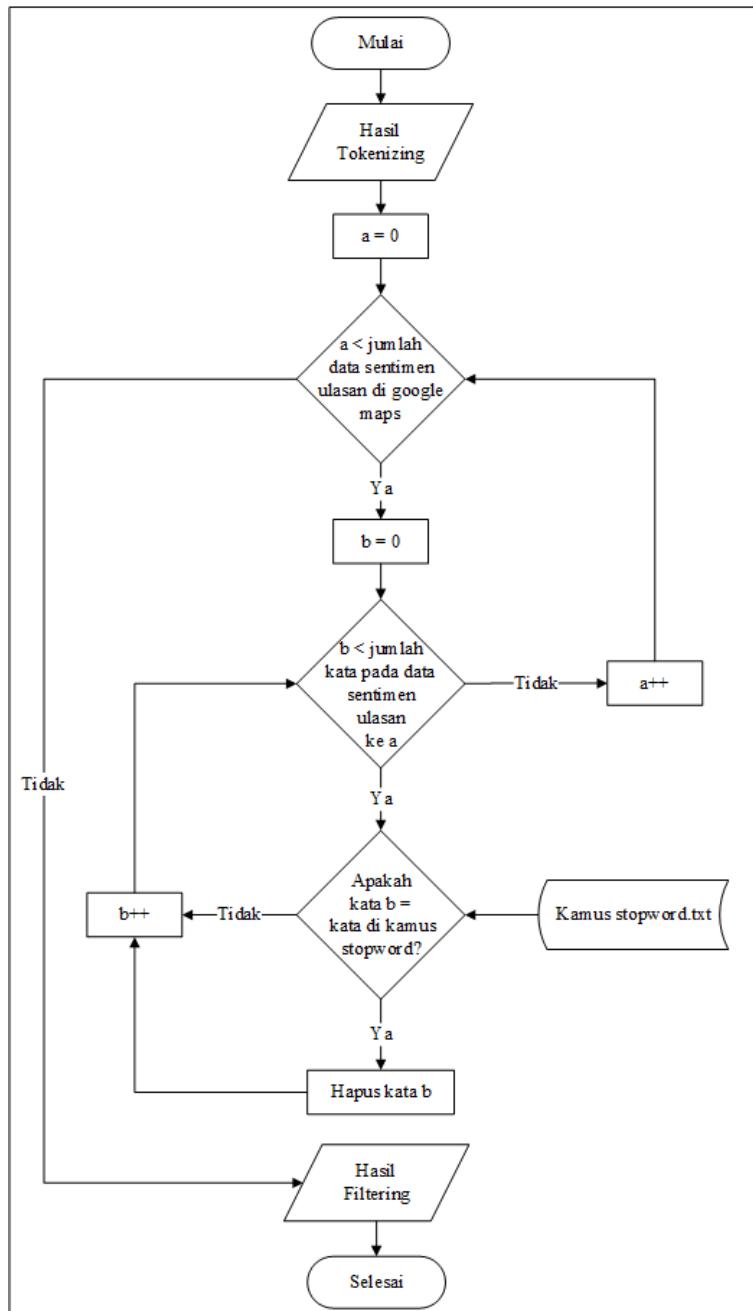


Gambar 3.14. Flowchart Tokenizing

l) Flowchart filtering

Flowchart filtering merupakan penjelasan alur kerja dari proses *filtering*. *Filtering* adalah proses mengambil kata-kata penting yang dihasilkan oleh *token*. Pada penelitian ini akan menggunakan algoritma *stopword removal* untuk membuang kata-kata yang tidak penting yang didapat dari hasil token seperti "yang", "di", "dari", "untuk", "ke" yang memiliki frekuensi kemunculan yang tinggi dalam sebuah teks dokumen. Pada penelitian ini menggunakan daftar kamus *stopword* dari Tala (2003) yang terdapat pada file *stopword.txt*. Alur kerja dari proses *filtering* yaitu akan menerima masukan berupa data hasil *tokenizing* dan mengecek setiap kata pada data apakah kata tersebut terdapat pada kamus *stopword*. Jika iya maka kata tersebut akan dihapus, jika tidak maka akan disimpan. *Filtering* akan

melakukan proses ini secara berulang sampai semua kata pada data yang merupakan *stopword* dihapus. Berikut ini merupakan *flowchart filtering* yang dapat dilihat pada gambar 3.15.

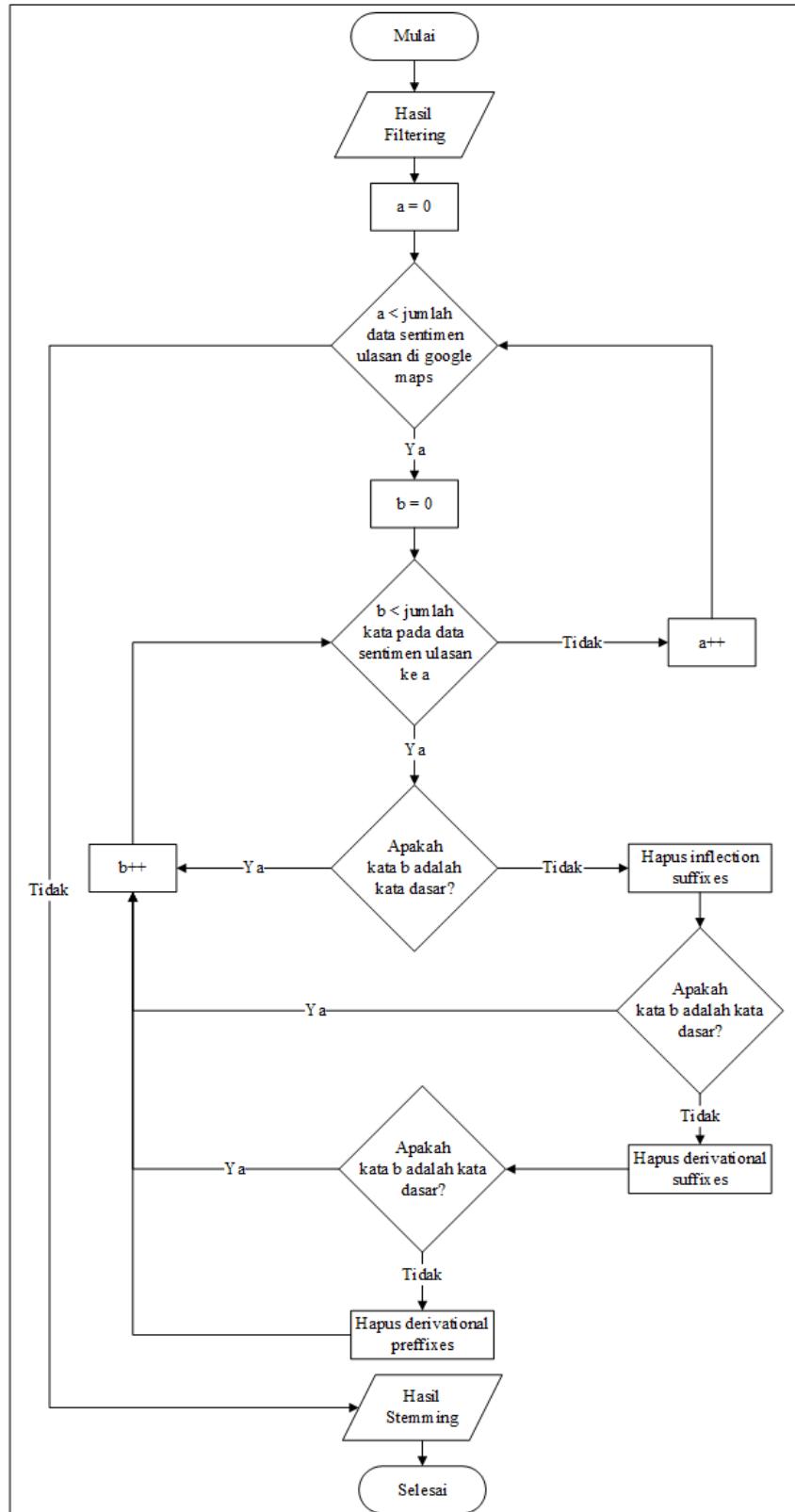


Gambar 3.15. *Flowchart Filtering*

m) *Flowchart stemming*

Flowchart stemming merupakan penjelasan alur kerja dari proses *stemming*. *Stemming* adalah tahapan untuk mengubah kata atau token hasil *filtering* menjadi kata dasarnya dengan menghilangkan imbuhan pada kata tersebut. Algoritma *stemming* yang digunakan pada penelitian ini adalah algoritma *stemming* Nazief dan Adriani. Alur kerja dari proses *stemming* yaitu akan menerima masukan berupa data hasil *filtering* dan mengecek setiap kata pada data apakah kata tersebut merupakan kata dasar. Jika iya maka kata tersebut akan disimpan, jika tidak maka *stemming* akan melakukan proses hapus *inflection*

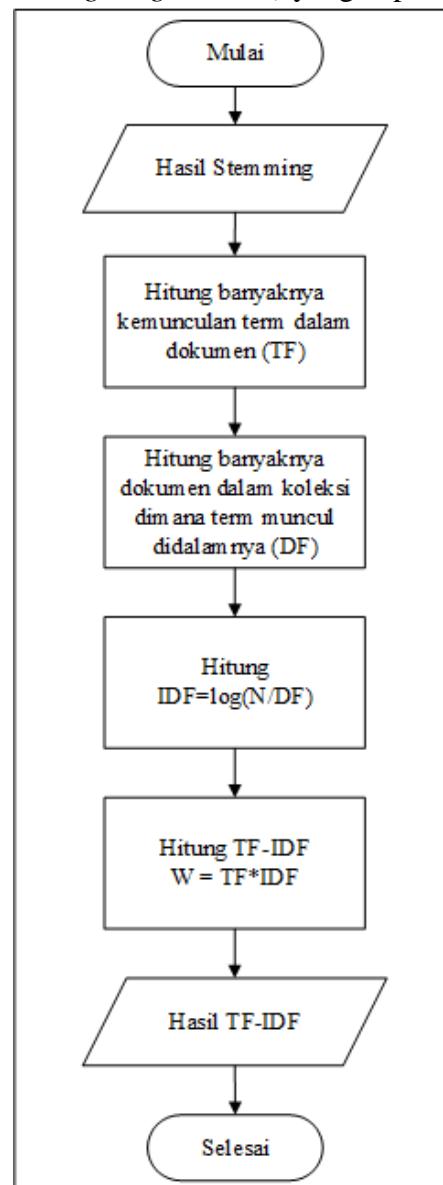
suffixes, hapus *derivational suffixes*, dan hapus *derivational prefixes* sehingga diperoleh kata dasar dari kata tersebut. Stemming akan melakukan proses ini secara berulang sampai semua kata pada data berupa kata dasar. Berikut ini merupakan *flowchart stemming* yang dapat dilihat pada gambar 3.16.



Gambar 3.16. *Flowchart Stemming*

n) Flowchart ekstraksi fitur (*term weighting* TF-IDF)

Flowchart ekstraksi fitur (*term weighting* TF-IDF) merupakan penjelasan alur kerja dari proses pembobotan fitur dengan TF-IDF. Alur kerja dari proses ekstraksi fitur (*term weighting* TF-IDF) yaitu akan menerima masukan berupa data hasil *stemming*, kemudian setiap fitur akan dihitung frekuensi kemunculannya dalam dokumen yang disebut *term frequency* (TF) dan dihitung banyaknya dokumen dalam koleksi dimana fitur tersebut muncul didalamnya *document frequency* (DF). Setelah itu, hitung nilai *inverse document frequency* (IDF), dan yang terakhir akan dihitung nilai *term frequency-invers document frequency* (TF-IDF). Ekstraksi fitur (*term weighting* TF-IDF) akan melakukan proses ini secara berulang sampai semua fitur atau kata diberi bobot semua. Berikut ini merupakan flowchart ekstraksi fitur (*term weighting* TF-IDF) yang dapat dilihat pada gambar 3.17.

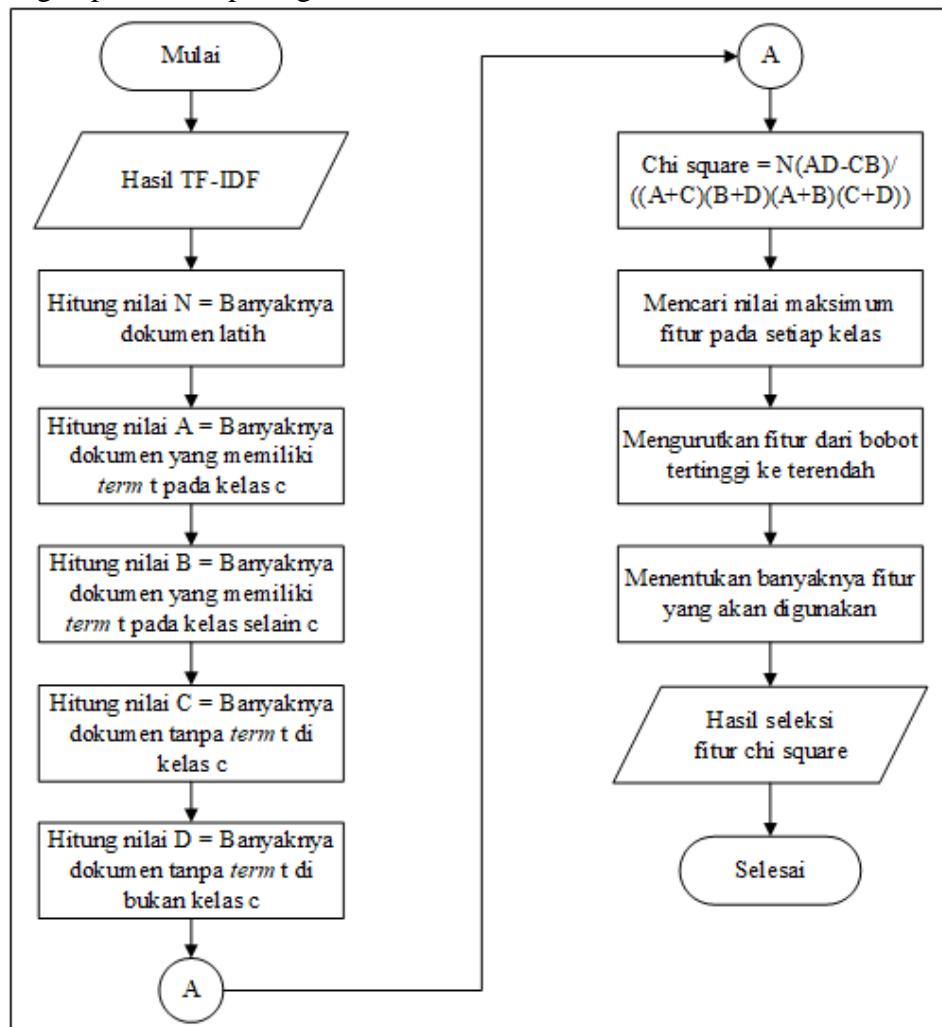


Gambar 3.17. *Flowchart Ekstraksi Fitur (Term Weighting TF-IDF)*

o) Flowchart seleksi fitur *chi square*

Flowchart seleksi fitur *chi square* merupakan penjelasan alur kerja dari proses seleksi fitur *chi square*. Alur kerja dari proses seleksi fitur *chi square* yaitu akan menerima

masukan berupa data hasil ekstraksi fitur (*term weighting* TF-IDF), kemudian hitung banyaknya data latih (N), banyaknya dokumen yang memiliki *term t* pada kelas c (A), banyaknya dokumen yang memiliki *term t* pada kelas selain c (B), banyaknya dokumen tanpa *term t* di kelas c (C), dan banyaknya dokumen tanpa *term t* di bukan kelas c. Kemudian hitung bobot *chi square* semua fitur pada kelas positif dan negatif. Bobot *chi square* dari kelas positif dan negatif dibandingkan untuk mendapatkan bobot *chi square* maksimal. Setelah itu, mengurutkan fitur dari bobot *chi square* tertinggi ke terendah dan menentukan banyaknya fitur yang akan digunakan. Berikut ini merupakan *flowchart* seleksi fitur *chi square* yang dapat dilihat pada gambar 3.18.

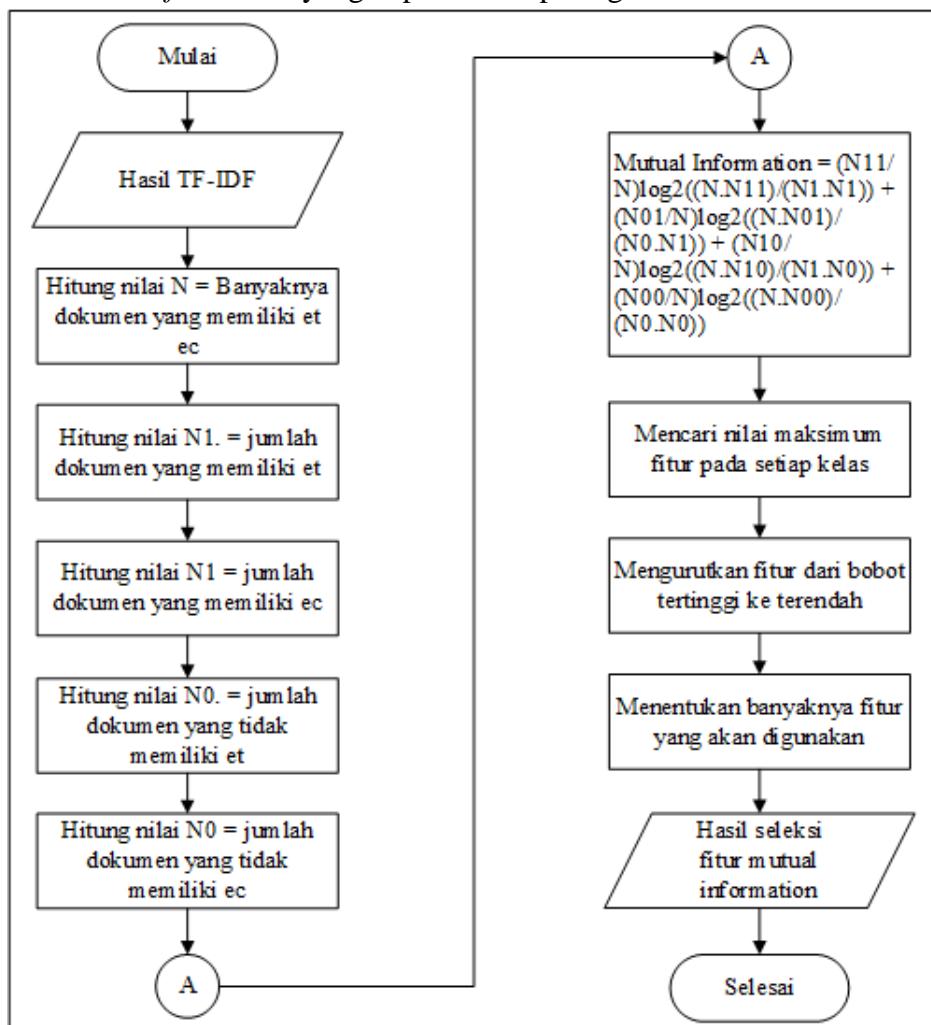


Gambar 3.18. *Flowchart* Seleksi Fitur *Chi Square*

p) *Flowchart* seleksi fitur *mutual information*

Flowchart seleksi fitur *mutual information* merupakan penjelasan alur kerja dari proses seleksi fitur *mutual information*. Alur kerja dari proses seleksi fitur *mutual information* yaitu akan menerima masukan berupa data hasil ekstraksi fitur (*term weighting* TF-IDF), kemudian hitung banyaknya dokumen yang memiliki et (N), jumlah dokumen yang memiliki et (N₁), jumlah dokumen yang memiliki ec (N₁), jumlah dokumen yang tidak memiliki et (N₀), dan jumlah dokumen yang tidak memiliki ec (N₀). Kemudian hitung bobot *mutual information* semua fitur pada kelas positif dan negatif. Bobot *mutual information* dari kelas positif dan negatif dibandingkan untuk mendapatkan bobot *mutual information*

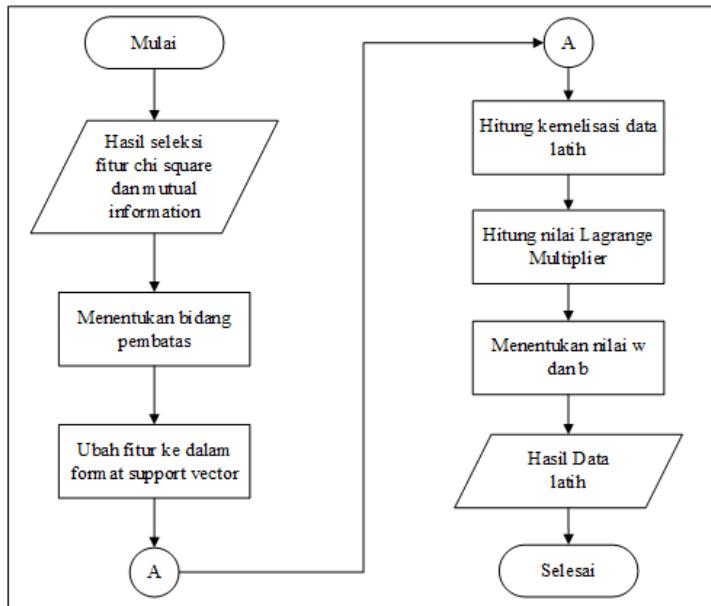
maksimal. Setelah itu, mengurutkan fitur dari bobot *mutual information* tertinggi ke terendah dan menentukan banyaknya fitur yang akan digunakan. Berikut ini merupakan *flowchart* seleksi fitur *mutual information* yang dapat dilihat pada gambar 3.19.



Gambar 3.19. *Flowchart Seleksi Fitur Mutual Information*

q) *Flowchart tahap pelatihan support vector machine*

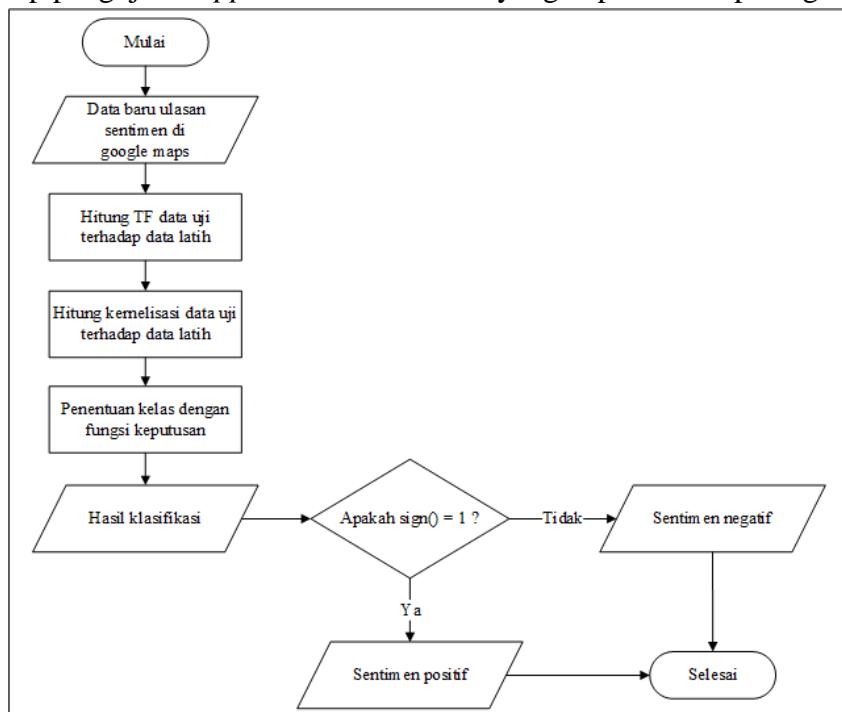
Flowchart tahap pelatihan *support vector machine* merupakan penjelasan alur kerja dari proses pelatihan metode klasifikasi *support vector machine*. Alur kerja dari proses pelatihan *support vector machine* yaitu akan menerima masukan berupa data hasil seleksi fitur *chi square* dan *mutual information*. Kemudian menentukan bidang pembatas dari data hasil seleksi fitur *chi square* dan hasil seleksi fitur *mutual information* dan mengubahnya ke dalam format *support vector*. Setelah data tersebut diubah menjadi format *support vector*, hitung kernelisasi data latih tersebut sehingga diperoleh matriks. Hasil matriks tersebut digunakan untuk menghitung nilai *lagrange multiplier* dan mencari nilai α yang menghasilkan nilai *lagrange multiplier* yang maksimal. Setelah itu, menentukan nilai w (bobot) dan nilai b (bias) untuk menghasilkan model *support vector machine* dengan seleksi fitur *chi square* dan model *support vector machine* dengan seleksi fitur *mutual information*. Berikut ini merupakan *flowchart* tahap pelatihan *support vector machine* yang dapat dilihat pada gambar 3.20.



Gambar 3.20. Flowchart Tahap Pelatihan Support Vector Machine

r) Flowchart tahap pengujian support vector machine

Flowchart tahap pengujian *support vector machine* merupakan penjelasan alur kerja dari proses pengujian metode klasifikasi *support vector machine*. Alur kerja dari proses pengujian *support vector machine* yaitu akan menerima masukan berupa data baru ulasan sentimen di *google maps* yang disebut data uji. Kemudian, hitung TF dari data uji terhadap data latih. Lakukan perhitungan kernelisasi data uji terhadap data latih. Hasil kernelisasi tersebut kemudian digunakan untuk penentuan kelas dengan fungsi keputusan *sign()* sehingga diperoleh hasil klasifikasi. Jika hasil dari fungsi *sign()* = 1 maka data uji merupakan sentimen positif jika tidak maka merupakan sentimen negatif. Berikut ini merupakan flowchart tahap pengujian *support vector machine* yang dapat dilihat pada gambar 3.21.



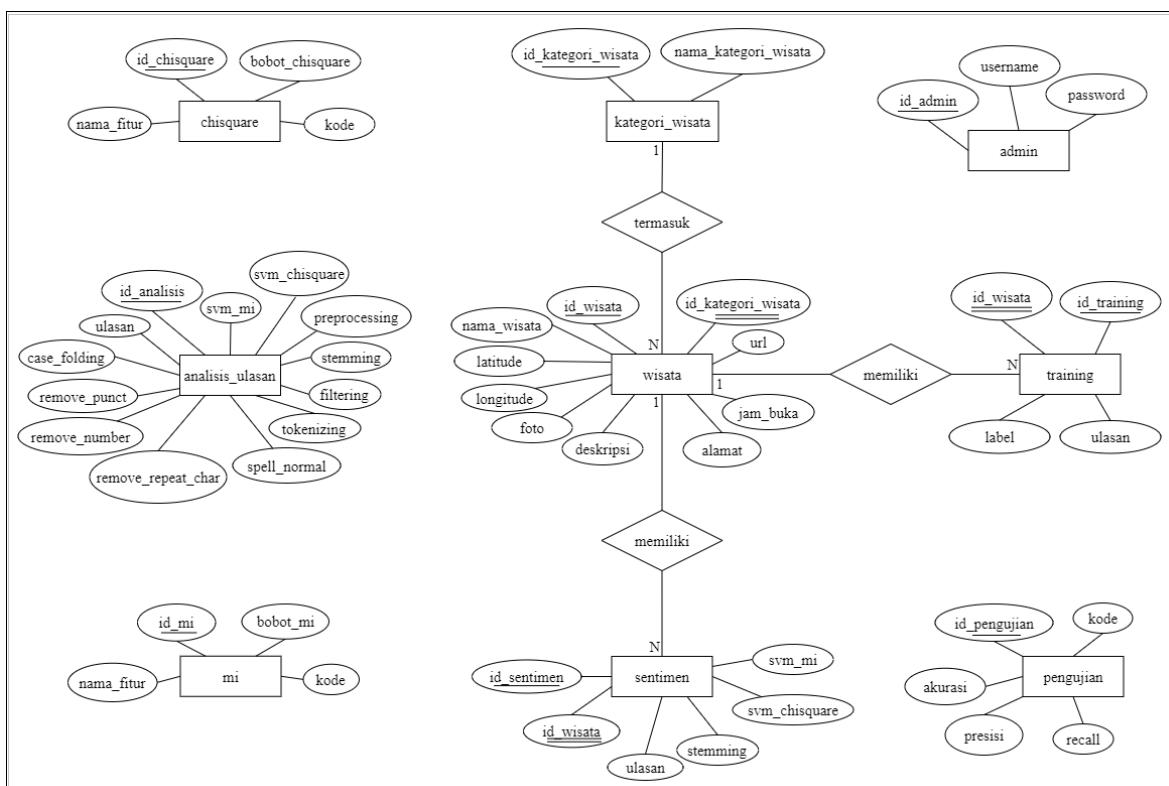
Gambar 3.21. Flowchart Tahap Pengujian Support Vector Machine

3.2.2.3 Perancangan basis data

Pada pembuatan sistem ini diperlukan penyimpanan ke dalam basis data. Basis data yang digunakan dalam penelitian adalah *mysql* dan *server http apache*. Basis data merupakan sekumpulan data yang disimpan dengan cara terorganisir atau tersusun secara sistematik yang dapat diakses informasinya dengan menggunakan kode program. Perancangan basis data dalam pengembangan sistem ini dibagi menjadi tiga antara lain *entity relationship diagram* (erd), struktur tabel, dan relasi antar tabel (rat).

a) Entity relationship diagram (ERD)

Entity relationship diagram (ERD) adalah diagram yang menggambarkan hubungan antara entitas dan atributnya yang digunakan untuk merancang sebuah basis data. *Entity relationship diagram* (ERD) ini terdiri dari empat komponen antara lain entitas, atribut, relasi, dan garis. Entitas merupakan suatu objek yang dapat dibedakan dengan hal lain atau suatu objek yang unik. Atribut merupakan bagian dari entitas yang berfungsi mendeskripsikan karakteristik entitas itu sendiri. Relasi merupakan hubungan antar entitas satu dengan entitas yang lainnya. Garis merupakan penghubung antara entitas dengan atribut atau penghubung kumpulan entitas yang berelasi. Pada sistem ini, terdapat sembilan entitas antara lain admin, wisata, kategori wisata, *training*, sentimen, analisis ulasan, *chisquare*, mi, dan pengujian. Berikut ini detail perancangan *entity relationship diagram* (ERD) yang digunakan dalam penerapan analisis sentimen ulasan di *google maps* menggunakan metode *support vector machine* dengan seleksi fitur *mutual information* dan *chi square* untuk rekomendasi destinasi tempat pariwisata di Kota Yogyakarta yang dapat dilihat pada gambar 3.22.



Gambar 3.22. Entity Relationship Diagram

b) Struktur tabel

Struktur tabel yang membangun basis data pada penerapan analisis sentimen ulasan di *google maps* menggunakan metode *support vector machine* dengan seleksi fitur *mutual information* dan *chi square* untuk rekomendasi destinasi tempat pariwisata di Kota Yogyakarta antara lain :

1. Tabel Admin

Tabel admin digunakan untuk menyimpan dan mengakses data admin. Tabel admin terdiri dari tiga atribut yaitu *id_admin*, *username*, *password*. *Id_admin* merupakan *primary key* dari tabel admin. Atribut *username* dan *password* digunakan untuk masuk ke dalam halaman admin. Struktur tabel admin dapat dilihat pada tabel 3.61.

Tabel 3.61. Struktur Tabel Admin

| No | Nama Fields | Tipe Data | Constraint | Keterangan |
|----|-----------------|-------------|------------------------------------|-----------------------|
| 1 | <i>id_admin</i> | Int(11) | <i>Primary_Key, Auto_Increment</i> | Id admin |
| 2 | <i>username</i> | Varchar(30) | <i>None</i> | <i>Username</i> admin |
| 3 | <i>password</i> | Varchar(30) | <i>None</i> | <i>Password</i> admin |

2. Tabel Wisata

Tabel wisata digunakan untuk menyimpan dan mengakses data wisata. Tabel wisata terdiri dari sepuluh atribut yaitu *id_wisata*, *id_kategori_wisata*, *nama_wisata*, *latitude*, *longitude*, foto, deskripsi, alamat, jam_buka dan *url*. *Id_wisata* merupakan *primary key* dari tabel wisata sedangkan, *id_kategori_wisata* merupakan *foreign key* dari tabel wisata. Struktur tabel wisata dapat dilihat pada tabel 3.62.

Tabel 3.62. Struktur Tabel Wisata

| No | Nama Fields | Tipe Data | Constraint | Keterangan |
|----|---------------------------|-------------|------------------------------------|-------------------------------|
| 1 | <i>id_wisata</i> | Int(11) | <i>Primary Key, Auto_Increment</i> | Id wisata |
| 2 | <i>id_kategori_wisata</i> | Int(11) | <i>Foreign Key</i> | Id kategori wisata |
| 3 | <i>nama_wisata</i> | Varchar(75) | <i>None</i> | Nama wisata |
| 4 | <i>latitude</i> | Double | <i>None</i> | Letak <i>latitude</i> wisata |
| 5 | <i>longitude</i> | Double | <i>None</i> | Letak <i>longitude</i> wisata |
| 6 | <i>foto</i> | Varchar(50) | <i>None</i> | Foto wisata |
| 7 | <i>deskripsi</i> | Text | <i>None</i> | Deskripsi wisata |
| 8 | <i>alamat</i> | Text | <i>None</i> | Alamat wisata |
| 9 | <i>jam_buka</i> | Varchar(30) | <i>None</i> | Jam buka wisata |
| 10 | <i>url</i> | Text | <i>None</i> | <i>url</i> dari wisata |

3. Tabel Kategori Wisata

Tabel kategori wisata digunakan untuk menyimpan dan mengakses data kategori wisata. Tabel wisata terdiri dari dua atribut yaitu *id_kategori_wisata* dan *nama_kategori_wisata*. *Id_kategori_wisata* merupakan *primary key* dari tabel kategori. Struktur tabel kategori wisata dapat dilihat pada tabel 3.63.

Tabel 3.63. Struktur Tabel Kategori Wisata

| No | Nama Fields | Tipe Data | Constraint | Keterangan |
|----|-----------------------------|-------------|------------------------------------|----------------------|
| 1 | <i>id_kategori_wisata</i> | Int(11) | <i>Primary Key, Auto_Increment</i> | Id kategori wisata |
| 2 | <i>nama_kategori_wisata</i> | Varchar(25) | <i>None</i> | Nama kategori wisata |

4. Tabel *Training*

Tabel *training* digunakan untuk menyimpan dan mengakses data *training* atau data latih. Tabel *training* terdiri dari empat atribut yaitu *id_training*, *id_wisata*, ulasan, dan label. *Id_training* merupakan *primary key* dari tabel *training* sedangkan, *id_wisata* merupakan *foreign key* dari tabel *training*. Struktur tabel *training* dapat dilihat pada tabel 3.64.

Tabel 3.64. Struktur Tabel Training

| No | Nama Fields | Tipe Data | Constraint | Keterangan |
|----|-------------|----------------------------|-----------------------------|----------------------|
| 1 | id_training | Int(11) | Primary_Key, Auto_Increment | Id data training |
| 2 | id_wisata | Int(11) | Foreign Key | Id wisata |
| 3 | ulasan | Text | None | Ulasan data training |
| 4 | label | Enum('positif', 'negatif') | None | Label data training |

5. Tabel Sentimen

Tabel sentimen digunakan untuk menyimpan dan mengakses data sentimen atau data hasil klasifikasi sentimen svm dengan *chi square* dan svm dengan *mutual information*. Tabel sentimen terdiri dari enam atribut yaitu id_sentimen, id_wisata, ulasan, *stemming*, *svm_chisquare* dan *svm_mi*. Id_sentimen merupakan *primary key* dari tabel sentimen sedangkan, id_wisata merupakan *foreign key* dari tabel sentimen. Struktur tabel sentimen dapat dilihat pada tabel 3.65.

Tabel 3.65. Struktur Tabel Sentimen

| No | Nama Fields | Tipe Data | Constraint | Keterangan |
|----|---------------|-----------|-----------------------------|-----------------------------------------------------------------|
| 1 | id_sentimen | Int(11) | Primary_Key, Auto_Increment | Id data sentimen |
| 2 | id_wisata | Int(11) | Foreign Key | Id wisata |
| 3 | ulasan | Text | None | Ulasan data sentimen |
| 4 | stemming | Text | None | Hasil <i>text preprocessing</i> |
| 5 | svm_chisquare | Int(11) | None | Hasil klasifikasi sentimen svm dengan <i>chisquare</i> |
| 6 | svm_mi | Int(11) | None | Hasil klasifikasi sentimen svm dengan <i>mutual information</i> |

6. Tabel Analisis Ulasan

Tabel analisis ulasan digunakan untuk menyimpan dan mengakses data ulasan yang akan dianalisis dan hasil klasifikasinya (klasifikasi sentimen svm dengan *chi square* dan svm dengan *mutual information*). Tabel analisis ulasan terdiri dari tiga belas atribut yaitu id_analisis, ulasan, *case_folding*, *remove_punct*, *remove_number*, *remove_repeat_char*, *spell_normal*, *tokenizing*, *filtering*, *stemming*, *preprocessing*, *svm_chisquare* dan *svm_mi*. Id_analisis merupakan *primary key* dari tabel analisis ulasan. Struktur tabel analisis ulasan dapat dilihat pada tabel 3.66.

Tabel 3.66. Struktur Tabel Analisis Ulasan

| No | Nama Fields | Tipe Data | Constraint | Keterangan |
|----|--------------------|-----------|-----------------------------|-------------------------------------|
| 1 | id_analisis | Int(11) | Primary Key, Auto_Increment | Id dari ulasan yang di analisis |
| 2 | ulasan | Text | None | Ulasan yang dianalisis |
| 3 | case_folding | Text | None | Hasil <i>case folding</i> |
| 4 | remove_punct | Text | None | Hasil <i>remove punctuation</i> |
| 5 | remove_number | Text | None | Hasil <i>remove number</i> |
| 6 | remove_repeat_char | Text | None | Hasil <i>remove repeated char</i> |
| 7 | spell_normal | Text | None | Hasil <i>spelling normalization</i> |
| 8 | tokenizing | Text | None | Hasil <i>tokenizing</i> |
| 9 | filtering | Text | None | Hasil <i>filtering</i> |
| 10 | Stemming | Text | None | Hasil <i>stemming</i> |

Tabel 3.67. Lanjutan Struktur Tabel Analisis Ulasan

| No | Nama Fields | Tipe Data | Constraint | Keterangan |
|----|---------------|-----------|------------|-----------------------------------------------------------------|
| 11 | preprocessing | Text | None | Hasil <i>text preprocessing</i> |
| 12 | svm_chisquare | Int(11) | None | Hasil klasifikasi sentimen svm dengan <i>chisquare</i> |
| 13 | svm_mi | Int(11) | None | Hasil klasifikasi sentimen svm dengan <i>mutual information</i> |

7. Tabel *Chisquare*

Tabel *chisquare* digunakan untuk menyimpan dan mengakses data hasil pembobotan fitur dari perhitungan *chi square*. Tabel *chisquare* terdiri dari empat atribut yaitu *id_chisquare*, *nama_fitur*, *bobot_chisquare*, dan kode. *Id_chisquare* merupakan *primary key* dari tabel *chisquare*. Struktur tabel *chisquare* dapat dilihat pada tabel 3.68

Tabel 3.68. Struktur Tabel *Chisquare*

| No | Nama Fields | Tipe Data | Constraint | Keterangan |
|----|------------------------|-----------|------------------------------------|------------------------------------------------------|
| 1 | <i>id_chisquare</i> | Int(11) | <i>Primary_Key, Auto_Increment</i> | Id fitur dari seleksi fitur <i>chi square</i> |
| 2 | <i>nama_fitur</i> | Text | None | Nama fitur |
| 3 | <i>bobot_chisquare</i> | Float | None | Hasil bobot fitur dari perhitungan <i>chi square</i> |
| 4 | kode | Int(11) | None | Kode <i>k-fold</i> dari fitur |

8. Tabel MI

Tabel MI digunakan untuk menyimpan dan mengakses data hasil pembobotan fitur dari perhitungan *mutual information*. Tabel *mutual informatin* terdiri dari empat atribut yaitu *id_mi*, *nama_fitur*, *bobot_mi*, dan kode. *Id_mi* merupakan *primary key* dari tabel MI. Struktur tabel MI dapat dilihat pada tabel 3.69

Tabel 3.69. Struktur Tabel MI

| No | Nama Fields | Tipe Data | Constraint | Keterangan |
|----|-------------------|-----------|------------------------------------|--------------------------------------------------------------|
| 1 | <i>id_mi</i> | Int(11) | <i>Primary_Key, Auto_Increment</i> | Id fitur dari seleksi fitur <i>mutual information</i> |
| 2 | <i>nama_fitur</i> | Text | None | Nama fitur |
| 3 | <i>bobot_mi</i> | Float | None | Hasil bobot fitur dari perhitungan <i>mutual information</i> |
| 4 | kode | Int(11) | None | Kode <i>k-fold</i> dari fitur |

9. Tabel Pengujian

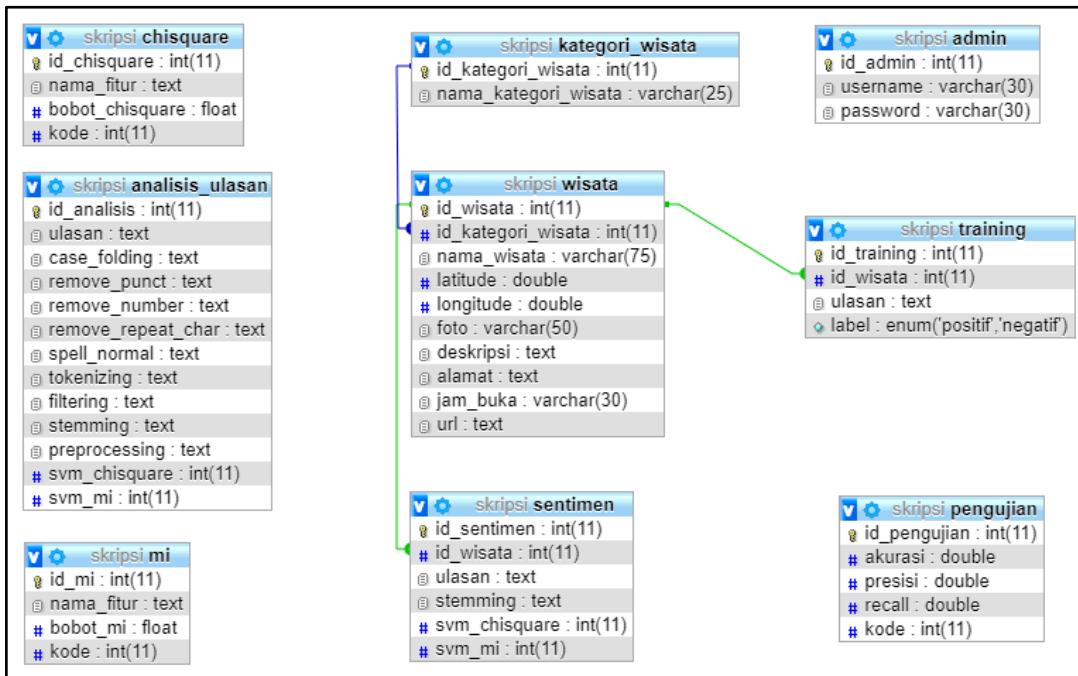
Tabel pengujian digunakan untuk menyimpan dan mengakses data hasil pengujian seperti akurasi, presisi, dan *recall*. Tabel pengujian terdiri dari lima atribut yaitu *id_pengujian*, akurasi, presisi, *recall* dan kode. *Id_pengujian* merupakan *primary key* dari tabel pengujian. Struktur tabel pengujian dapat dilihat pada tabel 3.70

Tabel 3.70. Struktur Tabel Pengujian

| No | Nama Fields | Tipe Data | Constraint | Keterangan |
|----|---------------------|-----------|------------------------------------|----------------------------------------------------------------------------------------------------------|
| 1 | <i>id_pengujian</i> | Int(11) | <i>Primary_Key, Auto_Increment</i> | Id pengujian |
| 2 | Akurasi | Double | None | Hasil akurasi pengujian |
| 3 | presisi | Double | None | Hasil presisi pengujian |
| 4 | <i>recall</i> | Double | None | Hasil <i>recall</i> pengujian |
| 5 | kode | Int(11) | None | Kode pembeda antara hasil pengujian svm dengan <i>chisquare</i> dan svm dengan <i>mutual information</i> |

c) Relasi antar tabel (RAT)

Relasi antar tabel (RAT) merupakan hubungan antara tabel yang satu dengan tabel lainnya, yang dihubungkan melalui *primary key* dan *foreign key*. Relasi antar tabel bertujuan untuk menggabungkan informasi dari tabel yang berelasi. Berikut ini detail perancangan relasi antar tabel yang digunakan dalam penerapan analisis sentimen ulasan di *google maps* menggunakan metode *support vector machine* dengan seleksi fitur *mutual information* dan *chi square* untuk rekomendasi destinasi tempat pariwisata di Kota Yogyakarta yang dapat dilihat pada gambar 3.23.



Gambar 3.23. Relasi Antar Tabel

3.2.2.4 Perancangan antarmuka

Perancangan antarmuka atau *user interface* merupakan tampilan visual atau grafis dari sebuah sistem yang berfungsi untuk menjembatani komunikasi antara sistem dengan pengguna. Perancangan antarmuka ini bertujuan untuk memberikan kemudahan dalam menggunakan sistem dan *user experience* sehingga menarik bagi pengguna. Pada penelitian ini, terdapat dua pengguna antara lain aktor admin dan aktor pengguna (*user*).

Berikut ini merupakan beberapa perancangan antarmuka untuk aktor admin :

1. Perancangan Antarmuka Halaman *Login*

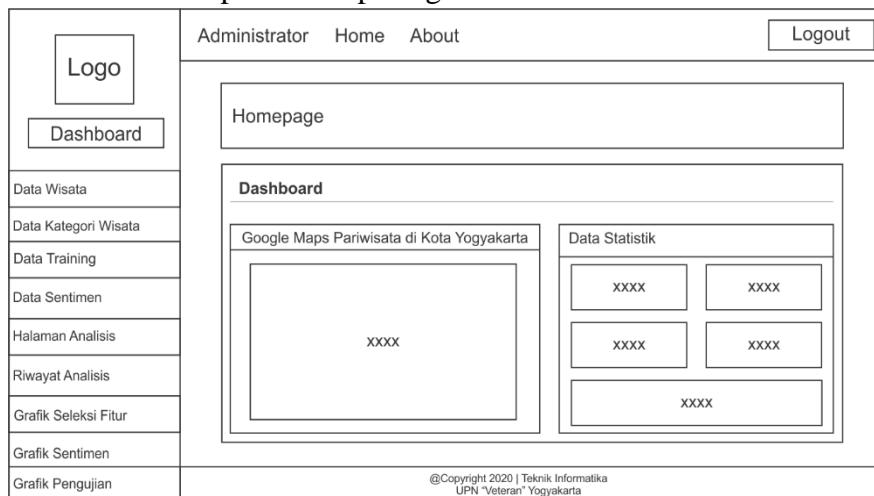
Halaman *login* merupakan halaman bagi aktor admin untuk masuk ke dalam sistem. Aktor admin memasukkan *username* dan *password* terlebih dahulu jika ingin masuk ke dalam sistem. Perancangan antarmuka pada halaman *login* dapat dilihat pada gambar 3.24.

Perancangan antarmuka halaman login yang menunjukkan struktur dasar dengan logo di bagian atas, tombol 'LOG IN' di tengah, dan dua input text untuk 'Username' dan 'Password' di bawahnya, serta tombol 'Login' di bagian bawah.

Gambar 3.24. Perancangan Antarmuka Halaman Login

2. Perancangan Antarmuka Halaman *Dashboard*

Halaman *dashboard* merupakan halaman yang pertama kali dilihat oleh aktor admin setelah melakukan proses *login*. Dalam halaman *dashboard* terdapat tampilan *google maps* daftar pariwisata di Kota Yogyakarta dan beberapa data statistik. Perancangan antarmuka pada halaman *dashboard* dapat dilihat pada gambar 3.25.



Gambar 3.25. Perancangan Antarmuka Halaman *Dashboard*

3. Perancangan Antarmuka Halaman *About*

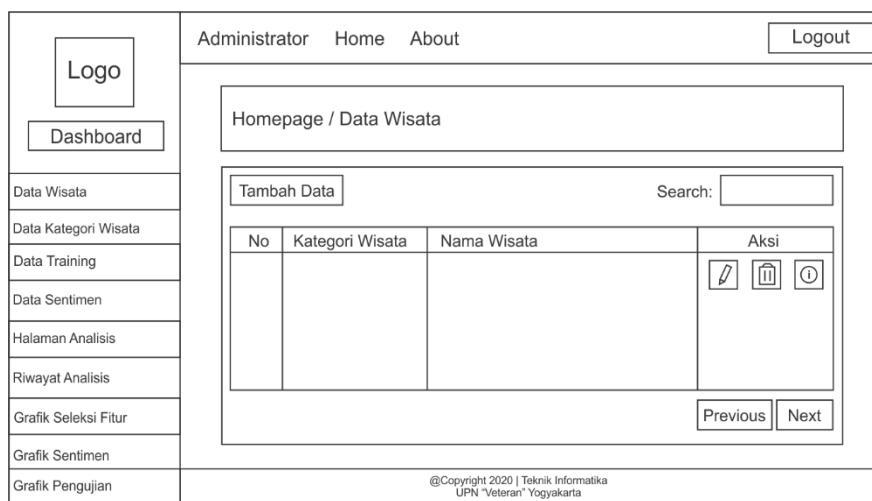
Halaman *about* merupakan halaman yang berisi keterangan mengenai isi dari menu-menu yang terdapat pada halaman admin. Perancangan antarmuka pada halaman *about* dapat dilihat pada gambar 3.26.



Gambar 3.26. Perancangan Antarmuka Halaman About

4. Perancangan Antarmuka Halaman Data Wisata

Halaman data wisata merupakan halaman yang berisi daftar data wisata yang ada di Kota Yogyakarta. Perancangan antarmuka pada halaman data wisata dapat dilihat pada gambar 3.27.



Gambar 3.27. Perancangan Antarmuka Halaman Data Wisata

Pada halaman data wisata terdapat tombol tambah data yang berfungsi menambah data wisata di Kota Yogyakarta. Aktor admin mengisi kategori wisata, nama wisata, *latitude*, *longitude*, foto, deskripsi, alamat, jam buka, dan url untuk wisata yang akan ditambahkan kemudian menekan tombol *save* untuk menyimpan data wisata baru atau *cancel* untuk membatalkan menyimpan data wisata baru. Perancangan antarmuka pada tombol tambah data wisata dapat dilihat pada gambar 3.28.

Tambah Data Wisata

Kategori Wisata :

Nama Wisata :

Latitude :

Longitude :

Foto : Pilih File Tidak ada file yang dipilih

Deskripsi :

Alamat :

Jam Buka :

URL :

Cancel Save

Gambar 3.28. Perancangan Antarmuka Tombol Tambah Data Wisata

Pada halaman data wisata terdapat beberapa pilihan menu aksi seperti detail, edit, dan hapus data wisata berdasarkan tombol yang di klik sesuai baris pada tabel. Tombol delete berfungsi untuk menghapus data wisata. Sedangkan, Tombol detail wisata berfungsi untuk menampilkan detail dari data wisata. Perancangan antarmuka pada tombol detail data wisata dapat dilihat pada gambar 3.29.

Detail Data Wisata

Nama Wisata :
xxxx

Kategori Wisata :
xxxx

Deskripsi :
xxxx

Jam Buka :
xxxx

Alamat :
xxxx

Latitude :
xxxx

Longitude :
xxxx

URL :
xxxx

Foto :
xxxx

Back

Gambar 3.29. Perancangan Antarmuka Tombol Detail Data Wisata

Pada tombol edit merupakan aksi untuk mengedit data wisata. Perancangan antarmuka pada tombol edit data wisata dapat dilihat pada gambar 3.30.

Logo

Administrator

Homepage

Tambah Data

| No | Kat |
|----|-----|
| | |

Edit Data Wisata

Kategori Wisata : xxxx

Nama Wisata : xxxx

Latitude : xxxx

Longitude : xxxx

Foto : Pilih File Tidak ada file yang dipilih

Deskripsi : xxxx

Alamat : xxxx

Jam Buka : xxxx

URL : xxxx

Aksi

Cancel Save

UPN "Veteran" Yogyakarta

Gambar 3.30. Perancangan Antarmuka Tombol Edit Data Wisata

5. Perancangan Antarmuka Halaman Data Kategori Wisata

Pada halaman data kategori wisata menampilkan daftar kategori wisata yang ada di Kota Yogyakarta. Perancangan antarmuka pada halaman data kategori wisata dapat dilihat pada gambar 3.31.

Logo

Administrator Home About Logout

Dashboard

Data Wisata

Data Kategori Wisata

Data Training

Data Sentimen

Halaman Analisis

Riwayat Analisis

Grafik Seleksi Fitur

Grafik Sentimen

Grafik Pengujian

Homepage / Data Kategori Wisata

Tambah Data Search:

| No | Nama Kategori Wisata | Aksi |
|----|----------------------|------|
| | | |

Previous Next

@Copyright 2020 | Teknik Informatika
UPN "Veteran" Yogyakarta

Gambar 3.31. Perancangan Antarmuka Halaman Data Kategori Wisata

Pada halaman data kategori wisata terdapat tombol tambah data yang berfungsi menambah data kategori wisata di Kota Yogyakarta. Aktor admin mengisi nama kategori wisata yang baru yang akan ditambahkan, kemudian menekan tombol *save* untuk menyimpan data kategori wisata baru atau *cancel* untuk membatalkan menyimpan data kategori wisata baru. Perancangan antarmuka pada tombol tambah data kategori wisata dapat dilihat pada gambar 3.32.

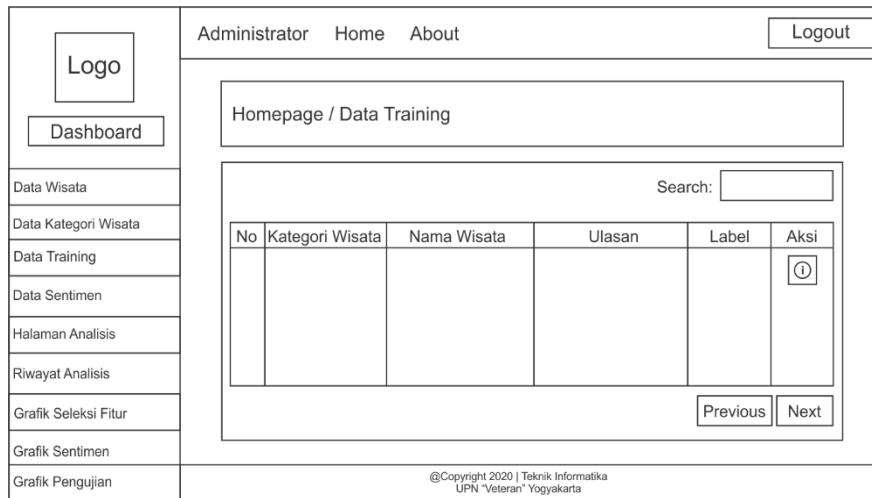
Gambar 3.32. Perancangan Antarmuka Tombol Tambah Data Kategori Wisata

Pada halaman ini juga terdapat tombol aksi hapus dan edit. Tombol aksi hapus berfungsi untuk menghapus kategori berdasarkan baris. Sedangkan tombol edit berfungsi untuk mengedit data kategori wisata berdasarkan baris. Perancangan antarmuka pada tombol edit data kategori wisata dapat dilihat pada gambar 3.33.

Gambar 3.33. Perancangan Antarmuka Tombol Edit Data Kategori Wisata

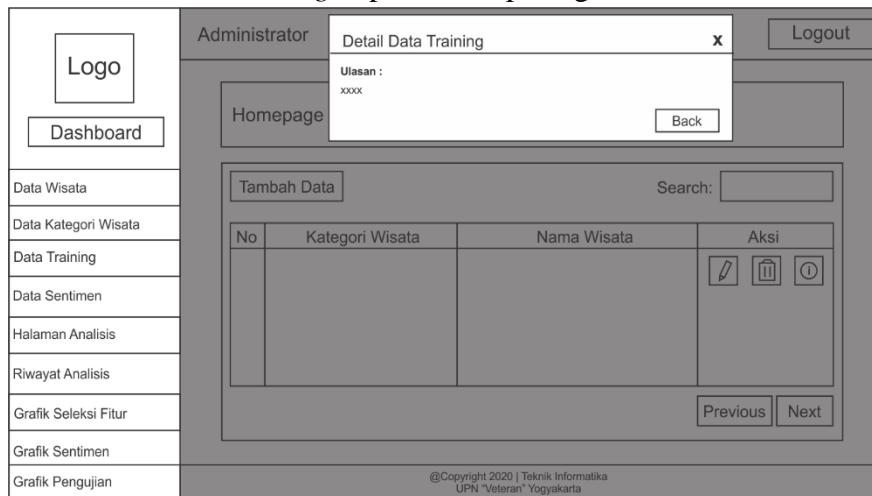
6. Perancangan Antarmuka Halaman Data *Training*

Pada halaman data *training* menampilkan daftar data *training* atau data latih yang digunakan untuk pembuatan atau pembentukan model. Perancangan antarmuka pada halaman data *training* dapat dilihat pada gambar 3.34.



Gambar 3.34. Perancangan Antarmuka Halaman Data *Training*

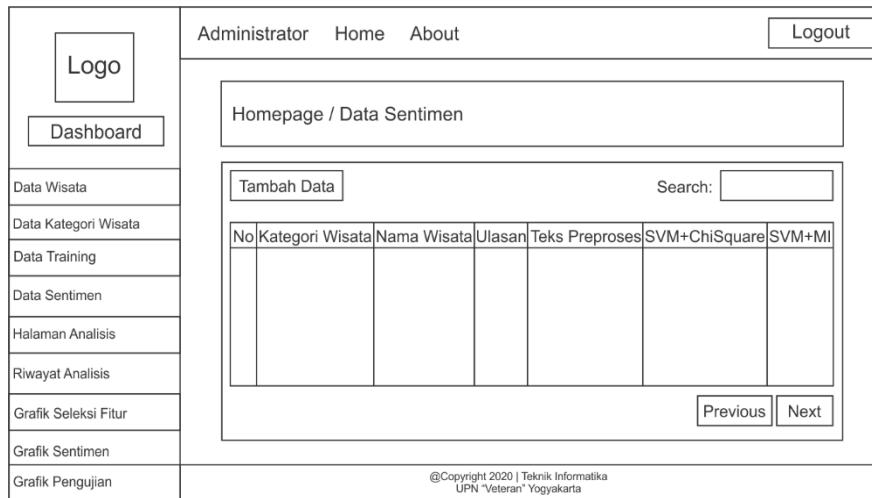
Pada halaman data *training* terdapat tombol aksi detail untuk menampilkan ulasan asli yang lebih lengkap atau detail sesuai dengan baris pada tombol. Perancangan antarmuka pada tombol aksi detail data *training* dapat dilihat pada gambar 3.35.



Gambar 3.35. Perancangan Antarmuka Tombol Detail Data *Training*

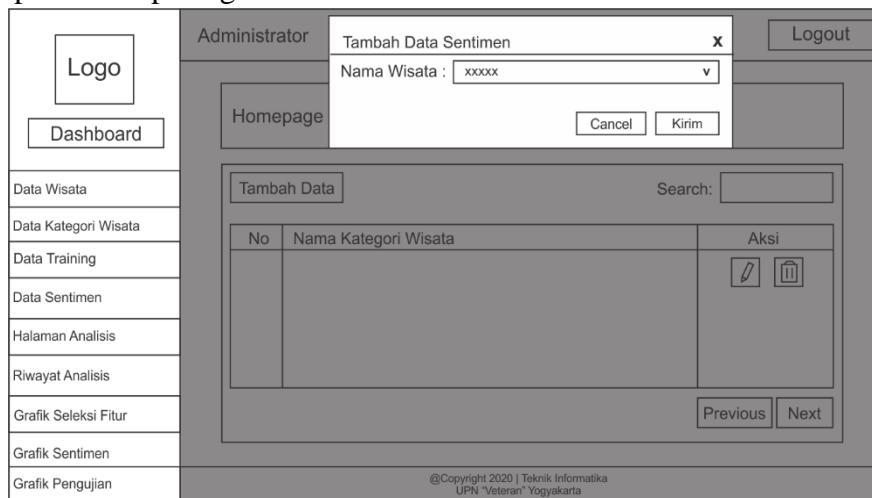
7. Perancangan Antarmuka Halaman Data Sentimen

Pada halaman data sentimen menampilkan daftar data latih yang juga menjadi data uji dan data uji baru hasil *scraping* yang sudah di analisis sentimen atau diklasifikasi (klasifikasi positif atau negatif) berdasarkan metode SVM dengan *Chi Square* dan SVM dengan *Mutual Information*. Perancangan antarmuka pada halaman data sentimen dapat dilihat pada gambar 3.36.



Gambar 3.36. Perancangan Antarmuka Halaman Data Sentimen

Pada halaman data sentimen terdapat tombol tambah data yang berfungsi menambah data sentimen berdasarkan nama wisata dengan cara *scraping* data ulasan di *google maps reviews*. Data yang di *scraping* tersebut kemudian akan menjadi data uji atau diklasifikasi ke dalam sentimen positif atau negatif. Perancangan antarmuka pada tombol tambah data sentimen dapat dilihat pada gambar 3.37.



Gambar 3.37. Perancangan Antarmuka Tombol Tambah Data Sentimen

8. Perancangan Antarmuka Halaman Analisis

Halaman analisis merupakan halaman untuk menguji atau mengklasifikasikan data dengan memasukkan ulasan. Ulasan yang telah dimasukkan akan menampilkan hasil dari proses *text preprocessing* dan hasil dari proses klasifikasi. Perancangan antarmuka pada halaman analisis dapat dilihat pada gambar 3.38.

Perancangan antarmuka halaman analisis yang menampilkan sidebar dengan menu seperti Logo, Dashboard, Data Wisata, Data Kategori Wisata, Data Training, Data Sentimen, Halaman Analisis, Riwayat Analisis, Grafik Seleksi Fitur, Grafik Sentimen, dan Grafik Pengujian. Di bagian atas, terdapat menu Administrator, Home, About, Logout, dan judul "Homepage / Halaman Analisis". Di bagian tengah, terdapat form "Ulasan" dengan input "Masukkan ulasan disini..." dan tombol "Preproses & Analisis". Berikutnya, terdapat daftar proses analisis: Ulasan, Case Folding, Remove Punctuation, Remove Number, Remove Repeated Characters, Spelling Normalization, Tokenizing, Filtering, Stemming, Hasil Sentimen SVM+ChiSquare, dan Hasil Sentimen SVM+MI. Di bagian bawah, terdapat kredensial ciptaan: "@Copyright 2020 | Teknik Informatika UPN "Veteran" Yogyakarta".

Gambar 3.38. Perancangan Antarmuka Halaman Analisis

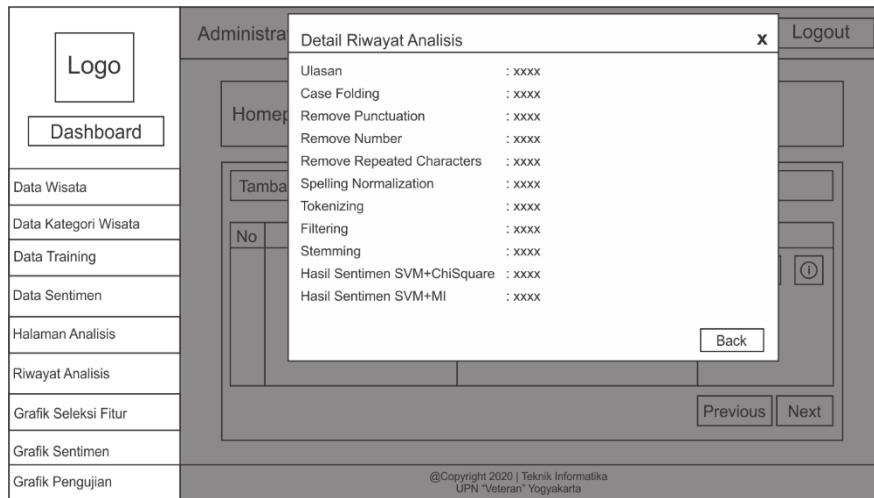
9. Perancangan Antarmuka Halaman Riwayat Analisis

Pada halaman riwayat analisis menampilkan daftar riwayat beberapa ulasan yang telah dianalisis atau diuji pada halaman analisis. Perancangan antarmuka pada halaman riwayat analisis dapat dilihat pada gambar 3.39.

Perancangan antarmuka halaman riwayat analisis yang menampilkan sidebar dengan menu Logo, Dashboard, Data Wisata, Data Kategori Wisata, Data Training, Data Sentimen, Halaman Analisis, Riwayat Analisis, Grafik Seleksi Fitur, Grafik Sentimen, dan Grafik Pengujian. Di bagian atas, terdapat menu Administrator, Home, About, Logout, dan judul "Homepage / Riwayat Analisis". Di bagian tengah, terdapat form "Search:" dan tabel riwayat analisis. Tabel ini memiliki kolom: No, Ulasan, Teks Preproses, SVM+ChiSquare, SVM+MI, dan Aksi. Di kolom Aksi, terdapat tombol detail (tanda info). Di bagian bawah, terdapat tombol "Previous" dan "Next". Di bagian bawah, terdapat kredensial ciptaan: "@Copyright 2020 | Teknik Informatika UPN "Veteran" Yogyakarta".

Gambar 3.39. Perancangan Antarmuka Halaman Riwayat Analisis

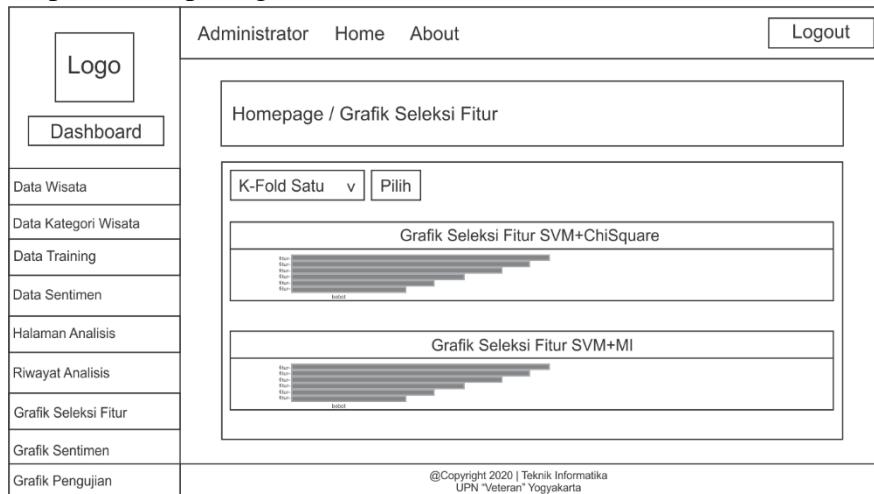
Pada halaman analisis ini juga terdapat tombol aksi detail untuk menampilkan detail dari hasil *text preprocessing* dan hasil klasifikasi pada ulasan sesuai dengan baris. Perancangan antarmuka pada tombol detail riwayat analisis dapat dilihat pada gambar 3.40.



Gambar 3.40. Perancangan Antarmuka Tombol Detail Riwayat Analisis

10. Perancangan Antarmuka Halaman Grafik Seleksi Fitur

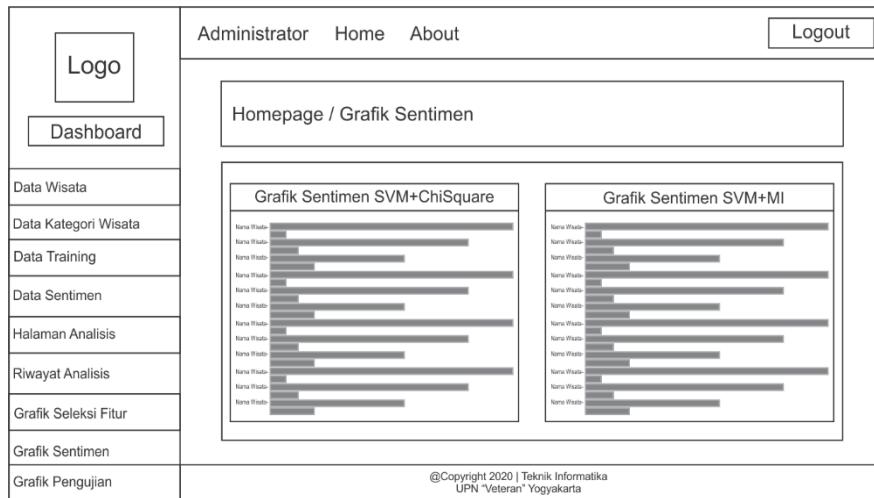
Pada halaman grafik seleksi fitur menampilkan bobot fitur atau kata berdasarkan perhitungan algoritma *chi square* dan *mutual information*. Daftar pembobotan fitur ini diurutkan berdasarkan bobot tertinggi ke terendah. Pada halaman ini hanya menampilkan 25 fitur teratas sesuai dengan *k-fold* yang dipilih. Perancangan antarmuka pada halaman grafik seleksi fitur dapat dilihat pada gambar 3.41.



Gambar 3.41. Perancangan Antarmuka Halaman Grafik Seleksi Fitur

11. Perancangan Antarmuka Halaman Grafik Sentimen

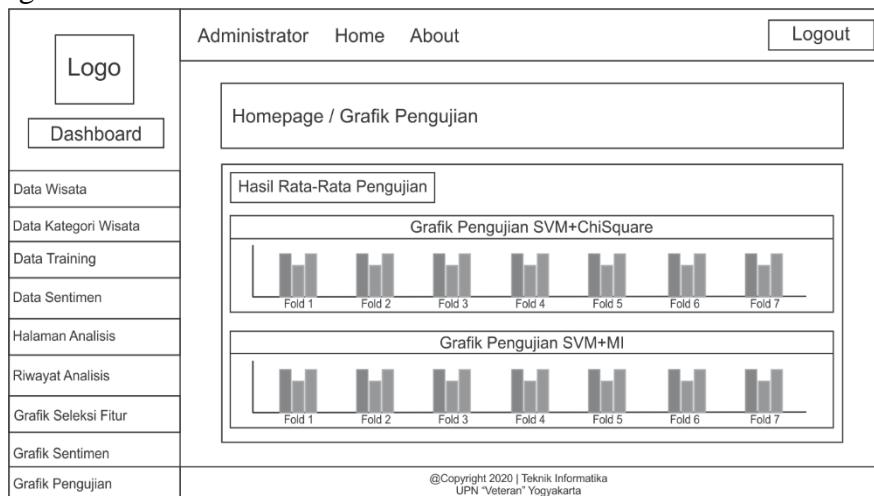
Pada halaman grafik sentimen menampilkan jumlah klasifikasi pada data sentimen positif dan negatif berdasarkan wisatanya. Pada halaman ini terdapat dua grafik sentimen SVM dengan *Chi Square* dan SVM dengan *Mutual Information*. Perancangan antarmuka pada halaman grafik sentimen dapat dilihat pada gambar 3.42.



Gambar 3.42. Perancangan Antarmuka Halaman Grafik Sentimen

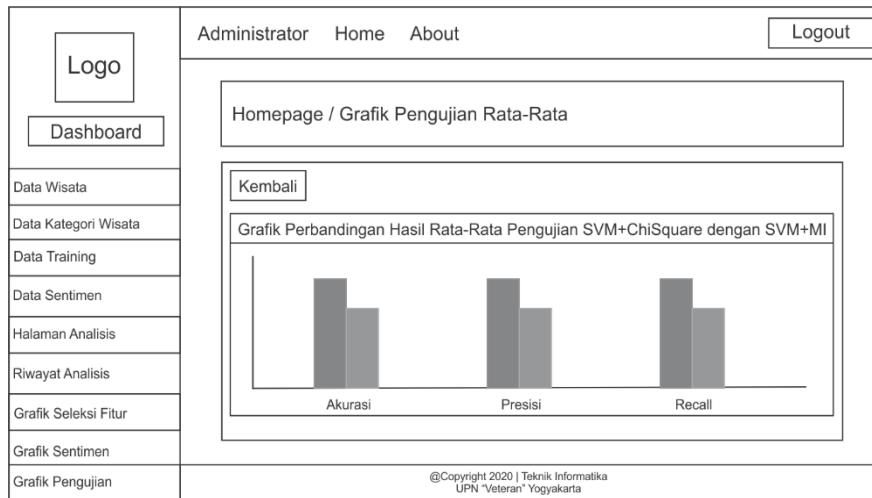
12. Perancangan Antarmuka Halaman Grafik Pengujian

Pada halaman grafik pengujian menampilkan grafik pengujian akurasi, presisi, *recall* metode SVM dengan *Chi Square* dan metode SVM dengan *Mutual Information* berdasarkan *k-fold cross validation*-nya. Perancangan antarmuka pada halaman grafik pengujian dapat dilihat pada gambar 3.43.



Gambar 3.43. Perancangan Antarmuka Halaman Grafik Pengujian

Pada halaman grafik pengujian juga terdapat tombol hasil rata-rata pengujian, tombol ini menampilkan hasil rata-rata dari akurasi, presisi, *recall* pada metode SVM dengan *Chi Square* dan metode SVM dengan *Mutual Information*. Perancangan antarmuka pada halaman grafik pengujian rata-rata dapat dilihat pada gambar 3.44.



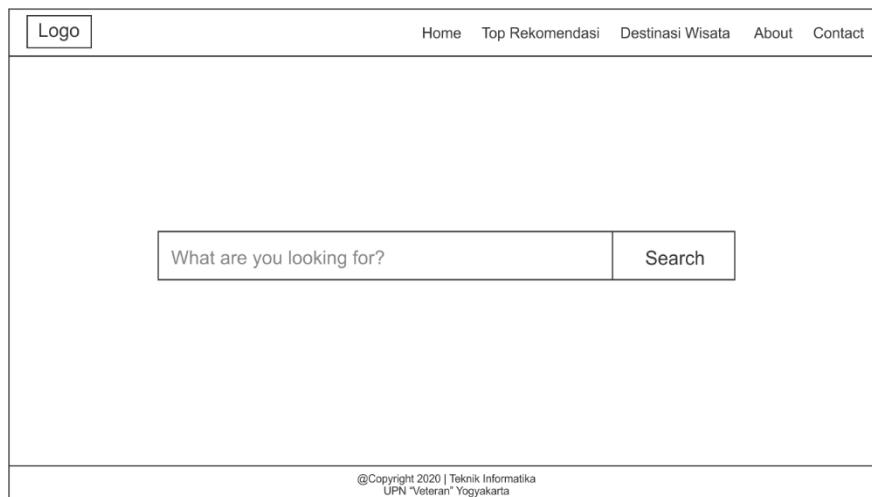
Gambar 3.44. Perancangan Antarmuka Halaman Grafik Pengujian Rata-Rata

Tombol kembali pada halaman grafik pengujian rata-rata berfungsi untuk kembali ke halaman pengujian atau halaman sebelumnya.

Berikut ini merupakan beberapa perancangan antarmuka untuk aktor pengguna atau *user* :

1. Perancangan Antarmuka Halaman *Home*

Pada halaman *home* terdapat tombol *search* atau pencarian yang berfungsi untuk menampilkan halaman top rekomendasi berdasarkan kata atau *query* yang diketikkan oleh aktor pengguna. Perancangan antarmuka pada halaman *home* dapat dilihat pada gambar 3.45.



Gambar 3.45. Perancangan Antarmuka Halaman *Home*

2. Perancangan Antarmuka Halaman Top Rekomendasi

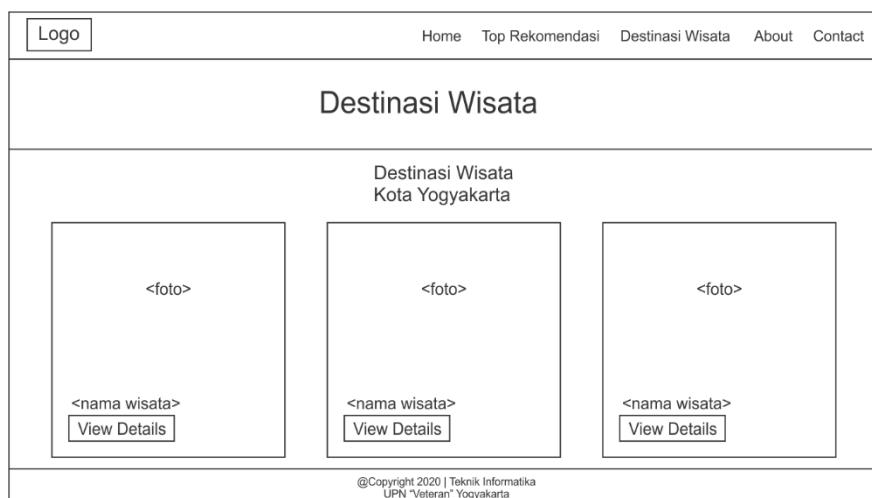
Pada halaman top rekomendasi menampilkan daftar top rekomendasi untuk semua kategori atau berdasarkan kategori museum, pendidikan, belanja, sejarah dan budaya. Perancangan antarmuka pada halaman top rekomendasi dapat dilihat pada gambar 3.46.



Gambar 3.46 Perancangan Antarmuka Halaman Top Rekomendasi

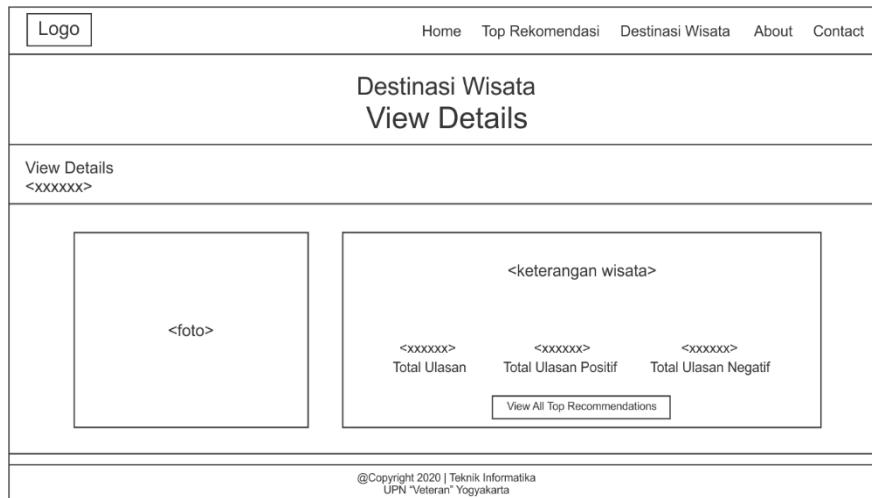
3. Perancangan Antarmuka Halaman Destinasi Wisata

Pada halaman destinasi wisata menampilkan daftar wisata yang ada di Kota Yogyakarta. Perancangan antarmuka pada halaman destinasi wisata dapat dilihat pada gambar 3.47.



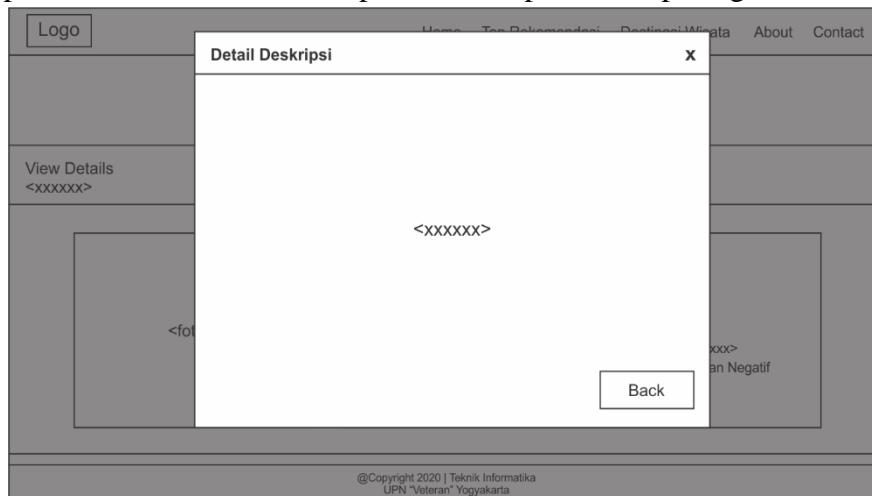
Gambar 3.47. Perancangan Antarmuka Halaman Destinasi Wisata

Pada halaman ini terdapat tombol *view details* yang jika aktor pengguna mengklik maka akan menuju halaman detail wisata yang diklik. Pada halaman ini menampilkan informasi singkat mengenai wisata, informasi jumlah ulasan, jumlah ulasan positif, dan jumlah ulasan negatif. Pada halaman detail wisata ini terdapat tombol untuk menuju halaman top rekomendasi untuk semua kategori wisata. Perancangan antarmuka pada halaman *view details* wisata dapat dilihat pada gambar 3.48.



Gambar 3.48. Perancangan Antarmuka Halaman *View Details* Wisata

Pada halaman *view details* wisata ini juga terdapat *link more* yang jika aktor pengguna mengklik akan menampilkan detail deskripsi wisata tersebut. Perancangan antarmuka pada *link more* detail deskripsi wisata dapat dilihat pada gambar 3.49.



Gambar 3.49. Perancangan Antarmuka *Link Detail Deskripsi* Wisata

4. Perancangan Antarmuka Halaman *About*

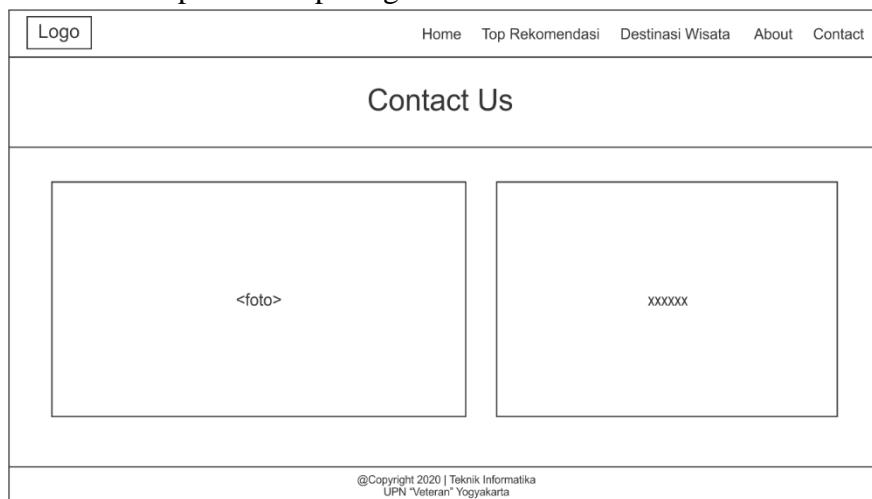
Pada halaman *about* menampilkan penjelasan singkat mengenai sistem ini. Perancangan antarmuka pada halaman *about* dapat dilihat pada gambar 3.50.



Gambar 3.50. Perancangan Antarmuka Halaman *About*

5. Perancangan Antarmuka Halaman *Contact*

Pada halaman *contact* menampilkan kontak yang dapat dihubungi jika aktor pengguna ingin memberikan kritik dan saran mengenai sistem ini. Perancangan antarmuka pada halaman *contact* dapat dilihat pada gambar 3.51.



Gambar 3.51. Perancangan Antarmuka Halaman *Contact*

3.2.3 Pengujian sistem

Pengujian sistem yang digunakan pada penelitian ini menggunakan pengujian *black box*. Pengujian *black box* merupakan pengujian yang hanya menguji atau menilai kegunaan dari sistem, tanpa mengetahui proses detail pada program yang ada didalamnya. Berikut ini merupakan pengujian *black box* yang dapat dilihat pada tabel 3.3

Tabel 3.71. Pengujian *Black Box*

| Aktor | Halaman | Detail Pengujian | Pengujian | |
|-------|-------------|---------------------------------------------------|-----------|-------|
| | | | Berhasil | Gagal |
| Admin | Login | Melakukan proses login Fungsi session | | |
| | Dashboard | Menampilkan google maps wisata di kota yogyakarta | | |
| | | Menampilkan data statistik | | |
| | About | Menampilkan informasi mengenai sistem | | |
| | Data Wisata | Menampilkan data wisata | | |

Tabel 3.72. Lanjutan Pengujian Black Box

| Aktor | Halaman | Detail Pengujian | Pengujian | |
|-------|----------------------------|--------------------------------------------------------------|-----------|-------|
| | | | Berhasil | Gagal |
| Admin | Data Wisata | Menambah data wisata | | |
| | | Mengedit data wisata | | |
| | | Menghapus data wisata | | |
| | | Menampilkan detail data wisata | | |
| | | Fungsi pencarian dan pengurutan | | |
| | Data Kategori Wisata | Menampilkan data kategori wisata | | |
| | | Menambah data kategori wisata | | |
| | | Mengedit data kategori wisata | | |
| | | Menghapus data kategori wisata | | |
| | Data Training | Fungsi pencarian dan pengurutan | | |
| | | Menampilkan data training | | |
| | | Menampilkan detail data training | | |
| User | Data Sentimen | Fungsi pencarian dan pengurutan | | |
| | | Menampilkan data sentimen | | |
| | | Menambah data sentimen | | |
| | Halaman Analisis | Menampilkan hasil proses klasifikasi | | |
| | | Melakukan proses klasifikasi | | |
| | Riwayat Analisis | Menampilkan hasil riwayat analisis | | |
| | | Menampilkan detail dari hasil analisis | | |
| | Grafik Seleksi Fitur | Menampilkan grafik data seleksi fitur | | |
| | Grafik Sentimen | Menampilkan grafik data sentimen | | |
| | Grafik Pengujian | Menampilkan grafik data pengujian | | |
| | Grafik Pengujian Rata-Rata | Menampilkan grafik data pengujian rata-rata | | |
| User | Home | Menampilkan daftar menu | | |
| | | Fungsi pencarian | | |
| | Top Rekomendasi | Menampilkan daftar top rekomendasi wisata di kota yogyakarta | | |
| | Destinasi Wisata | Menampilkan daftar destinasi wisata di kota yogyakarta | | |
| | View Details | Menampilkan informasi detail data wisata | | |
| | About | Menampilkan informasi sistem | | |
| | Contact | Menampilkan informasi kontak | | |

BAB IV

HASIL, PENGUJIAN DAN PEMBAHASAN

Pada bab ini akan menjelaskan mengenai hasil, pengujian dan pembahasan dari implementasi perancangan sistem analisis sentimen pada bab sebelumnya. Hasil dari implementasi perancangan analisis sentimen kemudian akan melalui tahap pengujian model dan pengujian sistem. Pada pengujian model akan menggunakan *confusion matrix* dan *k-fold cross validation*, sedangkan pada pengujian sistem akan menggunakan *black box*. Setelah melalui tahap pengujian maka akan diperoleh pembahasan mengenai apakah sistem yang telah dibuat sudah menjawab rumusan masalah pada penelitian ini.

4.1. Hasil Penelitian

Pada bagian ini akan membahas mengenai hasil implementasi analisis sentimen ulasan di *google maps* dan implementasi sistem yang telah dibuat berdasarkan rancangan pada bab sebelumnya.

4.1.1 Implementasi analisis sentimen ulasan di *google maps*

Implementasi analisis sentimen ulasan di *google maps* membahas tentang hasil penelitian yang berkaitan dengan metode yang digunakan berdasarkan metodologi penelitian yang telah dibuat.

4.1.1.1 Modul *web scraping*

Modul *web scraping* merupakan modul untuk mengumpulkan data ulasan tempat pariwisata yang ada di Kota Yogyakarta yang berada di *google maps* dengan menggunakan teknik *web scraping*. Modul *web scraping* pada penelitian ini menggunakan bahasa pemrograman *python* dengan *library selenium*. Proses *web scraping* dilakukan dengan memasukkan *url* dari tempat pariwisata yang akan diambil data ulasannya. *Library selenium* ini kemudian akan memanggil *chromedriver* untuk menampilkan hasil *url* dari tempat pariwisata tersebut, dan mencari *tag html section-review-content* untuk mengambil isi dari tag tersebut. Setelah semua data ulasan telah diambil maka akan dimasukkan ke dalam *file* dengan format csv. Berikut ini merupakan modul *web scraping* yang dapat dilihat pada Algoritma 1.

Algoritma 1: Web Scraping

```
1  require: selenium: module scraping
2  require: url_wisata: input url wisata
3  require: id, user, komentar: output
4
5  driver <- webdriver.Chrome(executable_path="..chromedriver.exe")
6  driver.get("url_wisata")
7  reviews <- driver.find_elements_by_xpath("//button[@class='widget-pane-
8          link']")
9  driver.execute_script(
10     'var d <- document.getElementsByClassName("section-layout section-scroll
11       box scrollable-y scrollable-show") [0];'
12     'var comment <- document.getElementsByClassName("section-review-review
13          content");'+
14     'async function scrolldown() {
15       while(comment.length < 10) do
16         d.scrollTop += 10;await sleep(100)
17       endwhile
18     +'scrolldown()
19   )
20  comment_text <- driver.find_elements_by_xpath("../div[@class='section-
21          review-content']")
```

Algoritma 2: Lanjutan Web Scraping

```
22 function writeTofile(data):
23     with open('datawisata_NamaWisata.csv', mode='ab') as csv_file:
24         fieldnames =['id', 'user','komentar']
25         writer <- csv.DictWriter(csv_file,fieldnames=fieldnames,
26                                 lineterminator='\n')
27         writer.writerow(data)
28 endfunction
29
30 for komen in comment_text do
31     section <- komen.find_element_by_xpath("./div[@class='section-review-
32             title']")
33     id_r <- section.find_element_by_xpath("./span").text
34     komentar <- komen.find_element_by_xpath("./span[@class='section-review-
35             -text']").text
36     data <- {'id':id_wisata,'user':id_r,'komentar': komentar}
37     output data
38     writeTofile(data)
39 endfor
```

4.1.1.2 Modul text preprocessing

Pada bagian ini akan menjelaskan tentang implementasi dari tahapan *text preprocessing*. *Text preprocessing* merupakan tahapan mengubah data tidak terstruktur maupun semi terstruktur menjadi data struktur sehingga data siap digunakan untuk proses analisis sentimen. Pada tahap *text preprocessing* data yang digunakan merupakan hasil dari data yang sudah diberi label secara manual. Berikut ini merupakan modul-modul yang digunakan pada tahap *text preprocessing*.

a) Modul Case Folding

Modul *case folding* merupakan modul untuk mengubah karakter yang terdapat pada data ulasan menjadi huruf kecil atau *lowercase*. Modul *case folding* menggunakan fungsi *lower()* yang terdapat pada *library nltk* di bahasa pemrograman *python*. Berikut ini merupakan modul *case folding* yang terdapat pada Algoritma 3.

Algoritma 3: Case Folding

```
1 require: nltk: module untuk pengolahan simbol dan bahasa alami
2 require: data: data array ulasan di google maps
3
4 function case_folding(tokens):
5     return tokens.lower() //mengubah menjadi huruf kecil
6 endfunction
7
8 data['Text_casefolding'] <- data['komen'].apply(lambda x: case_folding(x))
9 hasil_casefolding <- data['Text_casefolding']
10 output "casefolding"
11 output hasil_casefolding
//mengubah hasil case folding menjadi string
12 string str_hasil_casefolding <- str(hasil_casefolding)
```

b) Modul Remove Punctuation

Modul *remove punctuation* merupakan modul untuk menghapus karakter simbol dan tanda baca yang terdapat pada data ulasan. Modul *remove punctuation* menggunakan *library re* yang terdapat pada bahasa pemrograman *python*. Berikut ini merupakan modul *remove punctuation* yang terdapat pada Algoritma 4.

Algoritma 4: Remove Punctuation

```
1 require: data: data array ulasan di google maps hasil case folding
2 require: re: module untuk menghapus tanda baca dan simbol
3
4 function remove_punctuation(text):
5     text_nopunct <- ''
```

Algoritma 5: Lanjutan Remove Punctuation

```
6     text_nopunct <- re.sub('['+string.punctuation+']', ' ', text)
7     return text_nopunct
8 endfunction
9
10 data['Text_NoPuct'] <- data['Text_casefolding'].apply(lambda x:
11             remove_punctuation(x))
12 hasil_textnopunct <- data['Text_NoPuct']
13 output "Hapus Tanda Baca"
14 output hasil_textnopunct
15 //mengubah hasil remove punctuation menjadi string
16 string str_hasil_textnopunct <- str(hasil_textnopunct)
```

c) Modul Remove Number

Modul *remove number* merupakan modul untuk menghapus karakter angka yang terdapat pada data ulasan. Modul *remove number* menggunakan *library re* yang terdapat pada bahasa pemrograman *python*. Berikut ini merupakan modul *remove number* yang terdapat pada Algoritma 6.

Algoritma 6: Remove Number

```
1 require: data: data array ulasan di google maps hasil remove punctuation
2 require: re: module untuk menghapus angka
3
4 function remove_number(text):
5     text_nonum <- ''
6     text_nonum <- re.sub(r'\d+', ' ', text)
7     return text_nonum
8 endfunction
9
10 data['Text_Nonum'] <- data['Text_NoPuct'].apply(lambda x: remove_number(x))
11 hasil_nonum <- data['Text_Nonum']
12 output "Hapus Angka"
13 output hasil_nonum
14 //mengubah hasil remove number menjadi string
15 string str_hasil_nonum <- str(hasil_nonum)
```

d) Modul Remove Repeated Characters

Modul *remove repeated characters* merupakan modul untuk menghapus karakter yang memiliki dua atau lebih pengulangan karakter yang terdapat pada data ulasan. Modul *remove repeated characters* menggunakan *library re* yang terdapat pada bahasa pemrograman *python*. Berikut ini merupakan modul *remove number* yang terdapat pada Algoritma 7.

Algoritma 7: Remove Repeated Characters

```
1 require: data: data array ulasan di google maps hasil remove number
2 require: re: module untuk menghapus karakter yg memiliki 2 atau lebih
3 pengulangan karakter
4
5 function remove_repeated_characters(text):
6     pattern <- re.compile(r"(.)\1{1,}", re.DOTALL)
7     return pattern.sub(r"\1\1", text)
8 endfunction
9
10 data['text_doble'] <- data['Text_Nonum'].apply(lambda x: remove_repeated_
11         characters(x))
12 hasil_hurufdobel <- data['text_doble']
13 output "Hapus huruf Doble"
14 output hasil_hurufdobel
15 //mengubah hasil remove repeated characters menjadi string
16 string hasil_hurufdobel <- str(hasil_hurufdobel)
```

e) Modul Spelling Normalization

Modul *spelling normalization* merupakan modul untuk mengubah *term* atau kata yang tidak baku pada data ulasan menjadi kata baku berdasarkan kata yang terdapat pada file *spelling_normalization.txt*. Daftar kamus *spelling normalization* yang digunakan merupakan penelitian dari (Noehilasari, 2014) yang ditambah secara manual dari data yang digunakan. Berikut ini merupakan modul *spelling normalization* yang terdapat pada Algoritma 8.

Algoritma 8: Spelling Normalization

```
1  require: data: data array ulasan di google maps hasil remove repeated
2  characters
3  require: spelling_normalization: data array kamus spelling normalization
4
5  function open_kamus_prepro(x):
6      kamus <- {}
7      with open(x, 'r') as file :
8          for line in file do
9              slang <- line.replace('\"\", \"").split(':')
10             kamus[slang[0].strip()] <- slang[1].rstrip('\n').lstrip()
11     endfor
12     return kamus
13 endfunction
14
15 kamus_slang <- open_kamus_prepro('spelling_normalization.txt')
16
17 function spelling_normalization(text):
18     sentence_list <- text.split()
19     new_sentence <- []
20     for word in sentence_list do
21         for candidate_replacement in kamus_slang do
22             if candidate_replacement = word then
23                 word <- word.replace(candidate_replacement, kamus_slang
24                               [candidate_replacement])
25             Endif
26     endfor
27     new_sentence.append(word)
28   endfor
29   return " ".join(new_sentence)
30 endfunction
31
32 data['Text_slang'] <- data['text_doble'].apply(lambda x:
33                                         spelling_normalization(x))
34 hasil_slang <- data['Text_slang']
35 output "Spelling Normalization"
36 output hasil_slang
37 //mengubah hasil spelling normalization menjadi string
38 string str_hasil_slang <- str(hasil_slang)
```

f) Modul Tokenizing

Modul *tokenizing* merupakan modul untuk memecah atau memisahkan *string* pada data ulasan berdasarkan kata yang menyusunnya sehingga menghasilkan token. Modul *tokenizing* menggunakan fungsi *word_tokenize()* yang terdapat pada *library nltk* di bahasa pemrograman *python*. Berikut ini merupakan modul *tokenizing* yang terdapat pada Algoritma 9.

Algoritma 9: Tokenizing

```
1  require: data: data array ulasan di google maps hasil spelling
2  normalization
3  require: nltk: module untuk memecah string berdasarkan kata
4
5  tokens <- [word_tokenize(sen) for sen in data.Text_slang]
6  hasil_token <- tokens
```

Algoritma 10: Lanjutan Tokenizing

```
7  output "Tokenizing"
8  output hasil_token
9  //mengubah hasil tokenizing menjadi string
10 string str_hasil_token <- str(hasil_token)
```

g) Modul Filtering

Modul *filtering* merupakan modul untuk menghapus *term* atau kata dari hasil *tokenizing* yang dianggap tidak penting berdasarkan daftar kamus yang terdapat pada *file stopword.txt*. Daftar kamus *stopword* yang digunakan merupakan penelitian dari (Tala, 2003). Berikut ini merupakan modul *filtering* yang terdapat pada Algoritma 11.

Algoritma 11: Filtering

```
1  require: data: data array ulasan di google maps hasil tokenizing
2  require: stopword: data array kamus stopword
3
4  kamus_stopword <- []
5  with open('stopword.txt','r') as file :
6      for line in file do
7          slang <- line.replace("""","").strip()
8          kamus_stopword.append(slang)
9      endfor
10 //mengecek setiap kata atau token apakah terdapat pada kamus stopword
11 function remove_stop_words(tokens):
12     return [word for word in tokens if word not in kamus_stopword]
13 endfunction
14
15 data['filtered_words'] <- [remove_stop_words(sen) for sen in tokens]
16 hasil_stopword <- data['filtered_words']
17 output "Stopword"
18 output hasil_stopword
19 //mengubah hasil filtering menjadi string
20 string str_hasil_stopword <- str(hasil_stopword)
```

h) Modul Stemming

Modul *stemming* merupakan modul untuk merubah data ulasan hasil *filtering* menjadi kata dasarnya dengan mengecek apakah terdapat imbuhan *inflection suffixes*, *derivational suffixes* dan *derivational prefixes* pada kata tersebut. Modul *stemming* menggunakan fungsi *stemmer_factory()* yang terdapat pada *library sastrawi* di bahasa pemrograman *python*. Proses *stemming* dilakukan oleh *library sastrawi* dengan menerapkan algoritma Nazief dan Adriani. Berikut ini merupakan modul *stemming* yang terdapat pada Algoritma 12.

Algoritma 12: Stemming

```
1  require: data: data array ulasan di google maps hasil filtering
2  require: StemmerFactory: module Sastrawi.Stemmer.StemmerFactory untuk
3  merubah token menjadi kata dasar
4
5  factory <- StemmerFactory()
6  stemmer <- factory.create_stemmer()
7  function stemming(tokens):
8      data_stem <- []
9      for i in tokens do
10         kata <- stemmer.stem(i)
11         data_stem.append(kata)
12     endfor
13     return data_stem
14 endfunction
15 data['Akhir'] <- data['filtered_words'].apply(lambda x: stemming(x))
16 hasil_stemming <- data['Akhir']
17 output "Stemming"
18 output hasil_stemming
```

4.1.1.3 Modul *Data labeling*

Modul *data labeling* merupakan modul yang dilakukan setelah data ulasan melalui proses *text preprocessing* dimana merubah label negatif pada data ulasan dengan 0 dan merubah label positif pada data ulasan dengan 1 sebelum masuk ke dalam proses ekstraksi fitur (*term weighting TF-IDF*). Berikut ini merupakan modul *data labeling* yang terdapat pada Algoritma 13.

Algoritma 13: Data Labeling

```
1  require: data array ulasan di google maps hasil text preprocessing
2  require: y_train: data array variabel y_train
3
4  y_train <- []
5  for i in data['label'] do
6      //diberi angka 0 jika berlabel negatif
7      if i = 'NEGATIF' then
8          label = 0
9          y_train.append(label)
10     //diberi angka 1 jika berlabel positif
11     else if i = 'POSITIF' then
12         label = 1
13         y_train.append(label)
14     endif
15 endfor
16 output y_train
17 output len(y_train)
```

4.1.1.4 Modul ekstraksi fitur (*term weighting TF-IDF*)

Modul ekstraksi fitur (*term weighting TF-IDF*) merupakan modul yang digunakan untuk melakukan pembobotan kata atau fitur dengan menggunakan metode TF-IDF. Modul ekstraksi fitur (*term weighting TF-IDF*) menggunakan fungsi *TfidfVectorizer()* yang digunakan untuk menghitung nilai TF, IDF, dan TF-IDF, sedangkan fungsi *fit_transform* digunakan untuk menghasilkan vector matrix multidimensional. Kedua fungsi tersebut terdapat pada *library sklearn.feature_extraction.text* di dalam bahasa pemrograman *python*. Berikut ini merupakan modul ekstraksi fitur (*term weighting TF-IDF*) yang terdapat pada Algoritma 14.

Algoritma 14: Ekstraksi Fitur (Term Weighting TF-IDF)

```
1  require: data: data array ulasan di google maps hasil text preprocessing
2  require: sklearn.feature_extraction.text : module untuk ekstraksi fitur dari
3  data hasil text preprocessing dengan menggunakan term weighting TF-IDF
4  require: pickle: module untuk menyimpan hasil ekstraksi fitur ke dalam
5  file dengan format .pickle
6
7  tfidf_vectorize <- TfidfVectorizer()
8  tfidf_textfinal <- tfidf_vectorize.fit_transform(text_final)
9
10 //membuat file dengan nama pypickle_tfid.pickle
11 with open('pypickle_tfid.pickle', 'wb') as ftfid:
12     pickle.dump(tfidf_vectorize, ftfid, pickle.HIGHEST_PROTOCOL)
```

4.1.1.5 Modul seleksi fitur

Modul seleksi fitur merupakan modul yang digunakan untuk menyeleksi atau memilih fitur yang memiliki pengaruh paling tinggi terhadap pembangunan model klasifikasi metode *Support Vector Machine* dan menghilangkan fitur yang kurang berpengaruh atau tidak berpotensi menjadi indeks. Pada penelitian ini menggunakan modul seleksi fitur *chi square* dan *mutual information* untuk dibandingkan seleksi fitur mana yang menghasilkan akurasi terbaik pada metode *Support Vector Machine*.

a) Modul *Chi Square*

Modul *chi square* merupakan modul untuk melakukan pemilihan fitur berdasarkan metode *chi square*. Pada metode seleksi fitur *chi square* pertama menghitung nilai jumlah kemunculan fitur atau kata pada kelas positif dan kelas negatif dalam data latih (data ulasan). Kedua, hasil nilai *chi square* fitur dari kedua kelas tersebut akan dibandingkan, dan akan dipilih nilai *chi square* maksimal dari fitur tersebut. Pada langkah pertama dan kedua tersebut menggunakan fungsi *chi2()* yang terdapat pada *library sklearn.feature_selection* di dalam bahasa pemrograman *python*. Ketiga, setelah semua fitur telah diketahui nilai bobot *chi square*-nya, maka semua fitur akan diurutkan berdasarkan nilai bobot *chi square* tertinggi ke terendah. Selanjutnya mengambil 2400 fitur yang memiliki bobot tertinggi yang akan digunakan untuk pembuatan model klasifikasi metode *support vector machine* dengan menggunakan fungsi *SelectKBest()* yang terdapat pada *library sklearn.feature_selection* di dalam bahasa pemrograman *python*. Berikut ini merupakan modul seleksi fitur *chi square* yang terdapat pada Algoritma 15.

Algoritma 15: Seleksi Fitur Chi Square

```
1  require: data: data array ulasan di google maps hasil ekstraksi fitur
2  require: sklearn.feature_extraction.text: module untuk mengambil fitur
3  hasil ekstraksi fitur menggunakan term weighting TF-IDF
4  require: sklearn.feature_selection: module untuk pemilihan fitur
5
6  X_train, X_test <- tfidf_textfinal[train], tfidf_textfinal[test]
7  y <- np.array(y_train)
8  Y_train, Y_test <- y[train], y[test]
9
10 //menghitung bobot chi square setiap fitur
11 chi2score <- chi2(X_train, Y_train)
12 chidump <- []
13 for i in np.array(chi2score) do
14     for j in I do
15         //jika j merupakan tak terhingga
16         if math.isnan(j) then
17             j <- 1 //maka j diberi nilai 1
18         endif
19         chidump.append(j)
20     endfor
21 endfor
22
23 wscores_chi <- zip(tfidf_vectorize.get_feature_names(), chidump)
24 //mengurutkan fitur dari bobot chi square tertinggi ke terendah
25 wchi2 <- sorted(wscores_chi, key=lambda x:x[1])
26 //list 25 fitur yang memiliki bobot chi square tertinggi
27 topchi2 <- list(zip(*wchi2[-25:]))
28 sf_topchi2.append(topchi2)
29 x_chi <- range(len(topchi2[1]))
30 labels_chi <- topchi2[0]
31 //menentukan banyaknya fitur yang akan digunakan
32 ch2 <- SelectKBest(chi2, k=2400)
33 with open('pypickle_fschi.pickle', 'wb') as fschi:
34     pickle.dump(ch2, fschi, pickle.HIGHEST_PROTOCOL)
```

b) Modul *Mutual Information*

Modul *mutual information* merupakan modul untuk melakukan pemilihan fitur berdasarkan metode *mutual information*. Pada metode seleksi fitur *mutual information* pertama menghitung nilai jumlah kemunculan fitur atau kata pada kelas positif dan kelas negatif dalam data latih (data ulasan). Kedua, hasil nilai *mutual information* fitur dari kedua kelas tersebut akan dibandingkan, dan akan dipilih nilai *mutual information* terbesar dari

kedua bobot tersebut. Pada langkah pertama dan kedua tersebut menggunakan fungsi *mutual_info_classif()* yang terdapat pada *library sklearn.feature_selection* di dalam bahasa pemrograman *python*. Ketiga, setelah semua fitur telah diketahui nilai bobot *mutual information*-nya, maka semua fitur akan diurutkan berdasarkan nilai bobot *mutual information* tertinggi ke terendah. Selanjutnya mengambil 2400 fitur yang memiliki bobot tertinggi yang akan digunakan untuk pembuatan model klasifikasi metode *support vector machine* dengan menggunakan fungsi *SelectKBest()* yang terdapat pada *library sklearn.feature_selection* di dalam bahasa pemrograman *python*. Berikut ini merupakan modul seleksi fitur *mutual information* yang terdapat pada Algoritma 16.

Algoritma 16: Seleksi Fitur Mutual Information

```

1  require: data: data array ulasan di google maps hasil ekstraksi fitur
2  require: sklearn.feature_extraction.text: module untuk mengambil fitur
3  hasil ekstraksi fitur menggunakan term weighting TF-IDF
4  require: sklearn.feature_selection: module untuk pemilihan fitur
5
6  X_train, X_test <- tfidf_textfinal[train], tfidf_textfinal[test]
7  y <- np.array(y_train)
8  Y_train, Y_test <- y[train], y[test]
9
10 //menghitung bobot mutual information setiap fitur
11 mutual_info_score <- mutual_info_classif(X_train, Y_train)
12 wscores_mi <- list(zip(tfidf_vectorize.get_feature_names(),
13                         mutual_info_score))
14 //mengurutkan fitur dari bobot mutual information tertinggi ke terendah
15 w_mi <- sorted(wscores_mi, key=lambda x:x[1])
16 //list 25 fitur yang memiliki bobot mutual information tertinggi
17 top_mi <- list(zip(*w_mi[-25:]))
18 sf_topmi.append(top_mi)
19 x_mi <- range(len(top_mi[1]))
20 labels_mi <- top_mi[0]
21 //menentukan banyaknya fitur yang akan digunakan
22 fs <- SelectKBest(mutual_info_classif, k=2400)
23 with open('pypickle_fsmi.pickle', 'wb') as fsmi:
24     pickle.dump(fs, fsmi, pickle.HIGHEST_PROTOCOL)

```

4.1.1.6 Modul *support vector machine*

a) Modul Pelatihan *Support Vector Machine* dengan Seleksi Fitur *Chi Square*

Modul pelatihan *support vector machine* dengan seleksi fitur *chi square* merupakan modul untuk pembuatan model klasifikasi metode *support vector machine* dengan seleksi fitur *chi square*. Data array ulasan di *google maps* setelah seleksi fitur *chi square* kemudian akan dibagi menjadi data latih untuk pembuatan model klasifikasi SVM dan data uji untuk mengetahui hasil prediksi berdasarkan *k-fold*. Fungsi *ch2.fit_transform()* digunakan untuk mengubah data latih menjadi *vector matrix* multidimensional dengan TF-IDF hasil seleksi fitur *chi square*. Fungsi *ch2.transform* digunakan untuk mencocokan data uji dengan TF-IDF seleksi fitur *chi square*. Fungsi *SVC(kernel='linear', probability=True)* digunakan untuk menentukan jenis kernel yang akan digunakan (dalam penelitian ini menggunakan kernel linear) dan memunculkan probabilitas kelas positif dan negatif dari data uji yang diprediksi. Fungsi *fit()* digunakan untuk pembuatan model *support vector machine* dengan seleksi fitur *chi square*, kemudian model yang telah dibuat akan disimpan ke dalam *file* dengan format *pickle*. Fungsi *predict()* digunakan untuk memprediksi hasil data uji. Fungsi-fungsi di atas terdapat didalam *library sklearn* pada bahasa pemrograman *python*. Berikut

ini merupakan modul pelatihan *support vector machine* dengan seleksi fitur *chi square* yang terdapat pada Algoritma 17.

Algoritma 17: Pelatihan Support Vector Machine dengan Seleksi Fitur Chi Square

```
1  require: data: data array ulasan di google maps setelah seleksi fitur chi
2  square
3  require: sklearn.feature_extraction.text : module ekstrasi fitur
4  menggunakan term weighting TF-IDF
5  require: sklearn.feature_selection : module seleksi fitur
6
7  //mengubah tabel TF-IDF hasil seleksi fitur chi square dan menentukan data
8  yang menjadi data latih berdasarkan k-fold
9  x_train_chi2_selected <- ch2.fit_transform(X_train, Y_train)
10 //menentukan data latih yang akan menjadi data uji berdasarkan k-fold
11 x_validation_chi2_selected <- ch2.transform(X_test)
12
13 //menentukan fungsi kernel yang akan digunakan dan memunculkan probabilitas
14 kelas positif dan negatif dari hasil prediksi data uji
15 SVM_chi <- SVC(kernel='linear', probability=True)
16 //model klasifikasi svm dengan seleksi fitur chi square
17 SVMClassifier_chi <- SVM_chi.fit(x_train_chi2_selected, Y_train)
18
19 //hasil prediksi data latih yang menjadi data uji
20 predictions_chi <- SVMClassifier_chi.predict(x_validation_chi2_selected)
21 predictions_chitotal.append(predictions_chi)
22
23 //menyimpan model ke dalam file dengan format pickle
24 with open('pypickle_svmchi.pickle', 'wb') as fc:
25     pickle.dump(SVMClassifier_chi, fc, pickle.HIGHEST_PROTOCOL)
```

b) Modul Pengujian Support Vector Machine dengan Seleksi Fitur Chi Square

Modul pengujian *support vector machine* dengan seleksi fitur *chi square* merupakan modul untuk mengklasifikasikan data uji ulasan di *google maps* dengan menggunakan model klasifikasi metode *support vector machine* dengan seleksi fitur *chi square* yang telah dibuat pada proses pelatihan. Array data uji ulasan di *google maps* hasil *text preprocessing* kemudian dihitung kemunculan nilai TF-nya pada data latih dan dicari nilai TF-IDF-nya. Kemudian, mengubah data uji ke dalam bentuk *vector matrix* multidimensional. Selanjutnya membuka *file pickle* model klasifikasi *support vector machine* dengan seleksi fitur *chi square* untuk melakukan klasifikasi data uji. Fungsi *predict_proba()* digunakan untuk melihat probabilitas kelas positif dan negatif dari hasil prediksi data uji. Fungsi *predict()* digunakan untuk memunculkan hasil prediksi atau klasifikasi. Beberapa fungsi di atas terdapat didalam *library sklearn* pada bahasa pemrograman *python*. Berikut ini merupakan modul pengujian *support vector machine* dengan seleksi fitur *chi square* yang terdapat pada Algoritma 18.

Algoritma 18: Pengujian Support Vector Machine dengan Seleksi Fitur Chi Square

```
1  require: data: array data uji ulasan di google maps
2  require: sklearn.feature_extraction.text: module ekstrasi fitur menggunakan
3  term weighting TF-IDF
4  require: sklearn.feature_selection: module seleksi fitur
5
6  //text final merupakan hasil data uji setelah melalui text preprocessing
7  text_final <- [' '.join(sen) for sen in hasil_stemming]
8
9  //membaca file pickle dari tabel tfidf
10 with open('pypickle_tfidf.pickle', 'rb') as tfidf:
11     tfidf_vectorize <- pickle.load(tfidf)
12 //menghitung nilai TF text final (data uji) terhadap data latih dan mencari
13 nilai tfidfnya kemudian diubah ke dalam vector matrix multidimensional
14 data_tfidf <- tfidf_vectorize.transform(text_final)
```

Algoritma 19: Lanjutan Pengujian Support Vector Machine dengan Seleksi Fitur Chi Square

```
15 //membaca file pickle dari seleksi fitur chi square
16 with open('pypickle_fschi.pickle', 'rb') as fschi:
17     ch2 <- pickle.load(fschi)
18     data_chi <- ch2.transform(data_tfidf)
19
20 //membaca model svm dengan seleksi fitur chi square
21 with open('pypickle_svmchi.pickle', 'rb') as svmchi:
22     SVMClassifier_chi <- pickle.load(svmchi)
23
24 //untuk mengetahui perbandingan hasil dari prediksi data uji untuk kelas
25 positif dan negatif
26 hasil1 <- SVMClassifier_chi.predict_proba(data_chi)
27 //hasil prediksi data uji
28 hasil11 <- SVMClassifier_chi.predict(data_chi)
```

c) Modul Pelatihan *Support Vector Machine* dengan Seleksi Fitur *Mutual Information*

Modul pelatihan *support vector machine* dengan seleksi fitur *mutual information* merupakan modul untuk pembuatan model klasifikasi metode *support vector machine* dengan seleksi fitur *mutual information*. Data array ulasan di *google maps* setelah seleksi fitur *mutual information* kemudian akan dibagi menjadi data latih untuk pembuatan model klasifikasi SVM dan data uji untuk mengetahui hasil prediksi berdasarkan *k-fold*. Fungsi *fs.fit()* dan *fs.transform(x_train)* digunakan untuk mengubah data latih menjadi *vector matrix* multidimensional dengan TF-IDF hasil seleksi fitur *mutual information*. Fungsi *fs.transform(x_test)* digunakan untuk mencocokan data uji dengan TF-IDF seleksi fitur *mutual information*. Fungsi *SVC(kernel='linear', probability=True)* digunakan untuk menentukan jenis kernel yang akan digunakan (dalam penelitian ini menggunakan kernel linear) dan memunculkan probabilitas kelas positif dan negatif dari data uji yang diprediksi. Fungsi *fit()* digunakan untuk pembuatan model *support vector machine* dengan seleksi fitur *mutual information*, kemudian model yang telah dibuat akan disimpan ke dalam *file* dengan format *pickle*. Fungsi *predict()* digunakan untuk memprediksi hasil data uji. Fungsi-fungsi di atas terdapat didalam *library sklearn* pada bahasa pemrograman *python*. Berikut ini merupakan modul pelatihan *support vector machine* dengan seleksi fitur *mutual information* yang terdapat pada Algoritma 20.

Algoritma 20: Pelatihan Support Vector Machine dengan Seleksi Fitur Mutual Information

```
1 require: data: data array ulasan di google maps setelah seleksi fitur
2 mutual information
3 require: sklearn.feature_extraction.text : module ekstrasi fitur
4 menggunakan term weighting TF-IDF
5 require: sklearn.feature_selection : module seleksi fitur
6
7 //menentukan data latih dan mengubah menjadi TF-IDF hasil seleksi fitur
8 information
9 fs.fit(X_train, Y_train)
10 //mengubah data latih menjadi vector matrix multidimensional hasil dari TF-
11 IDF setelah seleksi fitur mutual information
12 X_train_fs <- fs.transform(X_train)
13 //menentukan data latih yang akan menjadi data uji berdasarkan k-fold
14 X_test_fs <- fs.transform(X_test)
15
16 //menentukan fungsi kernel yang akan digunakan dan memunculkan probalitas
17 kelas positif dan negatif dari hasil prediksi data uji
18 SVM_mi <- SVC(kernel='linear', probability=True)
19 //môdel klasifikasi svm dengan seleksi fitur mutual information
20 SVMClassifier_mi <- SVM_mi.fit(X_train_fs, Y_train)
```

Algoritma 21: Lanjutan Pelatihan Support Vector Machine dengan Seleksi Fitur Mutual Information

```
21 //hasil prediksi data latih yang menjadi data uji
22 predictions_mi <- SVMClassifier_mi.predict(X_test_fs)
23 predictions_mitotal.append(predictions_mi)
24
25 //menyimpan model svm dengan seleksi fitur mutual information ke dalam file
26 dengan format pickle
27 with open('pypickle_svmmi.pickle', 'wb') as fm:
28     pickle.dump(SVMClassifier_mi, fm, pickle.HIGHEST_PROTOCOL)
```

d) Modul Pengujian *Support Vector Machine* Dengan Seleksi Fitur Mutual Information

Modul pengujian *support vector machine* dengan seleksi fitur *mutual information* merupakan modul untuk mengklasifikasikan data uji ulasan di *google maps* dengan menggunakan model klasifikasi metode *support vector machine* dengan seleksi fitur *mutual information* yang telah dibuat pada proses pelatihan. Array data uji ulasan di *google maps* hasil *text preprocessing* kemudian dihitung kemunculan nilai TF-nya pada data latih dan dicari nilai TF-IDF-nya. Kemudian, mengubah data uji ke dalam bentuk *vector matrix* multidimensional. Selanjutnya membuka *file pickle* model klasifikasi *support vector machine* dengan seleksi fitur *mutual information* untuk melakukan klasifikasi data uji. Fungsi *predict_proba()* digunakan untuk melihat probabilitas kelas positif dan negatif dari hasil prediksi data uji. Fungsi *predict()* digunakan untuk memunculkan hasil prediksi atau klasifikasi. Beberapa fungsi di atas terdapat didalam *library sklearn* pada bahasa pemrograman *python*. Berikut ini merupakan modul pengujian *support vector machine* dengan seleksi fitur *mutual information* yang terdapat pada Algoritma 22.

Algoritma 22: Pengujian Support Vector Machine dengan Seleksi Fitur Mutual Information

```
1 require: data: array data uji ulasan di google maps
2 require: sklearn.feature_extraction.text: module ekstrasi fitur menggunakan
3 term weighting TF-IDF
4 require: sklearn.feature_selection: module seleksi fitur
5
6 //text final merupakan hasil data uji setelah melalui text preprocessing
7 text_final <- [' '.join(sen) for sen in hasil_stemming]
8
9 //membaca file pickle dari tabel tfidf
10 with open('pypickle_tfidf.pickle', 'rb') as tfidf:
11     tfidf_vectorize <- pickle.load(tfidf)
12
13 //menghitung nilai TF text final (data uji) terhadap data latih dan mencari
14 nilai tfidfnya kemudian diubah ke dalam vector matrix multidimensional
15 data_tfidf <- tfidf_vectorize.transform(text_final)
16
17 //membaca file pickle dari seleksi fitur mutual information
18 with open('pypickle_fsmi.pickle', 'rb') as fsmi:
19     fs <- pickle.load(fsmi)
20     data_mi <- fs.transform(data_tfidf)
21
22 //membaca model svm dengan seleksi fitur mutual information
23 with open('pypickle_svmmi.pickle', 'rb') as svmmi:
24     SVMClassifier_mi <- pickle.load(svmmi)
25
26 //untuk mengetahui perbandingan hasil dari prediksi data uji untuk kelas
27 positif dan negatif
28 hasil2 <- SVMClassifier_mi.predict_proba(data_mi)
29 //hasil prediksi data uji
30 hasil22 <- SVMClassifier_mi.predict(data_mi)
```

4.1.2 Implementasi sistem

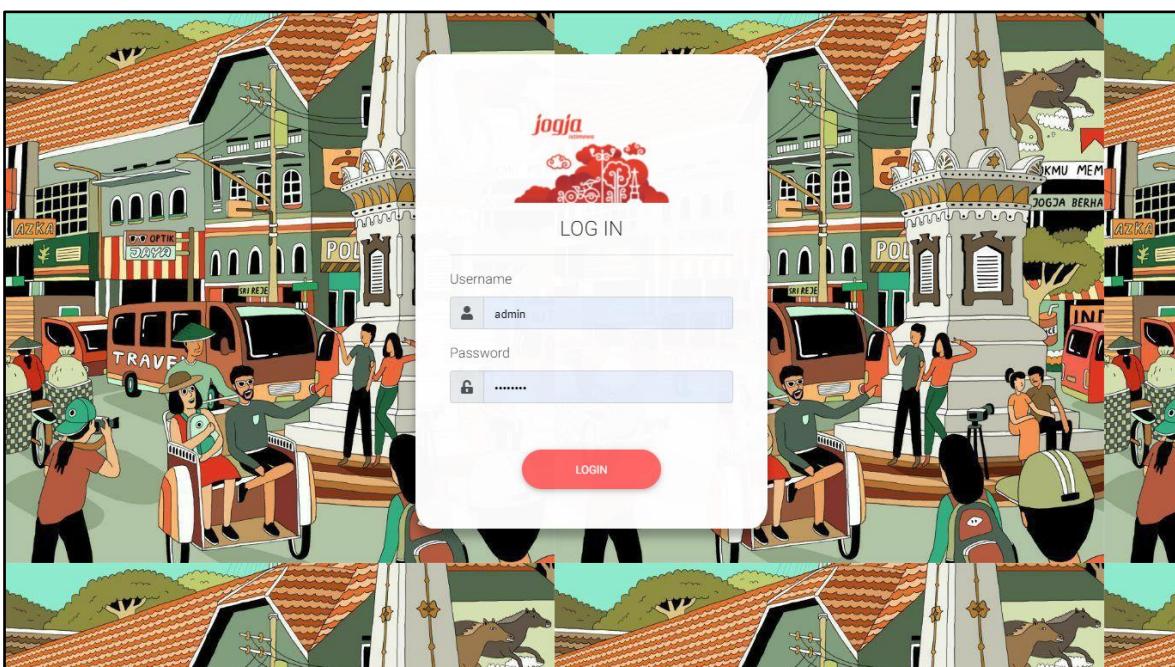
Pada bagian ini akan menampilkan hasil implementasi sistem berdasarkan rancangan yang telah dibuat pada bab sebelumnya. Pada penelitian ini terdapat dua implementasi sistem yaitu halaman admin dan halaman *user*. Berikut ini merupakan hasil *screenshot user interface* atau tampilan antarmuka dari hasil implementasi sistem yang telah dibangun.

4.1.2.1 Implementasi Halaman admin

Implementasi halaman admin berisikan tampilan *user interface* dari halaman admin. Berikut ini merupakan beberapa tampilan *user interface* dari halaman admin antara lain :

1. Halaman Login

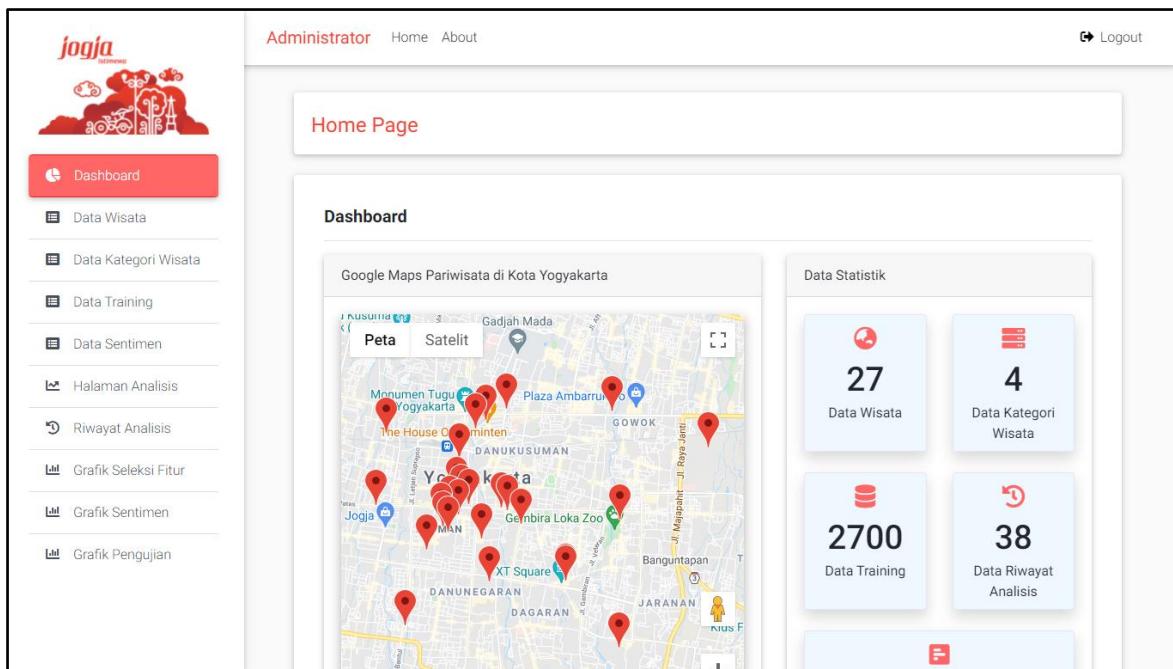
Halaman *login* merupakan halaman bagi aktor admin untuk masuk ke dalam sistem. Aktor admin memasukkan *username* dan *password* terlebih dahulu jika ingin masuk ke dalam sistem. Implementasi *user interface* pada halaman *login* dapat dilihat pada gambar 4.1.



Gambar 4.1. Implementasi User Interface Pada Halaman Login

2. Halaman Dashboard

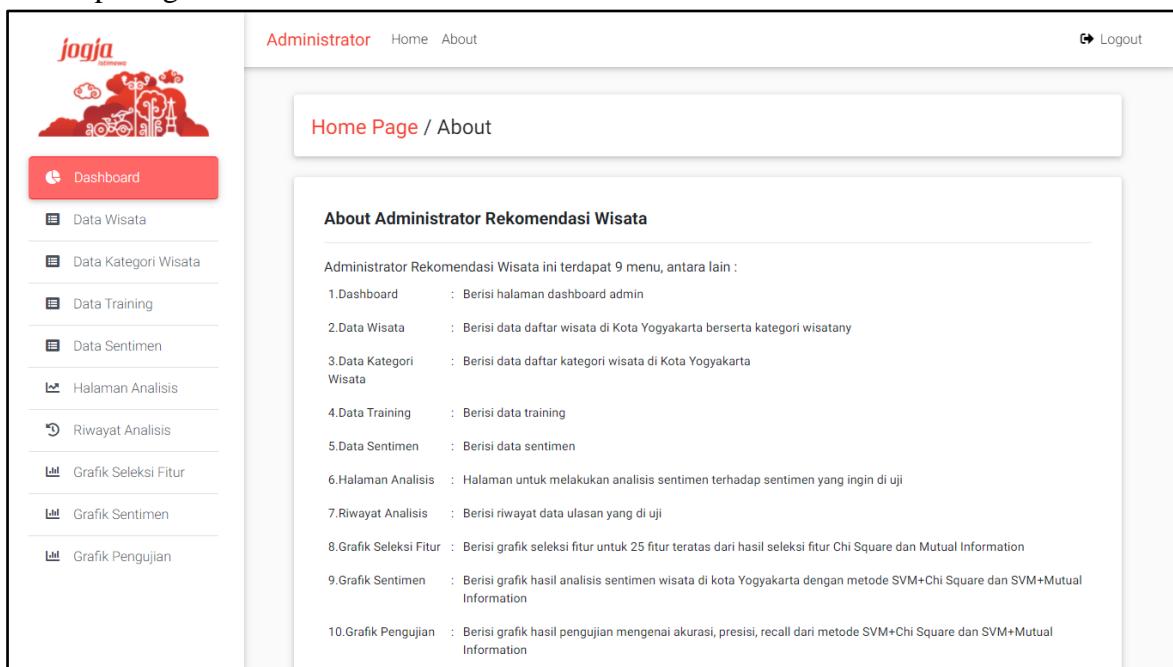
Halaman *dashboard* merupakan halaman yang pertama kali dilihat oleh aktor admin setelah melakukan proses *login*. Dalam halaman *dashboard* terdapat tampilan *google maps* letak pariwisata di Kota Yogyakarta dan beberapa data statistik. Beberapa data statistic yang ditampilkan antara lain data wisata, data kategori wisata, data training, data riwayat analisis dan data sentiment. Implementasi *user interface* pada halaman *dashboard* dapat dilihat pada gambar 4.2.



Gambar 4.2. Implementasi User Interface Pada Halaman Dashboard

3. Halaman About

Halaman *about* merupakan halaman yang berisi keterangan mengenai isi dari menu-menu yang terdapat pada halaman admin seperti menu *dashboard*, data wisata, data kategori wisata, data *training*, data sentimen, halaman analisis, riwayat analisis, grafik seleksi fitur, grafik sentimen dan grafik pengujian. Implementasi *user interface* pada halaman *about* dapat dilihat pada gambar 4.3.



Gambar 4.3. Implementasi User Interface Pada Halaman About

4. Halaman Data Wisata

Halaman data wisata merupakan halaman yang berisi daftar data wisata yang ada di Kota Yogyakarta. Implementasi *user interface* pada halaman data wisata dapat dilihat pada gambar 4.4.

The screenshot shows a web application interface for managing tourist data. On the left is a sidebar with a logo and links to various sections like Dashboard, Data Wisata, and Data Kategori Wisata. The main area has a header 'Administrator' and 'Logout'. Below is a title 'Home Page / Data Wisata' and a 'TAMBAH DATA' (Add Data) button. A toolbar at the top right includes 'Copy', 'CSV', 'Excel', 'PDF', and 'Print'. A search bar is also present. The central part is a table listing 8 tourist entries:

| No | Kategori Wisata | Nama Wisata | Aksi |
|----|--------------------|--------------------------|----------------------|
| 1 | Sejarah dan Budaya | Keraton Yogyakarta | [Edit, Delete, Info] |
| 2 | Museum | Museum Kereta Keraton | [Edit, Delete, Info] |
| 3 | Sejarah dan Budaya | Taman Sari | [Edit, Delete, Info] |
| 4 | Museum | Museum Benteng Vredeburg | [Edit, Delete, Info] |
| 5 | Belanja | Malioboro | [Edit, Delete, Info] |
| 6 | Belanja | Pasar Beringharjo | [Edit, Delete, Info] |
| 7 | Museum | Museum Sonobudoyo | [Edit, Delete, Info] |
| 8 | Pendidikan | Taman Pintar | [Edit, Delete, Info] |

The URL in the address bar is 'localhost:8080/skripsi/admin_datawisata.php'.

Gambar 4.4. Implementasi *User Interface* Pada Halaman Data Wisata

Pada halaman data wisata terdapat tombol tambah data yang berfungsi menambah data wisata di Kota Yogyakarta. Aktor admin mengisi kategori wisata, nama wisata, *latitude*, *longitude*, foto, deskripsi, alamat, jam buka, dan url untuk wisata yang akan ditambahkan. Implementasi *user interface* pada tombol tambah data wisata dapat dilihat pada gambar 4.5.

This screenshot shows a modal window titled 'Tambah Data Wisata' (Add Data). It is overlaid on the main 'Data Wisata' page. The modal contains several input fields:

- Kategori Wisata : Sejarah dan Budaya (selected from a dropdown menu)
- Nama Wisata : [empty input field]
- Latitude : [empty input field]
- Longitude : [empty input field]
- Foto : Pilih File (Browse button) - Tidak ada ...ang dipilih (No file selected)
- Deskripsi : [empty text area]
- Alamat : [empty input field]
- Jam Buka : [empty input field]
- URL : [empty input field]

Gambar 4.5. Implementasi *User Interface* Pada Tombol Tambah Data Wisata

Pada halaman data wisata terdapat beberapa pilihan menu aksi seperti detail, edit, dan hapus data wisata berdasarkan tombol yang di klik sesuai baris pada tabel. Tombol detail wisata berfungsi untuk menampilkan detail dari data wisata. Implementasi *user interface* pada tombol detail data wisata dapat dilihat pada gambar 4.6.

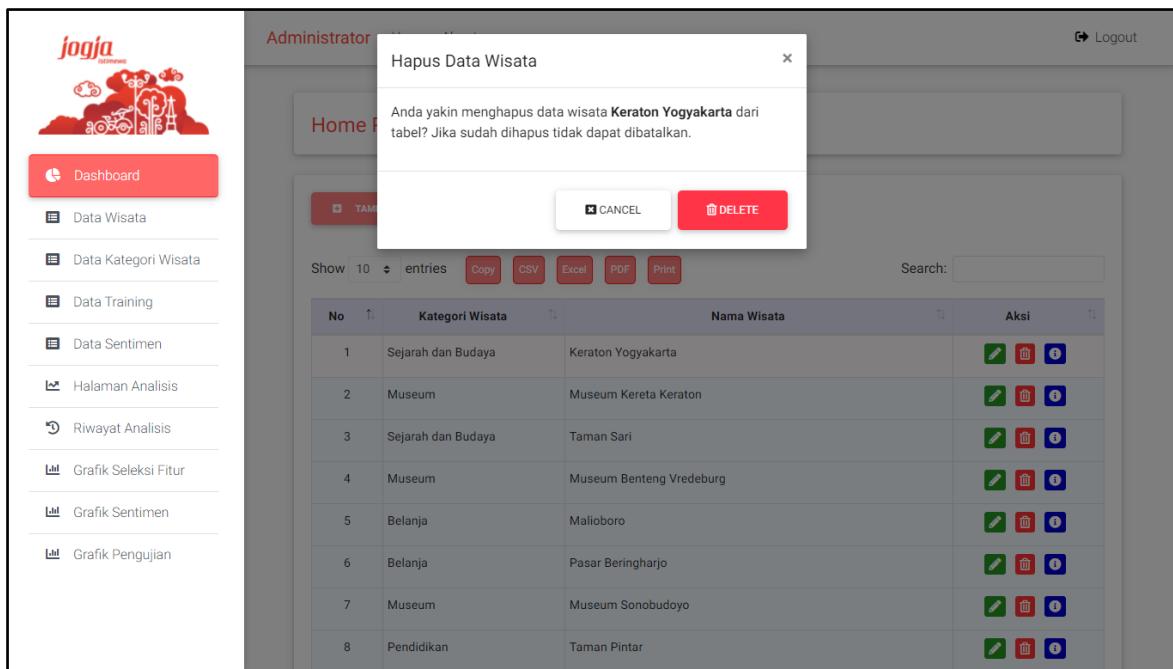
| No | | Aksi |
|----|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------|
| 1 | Keraton Yogyakarta | |
| 2 | Sejarah dan Budaya | |
| 3 | Keraton Ngayogyakarta Hadiningrat | |
| 4 | Merupakan istana resmi Kesultanan Ngayogyakarta Hadiningrat yang kini berlokasi di Kota Yogyakarta, Daerah Istimewa Yogyakarta, Indonesia. | |
| 5 | Walaupun kesultanan tersebut secara resmi telah menjadi bagian Republik Indonesia pada tahun 1950, kompleks bangunan keraton ini masih berfungsi sebagai tempat tinggal sultan dan rumah tangga istananya yang masih menjalankan tradisi kesultanan hingga saat ini. Keraton ini kini juga merupakan salah satu objek wisata di Kota Yogyakarta. | |
| 6 | Sebagian kompleks keraton merupakan museum yang menyimpan berbagai koleksi milik kesultanan, termasuk berbagai pemberian dari raja-raja Eropa, replika pusaka keraton, dan gamelan. Dari segi bangunannya, keraton ini merupakan salah satu contoh arsitektur istana Jawa yang terbaik, memiliki balairung-balairung mewah dan lapangan serta paviliun yang luas. | |
| 7 | Keraton Ngayogyakarta Hadiningrat atau Keraton Yogyakarta merupakan istana resmi Kesultanan Ngayogyakarta Hadiningrat yang kini berlokasi di Kota Yogyakarta, Daerah Istimewa Yogyakarta, Indonesia. Walaupun kesultanan tersebut secara resmi telah menjadi bagian Republik Indonesia pada tahun 1950, kompleks bangunan keraton ini masih berfungsi sebagai tempat tinggal sultan dan rumah tangga istananya yang masih menjalankan tradisi kesultanan hingga saat ini. Keraton ini kini juga merupakan salah satu objek wisata di Kota Yogyakarta. | |
| 8 | Sel-Min 8:30AM-3PM | |

Gambar 4.6. Implementasi *User Interface* Pada Tombol Detail Data Wisata

Pada tombol edit merupakan aksi untuk mengedit data wisata. Implementasi *user interface* pada tombol edit data wisata dapat dilihat pada gambar 4.7.

Gambar 4.7. Implementasi *User Interface* Pada Tombol Edit Data Wisata

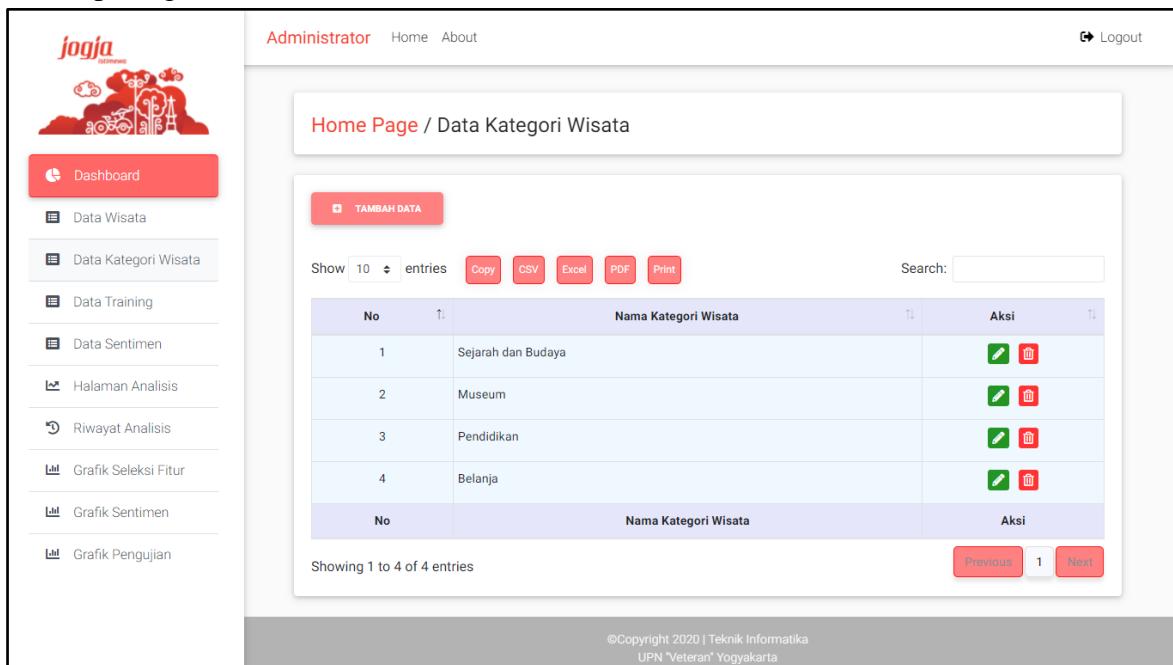
Tombol *delete* berfungsi untuk menghapus data wisata berdasarkan baris. Implementasi *user interface* pada tombol *delete* data wisata dapat dilihat pada gambar 4.8.



Gambar 4.8. Implementasi User Interface Pada Tombol Delete Data Wisata

5. Halaman Data Kategori Wisata

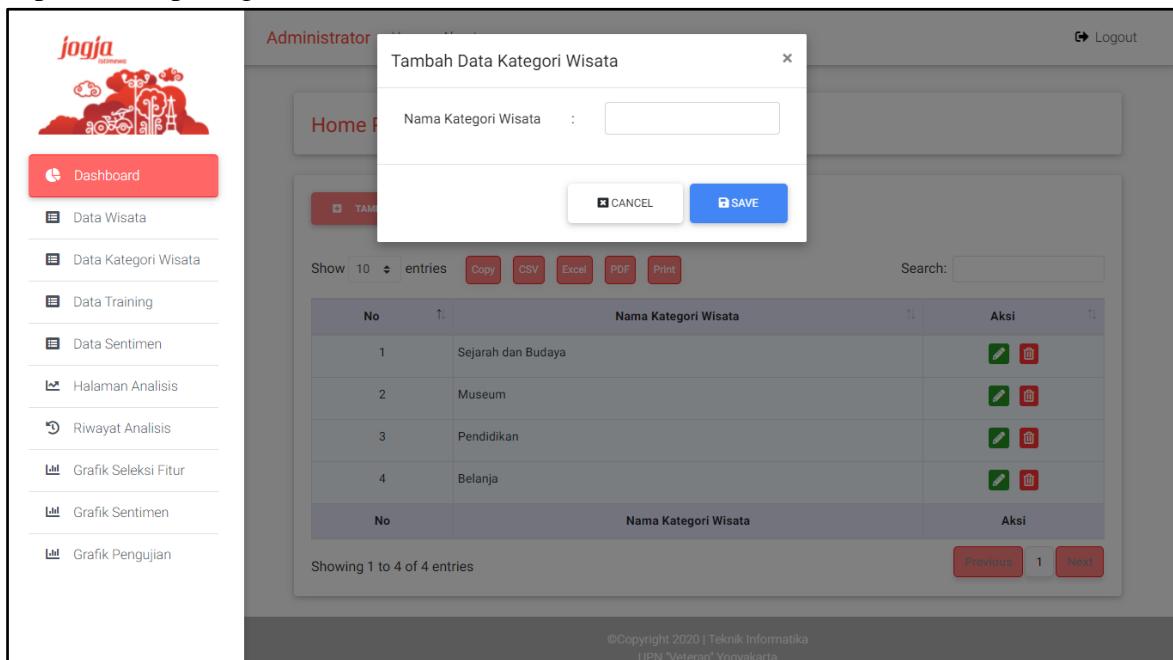
Pada halaman data kategori wisata menampilkan daftar kategori wisata yang ada di Kota Yogyakarta. Implementasi *user interface* pada halaman data kategori wisata dapat dilihat pada gambar 4.9.



Gambar 4.9. Implementasi User Interface Pada Halaman Data Kategori Wisata

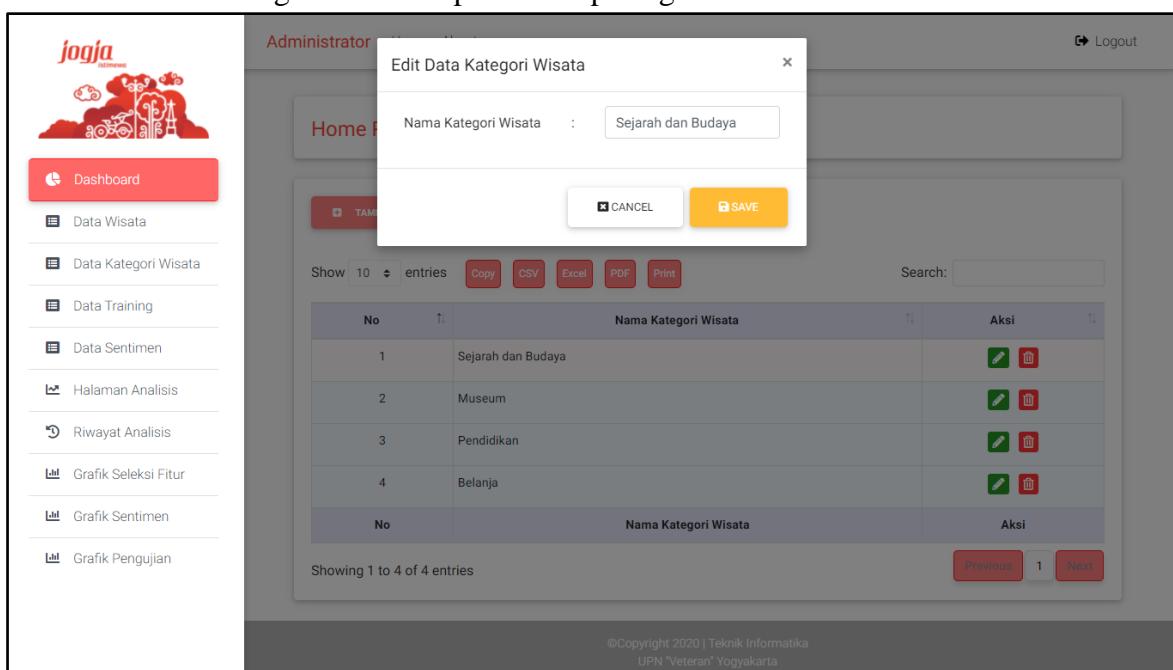
Pada halaman data kategori wisata terdapat tombol tambah data yang berfungsi menambah data kategori wisata di Kota Yogyakarta. Aktor admin mengisi nama kategori wisata yang baru yang akan ditambahkan, kemudian menekan tombol *save* untuk menyimpan data kategori wisata baru atau *cancel* untuk membatalkan menyimpan data

kategori wisata baru. Implementasi *user interface* pada tombol tambah data kategori wisata dapat dilihat pada gambar 4.10.



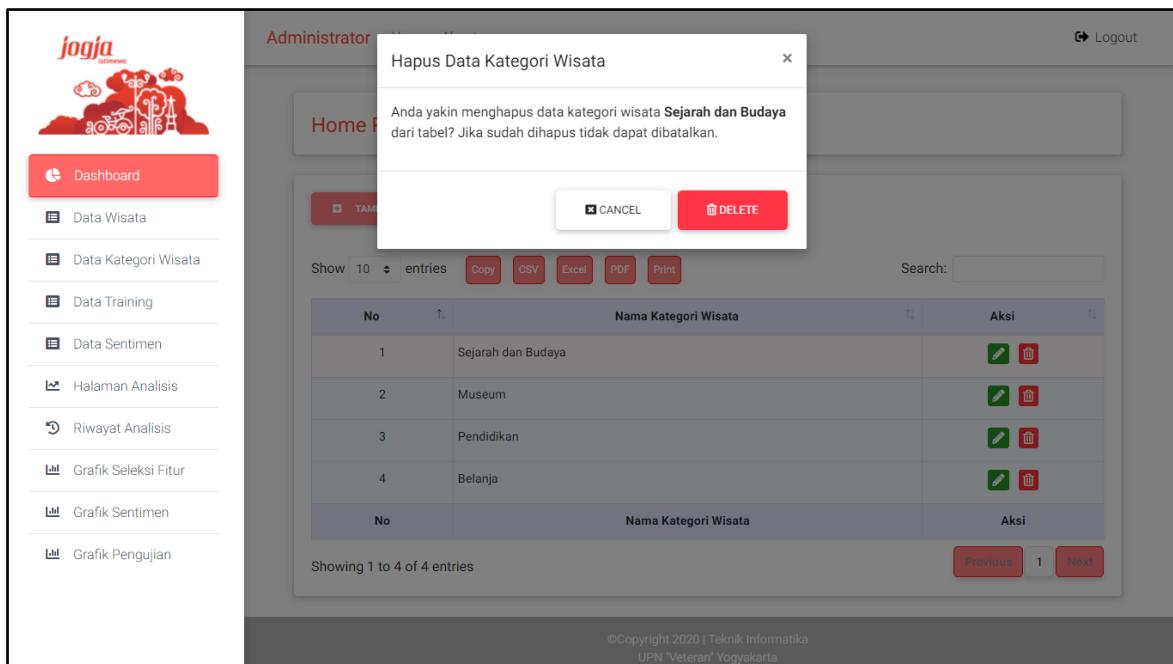
Gambar 4.10. Implementasi User Interface Pada Tombol Tambah Data Kategori Wisata

Pada halaman ini juga terdapat tombol aksi hapus dan edit. Tombol edit berfungsi untuk mengedit data kategori wisata berdasarkan baris. Implementasi *user interface* pada tombol edit data kategori dapat dilihat pada gambar 4.11.



Gambar 4.11. Implementasi User Interface Pada Tombol Edit Data Kategori Wisata

Tombol aksi *delete* berfungsi untuk menghapus kategori wisata berdasarkan baris. Implementasi *user interface* pada tombol *delete* data kategori wisata dapat dilihat pada gambar 4.12.



Gambar 4.12. Implementasi User Interface Pada Tombol Delete Data Kategori Wisata

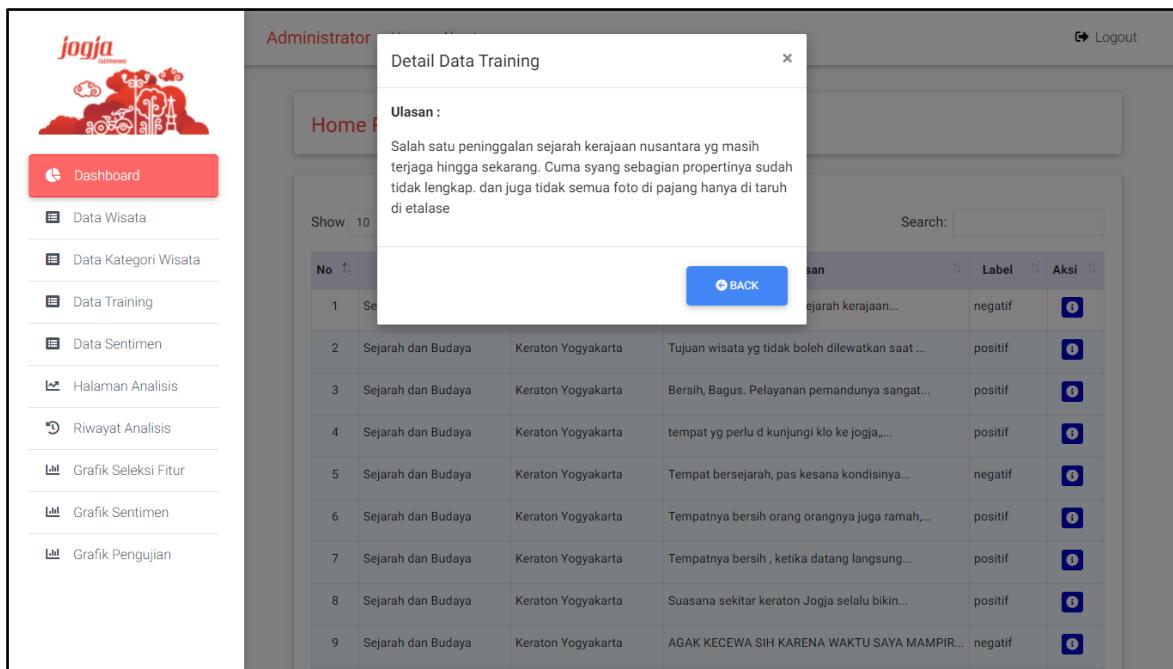
6. Halaman Data Training

Pada halaman data *training* menampilkan daftar data *training* atau data latih yang digunakan untuk pembuatan atau pembentukan model. Implementasi *user interface* pada halaman data *training* dapat dilihat pada gambar 4.13.

| No | Kategori Wisata | Nama Wisata | Ulasan | Label | Aksi |
|----|--------------------|--------------------|--------------------------------------------------|---------|------------------------|
| 1 | Sejarah dan Budaya | Keraton Yogyakarta | Salah satu peninggalan sejarah kerajaan... | negatif | Detail |
| 2 | Sejarah dan Budaya | Keraton Yogyakarta | Tujuan wisata yg tidak boleh dilewatkan saat ... | positif | Detail |
| 3 | Sejarah dan Budaya | Keraton Yogyakarta | Bersih, Bagus. Pelayanan pemandunya sangat... | positif | Detail |
| 4 | Sejarah dan Budaya | Keraton Yogyakarta | tempat yg perlu d kunjungi klo ke jogja.... | positif | Detail |
| 5 | Sejarah dan Budaya | Keraton Yogyakarta | Tempat bersejarah, pas kesana kondisinya... | negatif | Detail |
| 6 | Sejarah dan Budaya | Keraton Yogyakarta | Tempatnya bersih orang orangnya juga ramah,... | positif | Detail |
| 7 | Sejarah dan Budaya | Keraton Yogyakarta | Tempatnya bersih , ketika datang langsung... | positif | Detail |
| 8 | Sejarah dan Budaya | Keraton Yogyakarta | Suasana sekitar keraton Jogja selalu bikin... | positif | Detail |
| 9 | Sejarah dan Budaya | Keraton Yogyakarta | AGAK KECEWA SIH KARENA WAKTU SAYA MAMPIR... | negatif | Detail |

Gambar 4.13. Implementasi User Interface Pada Halaman Data Training

Pada halaman data *training* terdapat tombol aksi detail untuk menampilkan ulasan asli yang lebih lengkap atau detail sesuai dengan baris pada tombol. Implementasi *user interface* pada tombol aksi detail data *training* dapat dilihat pada gambar 4.14.



Gambar 4.14. Implementasi User Interface Pada Tombol Detail Data Training

7. Halaman Data Sentimen

Pada halaman data sentimen menampilkan daftar data latih yang juga menjadi data uji dan data uji baru hasil *scraping* yang sudah di analisis sentimen atau diklasifikasi (klasifikasi positif atau negatif) berdasarkan metode SVM dengan *Chi Square* dan SVM dengan *Mutual Information*. Implementasi *user interface* pada halaman data sentimen dapat dilihat pada gambar 4.15.

| No | Kategori Wisata | Nama Wisata | Ulasan | Teks Preproses | SVM+ChiSquare | SVM+MI |
|----|--------------------|--------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------|---------------|---------|
| 1 | Sejarah dan Budaya | Keraton Yogyakarta | Salah satu peninggalan sejarah kerajaan nusantara yg masih terjaga hingga sekarang. Cuma syang sebagian propertinya sudah tidak lengkap. dan juga tidak semua foto di pajang hanya di taruh di etalase | salah tinggal sejarah raja nusantara jaga sayang properti lengkap foto pajang taruh etalase | negatif | negatif |
| 2 | Sejarah dan Budaya | Keraton Yogyakarta | Tujuan wisata yg tidak boleh dilewatkan saat berwisata ke jogja... disini kita bisa tahu dan belajar sejarah dan adat serta budaya jogjakarta | tuju wisata lewat wisata jogja | positif | positif |
| 3 | Sejarah dan Budaya | Keraton Yogyakarta | Bersih, Bagus. Pelayanan pemandunya sangat Mantap, | bersih bagus layan pandu mantap ramah orang jual | positif | positif |

Gambar 4.15. Implementasi User Interface Pada Halaman Data Sentimen

Pada halaman data sentimen terdapat tombol tambah data yang berfungsi menambah data sentimen berdasarkan nama wisata dengan cara *scraping* data ulasan di *google maps reviews*. Data yang di *scraping* tersebut kemudian akan menjadi data uji atau diklasifikasi

ke dalam sentimen positif atau negatif. Implementasi *user interface* pada tombol tambah data sentimen dapat dilihat pada gambar 4.16.

| No | Kategori Wisata | Nama Wisata | Ulasan | Teks Preproses | SVM+ChiSquare | SVM+MI |
|----|--------------------|--------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------|---------------|---------|
| 1 | Sejarah dan Budaya | Keraton Yogyakarta | Salah satu peninggalan sejarah kerajaan nusantara yg masih terjaga hingga sekarang. Cuma syang sebagian propertinya sudah tidak lengkap. dan juga tidak semua foto di pajang hanya di taruh di etalase | salah tinggal sejarah raja nusantara jaga sayang properti lengkap foto pajang taruh etalase | negatif | negatif |
| 2 | Sejarah dan Budaya | Keraton Yogyakarta | Tujuan wisata yg tidak boleh dilewatkan saat berwisata ke jogja... disini kita bisa tahu dan belajar sejarah dan adat serta budaya jogjakarta | tuju wisata lewat wisata jogja ajar sejarah adat budaya yogyakarta | positif | positif |
| 3 | Sejarah dan Budaya | Keraton Yogyakarta | Bersih, Bagus. Pelayanan pemandunya sangat Mantap, | bersih bagus layan pandu mantap ramah orang jual | positif | positif |

Gambar 4.16. Implementasi User Interface Pada Tombol Tambah Data Sentimen

8. Halaman Analisis

Halaman analisis merupakan halaman untuk menguji atau mengklasifikasikan data dengan memasukkan ulasan. Ulasan yang telah dimasukkan akan menampilkan hasil dari proses *text preprocessing* dan hasil dari proses klasifikasi. Implementasi *user interface* pada halaman analisis dapat dilihat pada gambar 4.17.

Gambar 4.17. Implementasi User Interface Pada Halaman Analisis

9. Halaman Riwayat Analisis

Pada halaman riwayat analisis menampilkan daftar riwayat beberapa ulasan yang telah dianalisis atau diuji pada halaman analisis. Implementasi *user interface* pada halaman riwayat analisis dapat dilihat pada gambar 4.18.

| No | Ulasan | Teks Preproses | SVM+ChiSquare | SVM+MI | Aksi |
|----|----------------------------------------|-------------------------------------------|---------------|---------|-------------------------|
| 1 | Makin terus tak terurus, apalagi... | urus covid social distance angkring... | positif | positif | <button>Detail</button> |
| 2 | Kesini sejak jaman gw SD jaman itu ... | zaman zaman kereta pajang bersih... | positif | positif | <button>Detail</button> |
| 3 | Ribet sih ada pemandunya + mungkin ... | ribet sih pandu ramai buru foto lelah... | positif | positif | <button>Detail</button> |
| 4 | Di depan udah dibantu beli tiket... | bantu beli tiket total seginipas pintu... | positif | positif | <button>Detail</button> |
| 5 | B aja | b | positif | positif | <button>Detail</button> |
| 6 | Semakin usang,, pemerintah harus... | usang perintah perhati pelihara tata... | negatif | negatif | <button>Detail</button> |
| 7 | Tidak layak | layak | positif | positif | <button>Detail</button> |
| 8 | Informasi yg didapat ga ada. Krn... | informasi kereta terang tunggu... | negatif | negatif | <button>Detail</button> |
| 9 | Penipuan, tidak ada yang... | tipu terburuburu panjang jalan | positif | positif | <button>Detail</button> |

Gambar 4.18. Implementasi *User Interface* Pada Halaman Riwayat Analisis

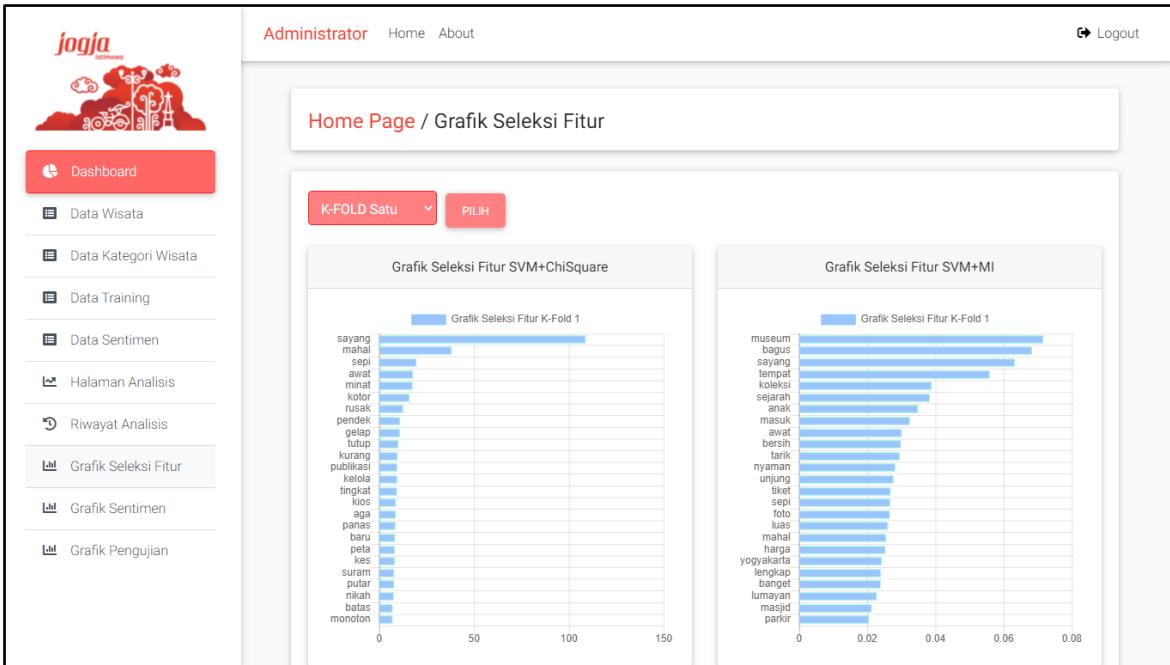
Pada halaman analisis ini juga terdapat tombol aksi detail untuk menampilkan detail dari hasil *text preprocessing* dan hasil klasifikasi pada ulasan sesuai dengan baris. Implementasi *user interface* pada tombol detail riwayat analisis dapat dilihat pada gambar 4.19.

| No | Ulasan | Teks Preproses | SVM+ChiSquare | SVM+MI | Aksi |
|----|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------|---------|-------------------------|
| 1 | untuk fasilitas yg baik dan lengkap tp pelayanan informasi tidak ramah dan tidak dapat menjelaskan informasi dengan jelas utk fasilitas kunjungan dihubungi via WhatsApp juga utk respon tidak dpt memberikan informasi yg memuaskan | untuk fasilitas yg baik dan lengkap tp pelayanan informasi tidak ramah dan tidak dapat menjelaskan informasi dengan jelas utk fasilitas kunjungan dihubungi via WhatsApp juga utk respon tidak dpt memberikan informasi yg memuaskan | positif | positif | <button>Detail</button> |
| 2 | untuk fasilitas yg baik dan lengkap tp pelayanan informasi tidak ramah dan tidak dapat menjelaskan informasi dengan jelas utk fasilitas kunjungan dihubungi via WhatsApp juga utk respon tidak dpt memberikan informasi yg memuaskan | untuk fasilitas yg baik dan lengkap tp pelayanan informasi tidak ramah dan tidak dapat menjelaskan informasi dengan jelas utk fasilitas kunjungan dihubungi via WhatsApp juga utk respon tidak dpt memberikan informasi yg memuaskan | positif | positif | <button>Detail</button> |
| 3 | untuk fasilitas yg baik dan lengkap tp pelayanan informasi tidak ramah dan tidak dapat menjelaskan informasi dengan jelas utk fasilitas kunjungan dihubungi via WhatsApp juga utk respon tidak dpt memberikan informasi yg memuaskan | untuk fasilitas yg baik dan lengkap tp pelayanan informasi tidak ramah dan tidak dapat menjelaskan informasi dengan jelas utk fasilitas kunjungan dihubungi via WhatsApp juga utk respon tidak dpt memberikan informasi yg memuaskan | positif | positif | <button>Detail</button> |
| 4 | untuk fasilitas yg baik dan lengkap tp pelayanan informasi tidak ramah dan tidak dapat menjelaskan informasi dengan jelas utk fasilitas kunjungan dihubungi via WhatsApp juga utk respon tidak dpt memberikan informasi yg memuaskan | untuk fasilitas yg baik dan lengkap tp pelayanan informasi tidak ramah dan tidak dapat menjelaskan informasi dengan jelas utk fasilitas kunjungan dihubungi via WhatsApp juga utk respon tidak dpt memberikan informasi yg memuaskan | positif | positif | <button>Detail</button> |
| 5 | untuk fasilitas yg baik dan lengkap tp pelayanan informasi tidak ramah dan tidak dapat menjelaskan informasi dengan jelas utk fasilitas kunjungan dihubungi via WhatsApp juga utk respon tidak dpt memberikan informasi yg memuaskan | untuk fasilitas yg baik dan lengkap tp pelayanan informasi tidak ramah dan tidak dapat menjelaskan informasi dengan jelas utk fasilitas kunjungan dihubungi via WhatsApp juga utk respon tidak dpt memberikan informasi yg memuaskan | positif | positif | <button>Detail</button> |
| 6 | untuk fasilitas yg baik dan lengkap tp pelayanan informasi tidak ramah dan tidak dapat menjelaskan informasi dengan jelas utk fasilitas kunjungan dihubungi via WhatsApp juga utk respon tidak dpt memberikan informasi yg memuaskan | untuk fasilitas yg baik dan lengkap tp pelayanan informasi tidak ramah dan tidak dapat menjelaskan informasi dengan jelas utk fasilitas kunjungan dihubungi via WhatsApp juga utk respon tidak dpt memberikan informasi yg memuaskan | positif | positif | <button>Detail</button> |
| 7 | untuk fasilitas yg baik dan lengkap tp pelayanan informasi tidak ramah dan tidak dapat menjelaskan informasi dengan jelas utk fasilitas kunjungan dihubungi via WhatsApp juga utk respon tidak dpt memberikan informasi yg memuaskan | untuk fasilitas yg baik dan lengkap tp pelayanan informasi tidak ramah dan tidak dapat menjelaskan informasi dengan jelas utk fasilitas kunjungan dihubungi via WhatsApp juga utk respon tidak dpt memberikan informasi yg memuaskan | positif | positif | <button>Detail</button> |
| 8 | untuk fasilitas yg baik dan lengkap tp pelayanan informasi tidak ramah dan tidak dapat menjelaskan informasi dengan jelas utk fasilitas kunjungan dihubungi via WhatsApp juga utk respon tidak dpt memberikan informasi yg memuaskan | untuk fasilitas yg baik dan lengkap tp pelayanan informasi tidak ramah dan tidak dapat menjelaskan informasi dengan jelas utk fasilitas kunjungan dihubungi via WhatsApp juga utk respon tidak dpt memberikan informasi yg memuaskan | positif | positif | <button>Detail</button> |
| 9 | untuk fasilitas yg baik dan lengkap tp pelayanan informasi tidak ramah dan tidak dapat menjelaskan informasi dengan jelas utk fasilitas kunjungan dihubungi via WhatsApp juga utk respon tidak dpt memberikan informasi yg memuaskan | untuk fasilitas yg baik dan lengkap tp pelayanan informasi tidak ramah dan tidak dapat menjelaskan informasi dengan jelas utk fasilitas kunjungan dihubungi via WhatsApp juga utk respon tidak dpt memberikan informasi yg memuaskan | positif | positif | <button>Detail</button> |

Gambar 4.19. Implementasi *User Interface* Pada Tombol Detail Riwayat Analisis

10. Halaman Grafik Seleksi Fitur

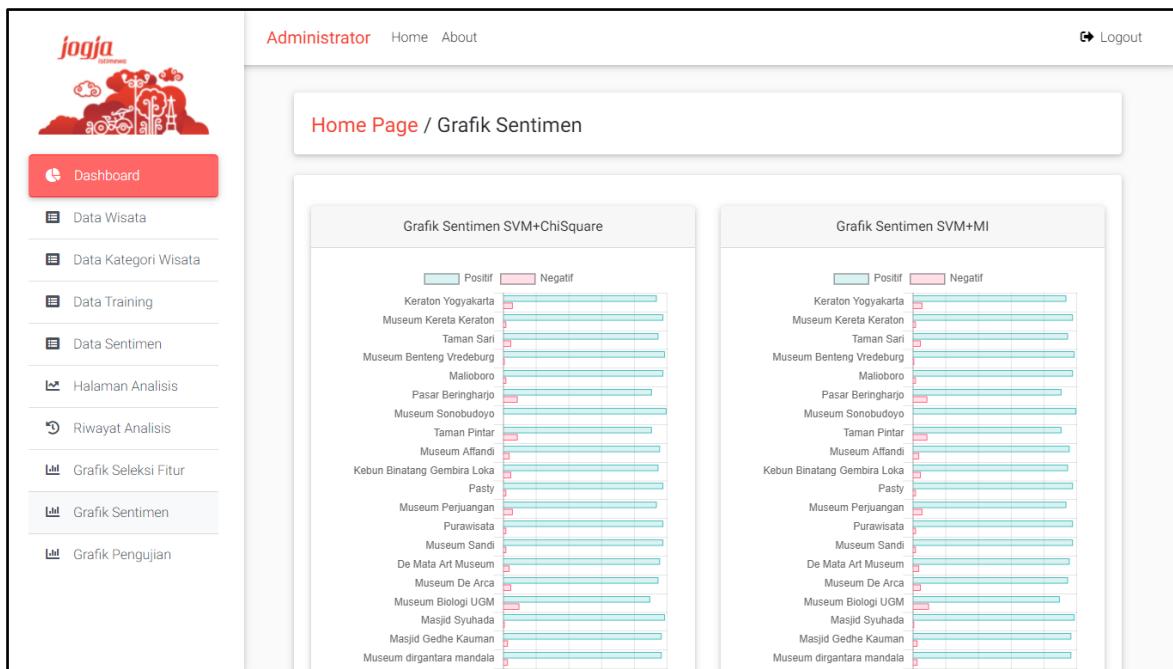
Pada halaman grafik seleksi fitur menampilkan bobot fitur atau kata berdasarkan perhitungan algoritma *chi square* dan *mutual information*. Daftar pembobotan fitur ini diurutkan berdasarkan bobot tertinggi ke terendah. Pada halaman ini hanya menampilkan 25 fitur teratas sesuai dengan *k-fold* yang dipilih. Implementasi *user interface* pada halaman grafik seleksi fitur dapat dilihat pada gambar 4.20.



Gambar 4.20. Implementasi *User Interface* Pada Halaman Grafik Seleksi Fitur

11. Halaman Grafik Sentimen

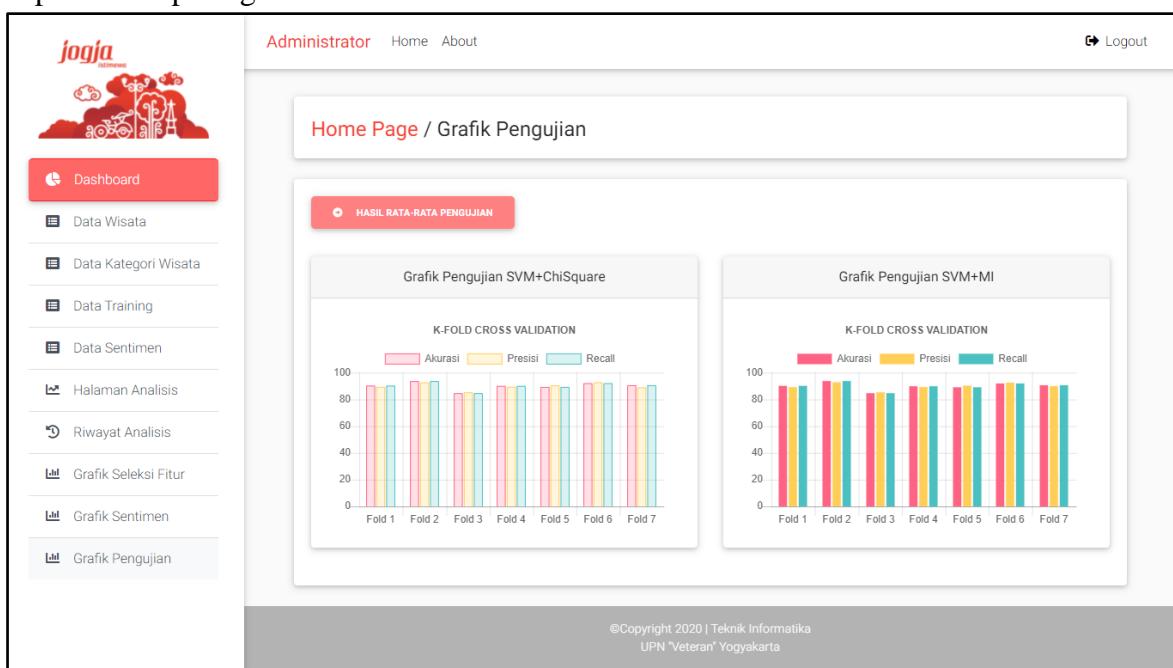
Pada halaman grafik sentimen menampilkan jumlah klasifikasi pada data sentimen positif dan negatif berdasarkan wisatanya. Pada halaman ini terdapat dua grafik sentimen SVM dengan *Chi Square* dan SVM dengan *Mutual Information*. Implementasi *user interface* pada halaman grafik sentimen dapat dilihat pada gambar 4.21.



Gambar 4.21. Implementasi User Interface Pada Halaman Grafik Sentimen

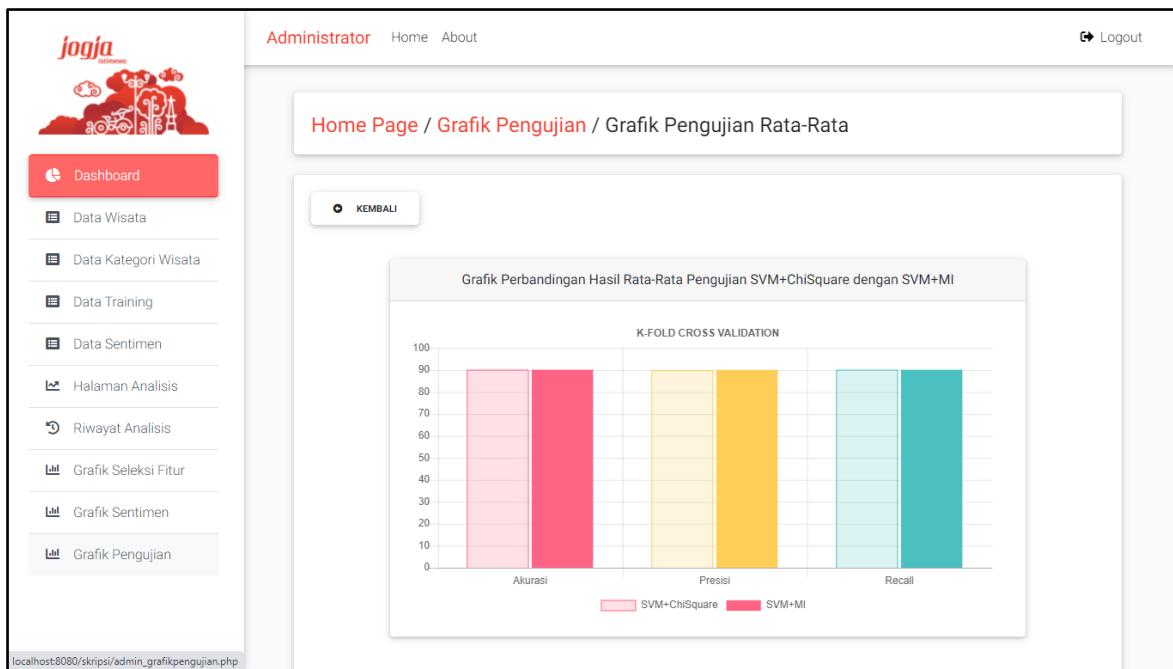
12. Halaman Grafik Pengujian

Pada halaman grafik pengujian menampilkan grafik pengujian akurasi, presisi, *recall* metode SVM dengan *Chi Square* dan metode SVM dengan *Mutual Information* berdasarkan *k-fold cross validation*-nya. Implementasi *user interface* pada halaman grafik pengujian dapat dilihat pada gambar 4.22.



Gambar 4.22. Implementasi User Interface Pada Halaman Grafik Pengujian

Pada halaman grafik pengujian juga terdapat tombol hasil rata-rata pengujian, tombol ini menampilkan hasil rata-rata dari akurasi, presisi, *recall* pada metode SVM dengan *Chi Square* dan metode SVM dengan *Mutual Information*. Implementasi *user interface* pada halaman grafik pengujian rata-rata dapat dilihat pada gambar 4.23.



Gambar 4.23. Implementasi User Interface Pada Halaman Grafik Pengujian Rata-Rata

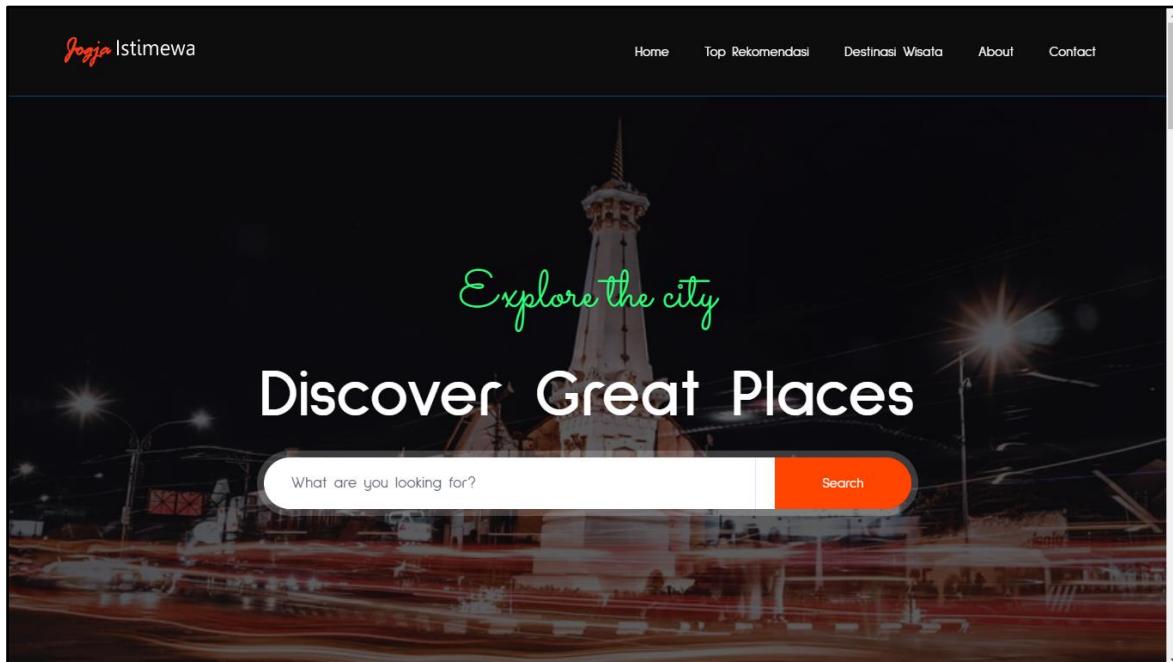
Tombol kembali pada halaman grafik pengujian rata-rata berfungsi untuk kembali ke halaman pengujian atau halaman sebelumnya.

4.1.2.2 Implementasi Halaman user

Implementasi halaman *user* berisikan tampilan *user interface* dari halaman *user*. Berikut ini merupakan beberapa tampilan *user interface* dari halaman *user* antara lain :

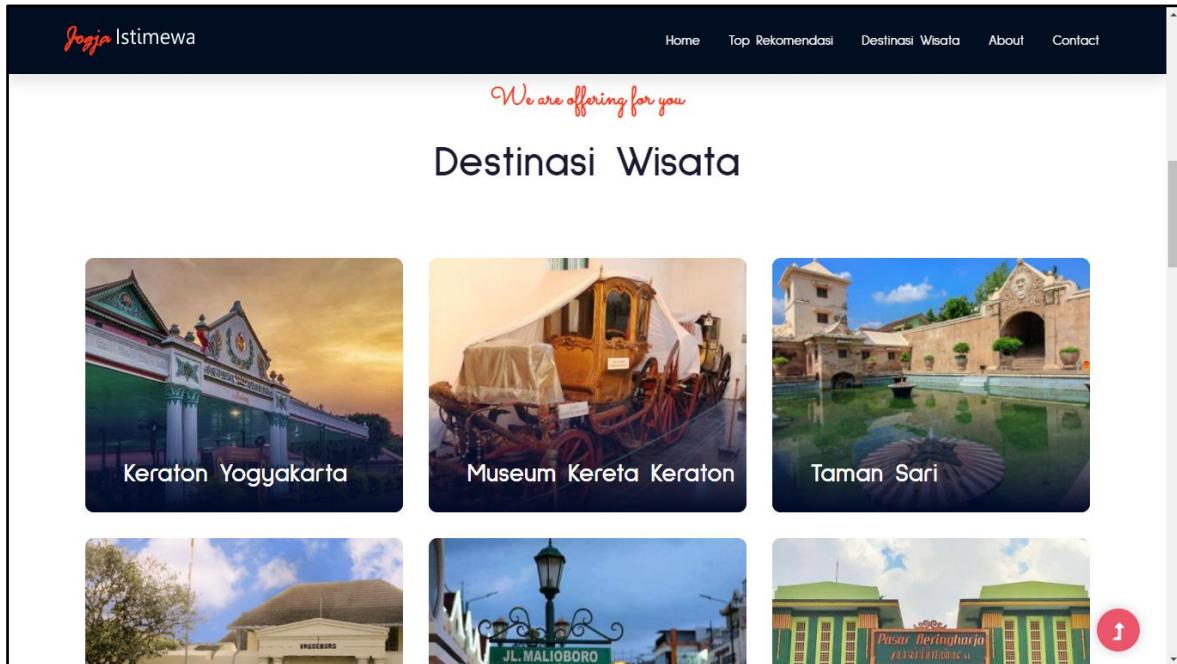
1. Halaman *Home*

Pada halaman *home* terdapat tombol *search* atau pencarian yang berfungsi untuk menampilkan halaman top rekomendasi berdasarkan kata yang diketikkan oleh aktor pengguna. Implementasi *user interface* pada halaman *home* dapat dilihat pada gambar 4.24.

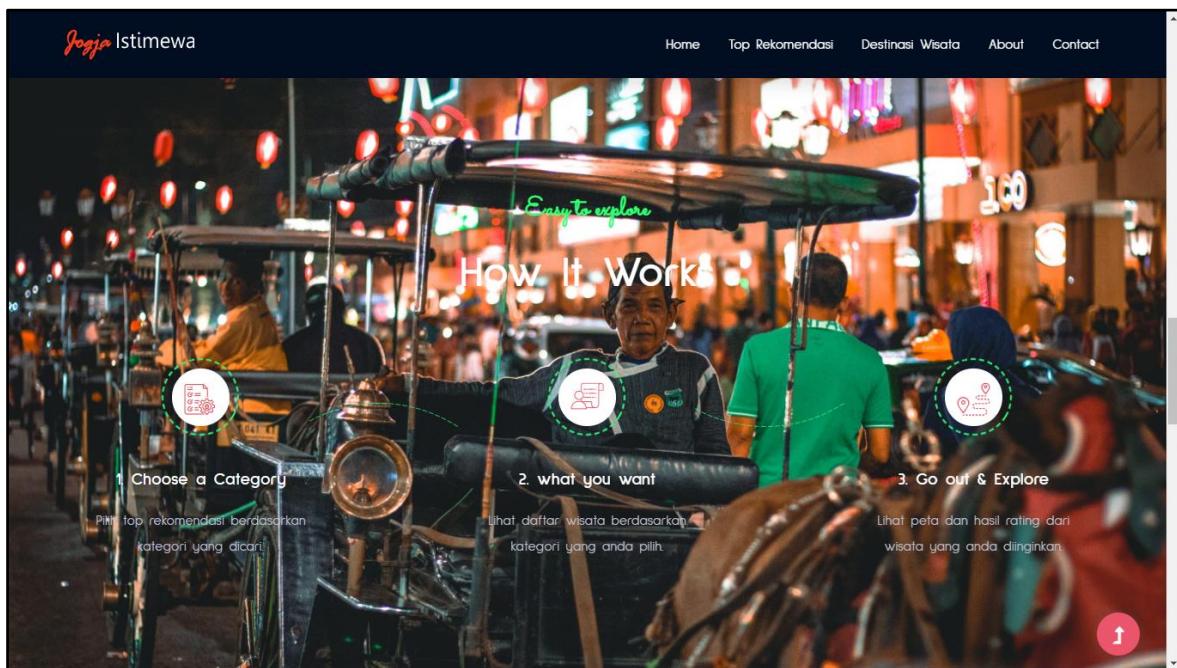


Gambar 4.24. Implementasi User Interface Pada Halaman Home

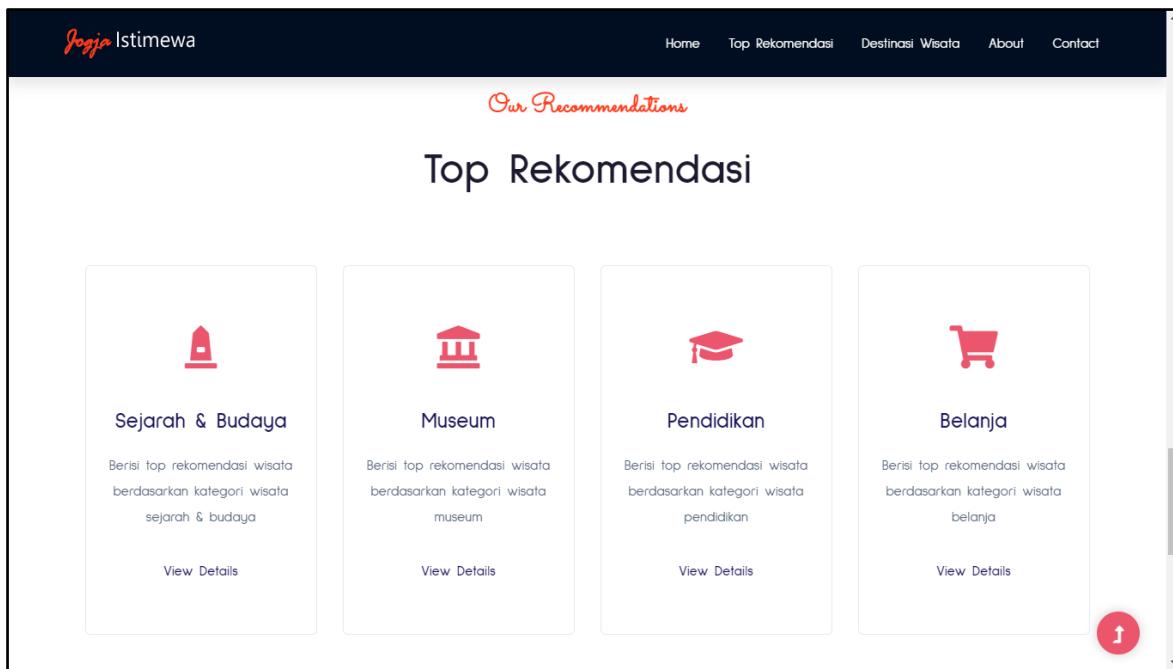
Pada halaman *home* juga terdapat menu menuju ke halaman destinasi wisata, gambar mengenai langkah-langkah atau panduan menggunakan sistem ini untuk *user* yang ingin mencari top rekomendasi, dan juga terdapat menu menuju halaman top rekomendasi yang dapat dilihat pada gambar 4.25, gambar 4.26, dan gambar 4.27.



Gambar 4.25. Implementasi *User Interface* Pada Menu Menuju Ke Halaman Destinasi Wisata



Gambar 4.26. Implementasi *User Interface* Pada Panduan Menggunakan Sistem Ini Untuk *User* Yang Ingin Mencari Top Rekomendasi



Gambar 4.27. Implementasi *User Interface* Pada Menu Menuju Ke Halaman Top Rekomendasi

2. Halaman Top Rekomendasi

Pada halaman top rekomendasi menampilkan daftar top rekomendasi untuk semua kategori atau berdasarkan kategori museum, pendidikan, belanja, sejarah dan budaya, dan berdasarkan kata kunci yang diketikkan pada menu pencarian yang berada pada halaman home. Implementasi *user interface* pada halaman top rekomendasi semua kategori wisata dapat dilihat pada gambar 4.28.

| Peringkat | Nama Wisata | Rating |
|-----------|-----------------------------|--------|
| 1 | Makam Raja Mataram Kotagede | 3.7% |
| | Museum Sonobudoyo | 3.7% |
| 2 | Jogja Gallery | 3.67% |
| | Masjid Syuhada | 3.67% |
| | Museum Benteng Vredeburg | 3.67% |
| 3 | Mallorbo | 3.63% |
| | Museum Kereta Keraton | 3.63% |

Gambar 4.28. Implementasi *User Interface* Pada Halaman Top Rekomendasi Semua Kategori Wisata

Berikut ini implementasi *user interface* pada halaman top rekomendasi berdasarkan kategori wisata yang dipilih dapat dilihat pada gambar 4.29.

| Peringkat | Nama Wisata | Rating |
|-----------|-----------------------------|--------|
| 1 | Makam Raja Mataram Kolagede | 16.67% |
| 2 | Masjid Syuhada | 16.5% |
| 3 | Purawisata | 16.33% |
| 4 | Masjid Gedhe Kauman | 16.17% |
| 5 | Taman Sari | 15.83% |
| 6 | Keraton Yogyakarta | 15.67% |

Gambar 4.29. Implementasi User Interface Pada Halaman Top Rekomendasi Berdasarkan Kategori Wisata Yang Dipilih

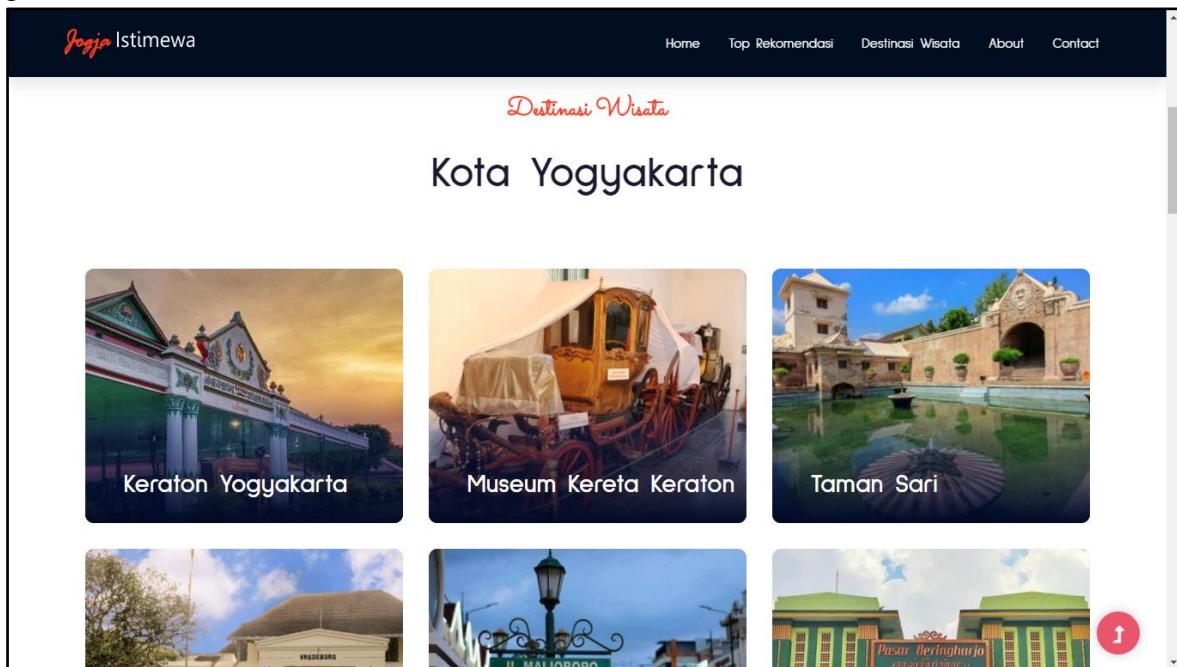
Berikut ini implementasi *user interface* pada halaman top rekomendasi berdasarkan kata kunci yang diketikkan pada menu pencarian yang berada pada halaman *home* yang dapat dilihat pada gambar 4.30.

| Peringkat | Nama Wisata | Rating |
|-----------|-----------------------|--------|
| 1 | Museum Kereta Keraton | 49% |
| 2 | Keraton Yogyakarta | 47% |

Gambar 4.30. Implementasi User Interface Pada Halaman Top Rekomendasi Berdasarkan Kata Kunci Yang Diketikkan Pada Menu Pencarian Yang Berada Pada Halaman Home

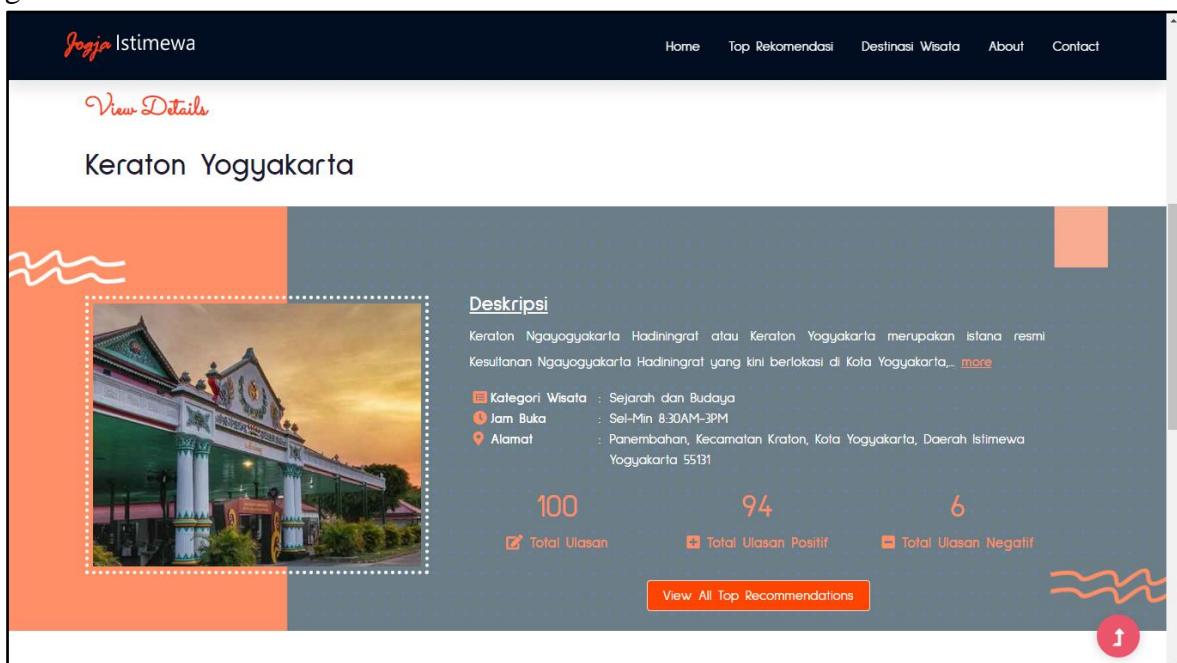
3. Halaman Destinasi Wisata

Pada halaman destinasi wisata menampilkan daftar wisata yang ada di Kota Yogyakarta. Implementasi *user interface* pada halaman destinasi wisata dapat dilihat pada gambar 4.31.



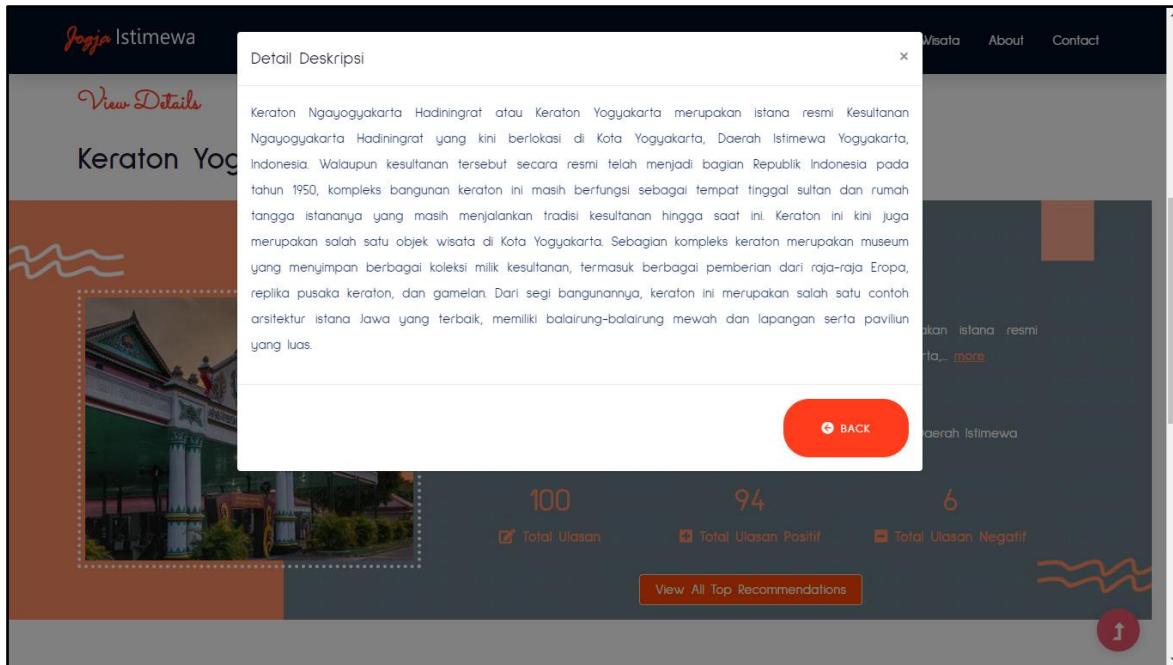
Gambar 4.31. Implementasi *User Interface* Pada Halaman Destinasi Wisata

Pada halaman ini terdapat tombol *view details* yang jika aktor *user* mengklik maka akan menuju halaman detail wisata yang diklik. Pada halaman ini menampilkan informasi singkat mengenai wisata, informasi jumlah ulasan, jumlah ulasan positif, dan jumlah ulasan negatif. Implementasi *user interface* pada halaman *view details* wisata dapat dilihat pada gambar 4.32.



Gambar 4.32. Implementasi *User Interface* Pada Halaman *View Details* Wisata

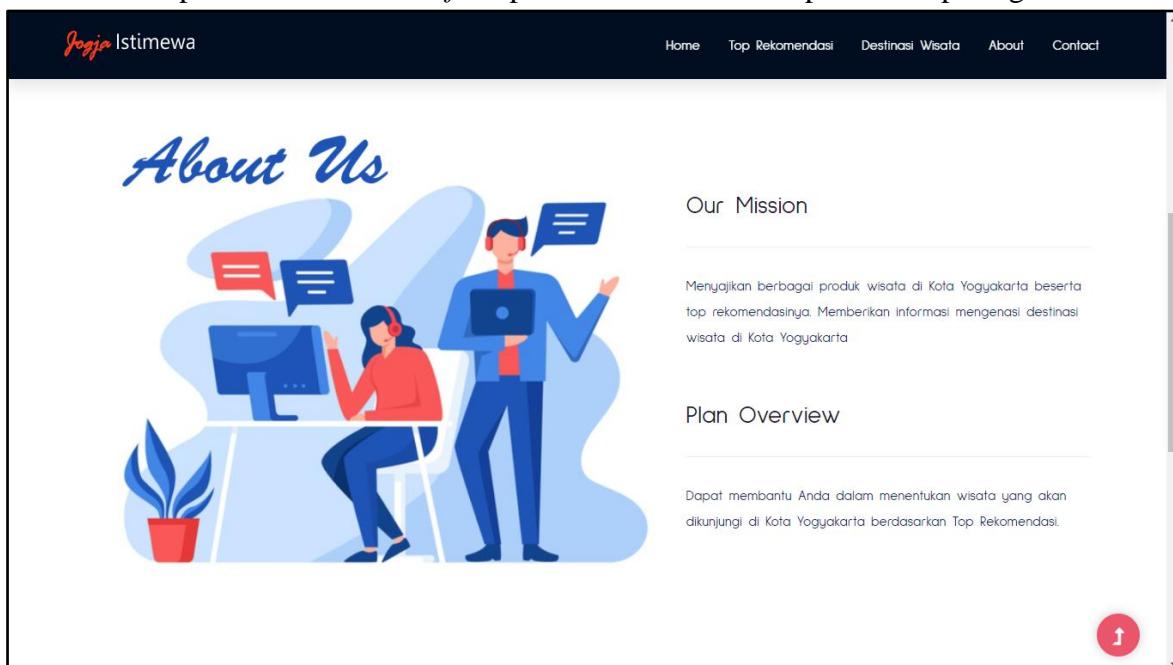
Pada halaman *view details* wisata ini juga terdapat *link more* yang jika aktor pengguna mengklik akan menampilkan detail deskripsi wisata tersebut. Implementasi *user interface* pada *link more* detail deskripsi wisata dapat dilihat pada gambar 4.33.



Gambar 4.33. Implementasi User Interface Pada Link More Detail Deskripsi Wisata

4. Halaman *About*

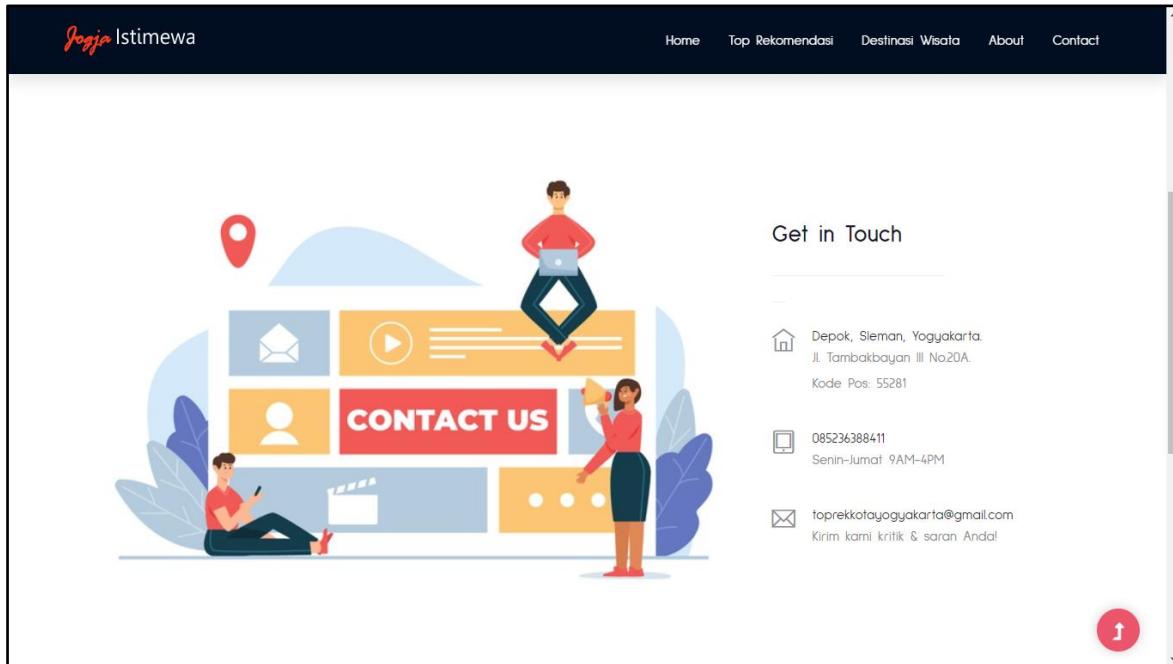
Pada halaman *about* menampilkan penjelasan singkat mengenai tujuan dibuatnya sistem ini. Implementasi *user interface* pada halaman *about* dapat dilihat pada gambar 4.34.



Gambar 4.34. Implementasi User Interface Pada Halaman About

5. Halaman *Contact*

Pada halaman *contact* menampilkan kontak yang dapat dihubungi jika aktor pengguna ingin memberikan kritik dan saran mengenai sistem ini. Implementasi *user interface* pada halaman *contact* dapat dilihat pada gambar 4.35.



Gambar 4.35. Implementasi *User Interface* Pada Halaman *Contact*

4.2. Pengujian

Pada bagian ini akan menjelaskan mengenai hasil pengujian pada penelitian yang telah dilakukan. Pengujian pada penelitian ini dibagi menjadi dua bagian yaitu pengujian model dan pengujian sistem.

4.2.1 Pengujian model

Pengujian model merupakan pengujian untuk melihat performansi dari model yang telah dibangun oleh metode yang digunakan pada penelitian ini. Pada penelitian ini menggunakan pengujian model *confusion matrix* dan *k-fold cross validation*.

4.2.1.1 Confusion matrix

Confusion matrix merupakan pengujian model untuk mengukur perbandingan performansi dari metode *support vector machine* dengan seleksi fitur *chi square* dan metode *support vector machine* dengan seleksi fitur *mutual information*. *Confusion matrix* memiliki empat kemungkinan yang merepresentasikan hasil dari proses klasifikasi, diantaranya *True Positive* (TP), *True Negative* (TN), *False Positive* (FP), *False Negative* (FN) yang akan menghasilkan nilai akurasi, presisi, dan *recall* dari model yang dibangun. Pada penelitian ini sebanyak 2700 data yang digunakan akan dibagi menjadi data latih dan data uji secara bergantian untuk setiap iterasinya, dimana sebanyak 385 atau 386 data digunakan sebagai data uji dan 2315 atau 2314 data sisanya digunakan sebagai data latih. Berikut ini merupakan hasil perhitungan *True Positive* (TP), *True Negative* (TN), *False Positive* (FP), *False Negative* (FN) dari metode *support vector machine* (SVM) dengan seleksi fitur *chi square* yang dapat dilihat pada tabel 4.1.

Tabel 4.1. Confusion Matrix Metode SVM dengan seleksi fitur Chi Square

| K-Fold | Confusion Matrix | | | |
|--------|------------------|----|----|----|
| | TP | FN | FP | TN |
| 1 | 338 | 3 | 34 | 11 |
| 2 | 352 | 5 | 19 | 10 |
| 3 | 311 | 2 | 57 | 16 |
| 4 | 338 | 2 | 36 | 10 |
| 5 | 328 | 0 | 41 | 17 |
| 6 | 341 | 0 | 30 | 14 |
| 7 | 344 | 2 | 34 | 5 |

Berikut ini merupakan hasil perhitungan *True Positive* (TP), *True Negative* (TN), *False Positive* (FP), *False Negative* (FN) dari metode *support vector machine* (SVM) dengan seleksi fitur *mutual information* yang dapat dilihat pada tabel 4.2.

Tabel 4.2. Confusion Matrix Metode SVM dengan seleksi fitur Mutual Information

| K-Fold | Confusion Matrix | | | |
|--------|------------------|----|----|----|
| | TP | FN | FP | TN |
| 1 | 338 | 3 | 34 | 11 |
| 2 | 352 | 5 | 18 | 11 |
| 3 | 311 | 2 | 56 | 17 |
| 4 | 338 | 2 | 36 | 10 |
| 5 | 328 | 0 | 41 | 17 |
| 6 | 341 | 0 | 30 | 14 |
| 7 | 345 | 1 | 34 | 5 |

4.2.1.2 K-fold cross validation

K-fold cross validation merupakan validasi untuk mengukur rata-rata keberhasilan dari model yang dibangun. Pada penelitian ini akan menggunakan iterasi $k = 7$ dimana sebanyak 2700 dataset yang digunakan akan dibagi menjadi 7 bagian. Pada iterasi pertama dataset ke-1 sampai ke-385 akan digunakan sebagai data uji sedangkan dataset ke-386 sampai 2700 akan digunakan sebagai data latih. Pembagian dataset menjadi data latih dan data uji ini akan dilakukan secara berulang sampai iterasi ke tujuh (sampai semua data pernah menjadi data uji dan data latih). Hasil dari *k-fold cross validation* ini akan memvalidasi perbandingan dari tingkat akurasi, presisi dan *recall* dari model klasifikasi yang dibangun oleh metode *support vector machine* dengan seleksi fitur *chi square* dan metode *support vector machine* dengan seleksi fitur *mutual information* untuk setiap iterasinya, sehingga dapat diketahui seleksi fitur mana yang lebih baik dalam meningkatkan performansi metode *support vector machine*. Berikut ini merupakan hasil perhitungan akurasi, presisi dan *recall* untuk setiap *k-fold cross validation* dari metode *support vector machine* (SVM) dengan seleksi fitur *chi square* yang dapat dilihat pada tabel 4.3.

Tabel 4.3. Pengujian K-Fold Cross Validation Metode SVM dengan seleksi fitur Chi Square

| K-Fold | Akurasi | Presisi | Recall |
|------------------|---------------|---------------|---------------|
| 1 | 90.41% | 89.43% | 90.41% |
| 2 | 93.78% | 92.76% | 93.78% |
| 3 | 84.72% | 85.34% | 84.72% |
| 4 | 90.16% | 89.54% | 90.16% |
| 5 | 89.38% | 90.56% | 89.38% |
| 6 | 92.21% | 92.84% | 92.21% |
| 7 | 90.65% | 89.02% | 90.65% |
| Rata-Rata | 90.19% | 89.93% | 90.19% |

Berikut ini merupakan hasil perhitungan akurasi, presisi dan *recall* untuk setiap *k-fold cross validation* dari metode *support vector machine* (SVM) dengan seleksi fitur *mutual information* yang dapat dilihat pada tabel 4.4.

Tabel 4.4. Pengujian K-Fold Cross Validation Metode SVM dengan seleksi fitur Mutual Information

| K-Fold | Akurasi | Presisi | Recall |
|------------------|--------------|--------------|--------------|
| 1 | 90.41% | 89.43% | 90.41% |
| 2 | 94.04% | 93.15% | 94.04% |
| 3 | 84.97% | 85.64% | 84.97% |
| 4 | 90.16% | 89.54% | 90.16% |
| 5 | 89.38% | 90.56% | 89.38% |
| 6 | 92.21% | 92.84% | 92.21% |
| 7 | 90.91% | 90.25% | 90.91% |
| Rata-Rata | 90.3% | 90.2% | 90.3% |

Berikut ini merupakan perbandingan hasil perhitungan rata-rata akurasi, presisi dan *recall* untuk setiap *k-fold cross validation* dari metode *support vector machine* (SVM) dengan seleksi fitur *chi square* dan metode *support vector machine* (SVM) dengan seleksi fitur *mutual information* yang dapat dilihat pada tabel 4.5.

Tabel 4.5. Perbandingan Hasil Rata-Rata Pengujian K-Fold Cross Validation

| Akurasi | | Presisi | | Recall | |
|----------------|--------------|----------------|--------------|----------------|--------------|
| SVM+Chi Square | SVM+MI | SVM+Chi Square | SVM+MI | SVM+Chi Square | SVM+MI |
| 90.19% | 90.3% | 89.93% | 90.2% | 90.19% | 90.3% |

4.2.2 Pengujian sistem

Pengujian sistem merupakan pengujian untuk menguji program dari *software* (perangkat lunak) yang telah dibangun secara lengkap dan terintegrasi. Pengujian sistem bertujuan untuk memeriksa adanya kekurangan atau kesalahan yang ada pada sistem sehingga dapat diperbaiki. Pada penelitian ini menggunakan pengujian *black box* atau pengujian secara fungsional. Pengujian *black box* merupakan pengujian *software* yang dilakukan dengan cara melihat hasil eksekusi dari uji coba tanpa mengetahui struktur dari kode program perangkat lunak tersebut. Pada penelitian ini, pengujian *black box* sistem ini dilakukan oleh 10 responden terpilih. Berikut ini merupakan hasil dari pengujian *black box* dari sistem ini yang dapat dilihat pada tabel 4.6.

Tabel 4.6. Pengujian Black Box

| Aktor | Halaman | Detail Pengujian | Pengujian | |
|-------|----------------------|---------------------------------------------------|-----------|-------|
| | | | Berhasil | Gagal |
| Admin | Login | Melakukan proses login | 10/10 | |
| | | Fungsi session | 10/10 | |
| | Dashboard | Menampilkan google maps wisata di kota yogyakarta | 10/10 | |
| | | Menampilkan data statistik | 10/10 | |
| | About | Menampilkan informasi mengenai sistem | 10/10 | |
| | | Menampilkan data wisata | 10/10 | |
| | Data Wisata | Menambah data wisata | 10/10 | |
| | | Mengedit data wisata | 10/10 | |
| | | Menghapus data wisata | 10/10 | |
| | | Menampilkan detail data wisata | 10/10 | |
| | | Fungsi pencarian dan pengurutan | 10/10 | |
| | Data Kategori Wisata | Menampilkan data kategori wisata | 10/10 | |

Tabel 4.7. Lanjutan Pengujian Black Box

| Aktor | Halaman | Detail Pengujian | Pengujian | |
|-------|----------------------------|-----------------------------------------------------------------------------------------------|-----------|-------|
| | | | Berhasil | Gagal |
| Admin | Data Kategori Wisata | Menambah data kategori wisata | 10/10 | |
| | | Mengedit data kategori wisata | 10/10 | |
| | | Menghapus data kategori wisata | 10/10 | |
| | | Fungsi pencarian dan pengurutan | 10/10 | |
| | Data Training | Menampilkan data training | 10/10 | |
| | | Menampilkan detail data training | 10/10 | |
| | | Fungsi pencarian dan pengurutan | 10/10 | |
| | Data Sentimen | Menampilkan data sentimen | 10/10 | |
| | | Menambah data sentimen | 10/10 | |
| | | Fungsi pencarian dan pengurutan | 10/10 | |
| | Halaman Analisis | Melakukan proses klasifikasi | 10/10 | |
| | | Menampilkan hasil proses klasifikasi | 10/10 | |
| | Riwayat Analisis | Menampilkan data hasil riwayat analisis | 10/10 | |
| | | Menampilkan detail dari hasil analisis | 10/10 | |
| User | Grafik Seleksi Fitur | Menampilkan grafik data seleksi fitur | 10/10 | |
| | Grafik Sentimen | Menampilkan grafik data sentimen | 10/10 | |
| | Grafik Pengujian | Menampilkan grafik data pengujian | 10/10 | |
| | Grafik Pengujian Rata-Rata | Menampilkan grafik data pengujian rata-rata | 10/10 | |
| | Home | Menampilkan daftar menu | 10/10 | |
| | | Fungsi pencarian top rekomendasi | 10/10 | |
| | Top Rekomendasi | Menampilkan daftar top rekomendasi wisata di kota yogyakarta | 10/10 | |
| | | Menampilkan daftar top rekomendasi wisata di kota yogyakarta berdasarkan kategori | 10/10 | |
| | | Menampilkan daftar top rekomendasi wisata di kota yogyakarta berdasarkan kata kunci pencarian | 10/10 | |
| User | Destinasi Wisata | Menampilkan daftar destinasi wisata di kota yogyakarta | 10/10 | |
| | View Details | Menampilkan informasi detail data wisata | 10/10 | |
| | About | Menampilkan informasi sistem | 10/10 | |
| | Contact | Menampilkan informasi kontak | 10/10 | |

4.3. Pembahasan

Berdasarkan hasil dari implementasi dan pengujian pada penelitian ini, didapatkan bahwa analisis sentimen ulasan dari *google maps* telah menghasilkan rekomendasi tempat pariwisata yang ada di Kota Yogyakarta, dengan jumlah tempat pariwisata sebanyak 27 tempat pariwisata, yang telah diurutkan dari tempat wisata dengan peringkat tertinggi ke terendah berdasarkan dari perhitungan pembobotan banyaknya jumlah sentimen positif pada masing-masing tempat pariwisata dari hasil klasifikasi.

Dari pengujian model yang telah dilakukan, tingkat performansi dari model klasifikasi *support vector machine* dengan seleksi fitur *mutual information* yang telah dibangun menghasilkan rata-rata akurasi sebesar 90.3%, rata-rata presisi sebesar 90.2% dan rata-rata *recall* sebesar 90.3%. Sedangkan, tingkat performansi dari model klasifikasi *support vector machine* dengan seleksi fitur *chi square* yang telah dibangun menghasilkan rata-rata akurasi sebesar 90.19%, rata-rata presisi sebesar 89.93%, dan rata-rata *recall* sebesar 90.19%. Dari hasil pengujian model, dapat dilihat bahwa penerapan seleksi fitur

mutual information pada metode *support vector machine* memiliki tingkat akurasi yang lebih unggul sebesar 0.11% dibandingkan dengan penerapan seleksi fitur *chi square* pada metode *support vector machine* ketika menggunakan jumlah fitur sebanyak 2400 fitur.

Dari tingkat performansi yang dihasilkan maka model klasifikasi yang dibangun oleh metode *support vector machine* dengan seleksi fitur *mutual information* yang digunakan untuk analisis sentimen ulasan di *google maps* sehingga menghasilkan rekomendasi tempat pariwisata di Kota Yogyakarta dengan hasil yang baik.

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan, maka diperoleh hasil kesimpulan sebagai berikut :

1. Pada penelitian ini berhasil membangun sistem analisis sentimen ulasan di *Google Maps* menggunakan metode *Support Vector Machine* dengan membandingkan seleksi fitur *Mutual Information* dan *Chi Square* sehingga diperoleh rekomendasi destinasi tempat pariwisata di Kota Yogyakarta dengan total tempat pariwisata sebanyak 27 yang telah diurutkan berdasarkan pembobotan jumlah sentimen positif wisata tersebut.
2. Hasil pengujian model klasifikasi dari metode *Support Vector Machine* dengan seleksi fitur *Mutual Information* dan metode *Support Vector Machine* dengan seleksi fitur *Chi Square* dimana menggunakan 2700 data ulasan yang dibagi menjadi data latih dan data uji menggunakan metode *k-fold cross validation* dengan $k=7$, memperlihatkan bahwa seleksi fitur *Mutual Information* memiliki hasil akurasi lebih baik sebesar 90.3% dengan presisi sebesar 90.2% dan *recall* sebesar 90.3% dibandingkan seleksi fitur *Chi Square* yang hanya menghasilkan akurasi sebesar 90.19% dengan presisi sebesar 89.93% dan *recall* sebesar 90.19%.

5.2 Saran

Adapun beberapa saran untuk penelitian selanjutnya berdasarkan kekurangan dan keterbatasan pada penelitian ini, antara lain :

1. Data tempat pariwisata yang akan dianalisis diperluas tidak hanya di Kota Yogyakarta saja.
2. Dapat menambahkan klasifikasi untuk kategori dari ulasan tidak hanya mengklasifikasikan sentimen pada ulasan saja. Misalnya mengkategorikan ulasan berdasarkan topik yang diulas seperti kebersihan, aksesibilitas, daya tarik, fasilitas dan sebagainya.
3. Dapat menambahkan data *training* ulasan tempat pariwisata dari sumber lain tidak hanya dari *google maps* saja, sehingga model klasifikasi yang dibangun dapat melakukan klasifikasi ulasan tempat wisata dengan lebih baik.

DAFTAR PUSTAKA

- Adriani, M., Asian, J., Nazief, B., Tahaghoghi, S. M., & Williams, H. E. (2007). Stemming Indonesian: A confix-stripping approach. *ACM Transactions on Asian Language Information Processing (TALIP)*, 6(4), 1–33.
- Azmi, M., Huda, A. K., & Setyanto, A. (2020). PEMANFAATAN DATA INSTAGRAM UNTUK MENGETAHUI REPUTASI TEMPAT WISATA DI LOMBOK. *TEKNIMEDIA: Teknologi Informasi Dan Multimedia*, 1(1), 39–46.
- Berry, M. W., & Kogan, J. (2010). *Text mining: Applications and theory*. John Wiley & Sons.
- Budi, I., & Aji, R. F. (2006). Efektifitas Seleksi Fitur dalam Sistem Temu-Kembali Informasi. *Seminar Nasional Aplikasi Teknologi Informasi (SNATI)*.
- Durgesh, K. S., & Lekha, B. (2010). Data classification using support vector machine. *Journal of Theoretical and Applied Information Technology*, 12(1), 1–7.
- Fransiska, S., Rianto, R., & Gufroni, A. I. (2020). Sentiment Analysis Provider By. U on Google Play Store Reviews with TF-IDF and Support Vector Machine (SVM) Method. *Scientific Journal of Informatics*, 7(2), 203–212.
- Gurusamy, V., & Kannan, S. (2014, October 9). *Preprocessing Techniques for Text Mining*.
- Han, J., & Kamber, M. (2006). Classification and prediction. *Data Mining: Concepts and Techniques*, 347–350.
- Handhayani, T., & Hiryanto, L. (2017). Predicting and Analyzing the Length of Study-Time Case Study: Computer Science Students. *ComTech: Computer, Mathematics and Engineering Applications*, 8(2), 107–114.
- Haq, F. U. (2020). PENGGUNAAN GOOGLE REVIEW SEBAGAI PENILAIAN KEPUASAN PENGUNJUNG DALAM PARIWISATA. *Tornare*, 2(1), 10–12.

- Harahap, N., & Lubis, S. D. (2019). *Metodologi Penelitian Kuantitatif*.
- Hasanah, R., & Yustanti, W. (2021). Analisis Sentimen dan Pemeringkatan Popularitas Tempat Wisata dengan Naïve Bayes dan AHP. *Journal of Emerging Information System and Business Intelligence (JEISBI)*, 2(1), 21–27.
- Indriani, A. (2014). Klasifikasi Data Forum dengan menggunakan Metode Naïve Bayes Classifier. *Seminar Nasional Aplikasi Teknologi Informasi (SNATI)*, 1.
- Ipmawati, J. (2016). Komparasi Teknik Klasifikasi Teks Mining Pada Analisis Sentimen. *IJNS-Indonesian Journal on Networking and Security*, 6(1).
- Irham, L. G., Adiwijaya, A., & Wisesty, U. N. (2019). Klasifikasi Berita Bahasa Indonesia Menggunakan Mutual Information dan Support Vector Machine. *JURNAL MEDIA INFORMATIKA BUDIDARMA*, 3(4), 284–292.
- Latham, P. E., & Roudi, Y. (2009). Mutual information. *Scholarpedia*, 4(1), 1658.
- Ling, J., Kencana, I. P. E. N., & Oka, T. B. (2014). Analisis Sentimen Menggunakan Metode Naïve Bayes Classifier Dengan Seleksi Fitur Chi Square. *E-Jurnal Matematika*, 3(3), 92–99.
- Mitchell, R. (2018). *Web scraping with Python: Collecting more data from the modern web*. O'Reilly Media, Inc.
- Murnawan, M. (2017). Pemanfaatan Analisis Sentimen Untuk Pemeringkatan Popularitas Tujuan Wisata. *Jurnal Penelitian Pos Dan Informatika*, 7(2), 109–120.
- Mustaqhfiri, M., Abidin, Z., & Kusumawati, R. (2012). Peringkasan teks otomatis berita berbahasa Indonesia menggunakan metode Maximum Marginal Relevance. *MATICS*.
- Negara, A. B. P., Muhardi, H., & Putri, I. M. (2020). Analisis Sentimen Maskapai Penerbangan Menggunakan Metode Naive Bayes dan Seleksi Fitur Information Gain. *Jurnal Teknologi Informasi Dan Ilmu Komputer (JTIIK)*, 7(3).

- Noehilasari, S. A. A. (2014). *PERIODISASI DAN PROSES PEMBENTUKAN KOSAKATA BAHASA GAUL TAHUN 1990-2012* [Skripsi, Universitas Negeri Yogyakarta].
<https://eprints.uny.ac.id/17951/>
- Nugroho, A. S., Witarto, A. B., & Handoko, D. (2003). Support vector machine. *Proceeding Indones. Sci. Meeiting Cent. Japan.*
- Nurajijah, N., & Riana, D. (2019). Algoritma naïve Bayes, decision tree, dan SVM untuk klasifikasi persetujuan pembiayaan nasabah koperasi syariah. *Jurnal Teknologi Dan Sistem Komputer*, 7(2), 77–82.
- Ogedebe, P. M., & Jacob, B. P. (2012). *Software Prototyping: A Strategy to Use When User Lacks Data Processing Experience.*
- Pratama, E. E., & Trilaksono, B. R. (2015). Klasifikasi Topik Keluhan Pelanggan Berdasarkan Tweet dengan Menggunakan Penggabungan Feature Hasil Ekstraksi pada Metode Support Vector Machine (SVM). *Jurnal Edukasi Dan Penelitian Informatika (JEPIN)*, 1(2), 53–59.
- Pulungan, A. F. (2019). *Analisis Kinerja Bray Curtis Distance, Canberra Distance dan Euclidean Distance pada Algoritma K-Nearest Neighbor.*
- Putu, N. L. P. M., & Amrullah, A. Z. (2021). Analisis Sentimen dan Pemodelan Topik Pariwisata Lombok Menggunakan Algoritma Naive Bayes dan Latent Dirichlet Allocation. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, 5(1), 123–131.
- Satria, D., & Mushthofa, M. (2013). Perbandingan Metode Ekstraksi Ciri Histogram dan PCA untuk Mendekripsi Stoma pada Citra Penampang Daun Freycinetia. *Jurnal Ilmu Komputer Dan Agri-Informatika*, 2(1), 20–28.
- Sembiring, K. (2007). *Modul SVM.*

- Simatupang, M. P., & Utomo, D. P. (2019). Analisa Testimonial Dengan Menggunakan Algoritma Text Mining Dan Term Frequency-Inverse Document Frequency (Tf-Idf) Pada Toko Allmeeart. *KOMIK (Konferensi Nasional Teknologi Informasi Dan Komputer)*, 3(1).
- Somantri, O., & Dairoh, D. (n.d.). Analisis Sentimen Penilaian Tempat Tujuan Wisata Kota Tegal Berbasis Text Mining. *Jurnal Edukasi Dan Penelitian Informatika (JEPIN)*, 5(2), 191–196.
- Sulaiman, M. A., & Labadin, J. (2015). Feature selection based on mutual information. *2015 9th International Conference on IT in Asia (CITA)*, 1–6.
- Sutoyo, E., & Almaarif, A. (2020). Twitter sentiment analysis of the relocation of Indonesia's capital city. *Bulletin of Electrical Engineering and Informatics*, 9(4), 1620–1630.
- Tala, F. (2003). *A study of stemming effects on information retrieval in Bahasa Indonesia*.
- Thaseen, I. S., & Kumar, C. A. (2017). Intrusion detection model using fusion of chi-square feature selection and multi class SVM. *Journal of King Saud University-Computer and Information Sciences*, 29(4), 462–472.
- Tineges, R., Triayudi, A., & Sholihat, I. D. (2020). Analisis Sentimen Terhadap Layanan Indihome Berdasarkan Twitter Dengan Metode Klasifikasi Support Vector Machine (SVM). *JURNAL MEDIA INFORMATIKA BUDIDARMA*, 4(3), 650–658.
- Turland, M. (2010). *Php*. Marco Tabini & Associates, Inc.
- Ukhti Ikhsani Larasati, I. U., Much Aziz Muslim, I. U., Riza Arifudin, I. U., & Alamsyah, I. U. (2019). Improve the Accuracy of Support Vector Machine Using Chi Square Statistic and Term Frequency Inverse Document Frequency on Movie Review Sentiment Analysis. *Scientific Journal of Informatics*, 6(1), 138–149.

- Ulfa, M. A., Irmawati, B., & Husodo, A. Y. (2018). Twitter Sentiment Analysis using Naive Bayes Classifier with Mutual Information Feature Selection. *Journal of Computer Science and Informatics Engineering (J-Cosine)*, 2(2), 106–111.
- Utama, H. S., Rosiyadi, D., Prakoso, B. S., & Ariadarma, D. (2019). Analisis Sentimen Sistem Ganjil Genap di Tol Bekasi Menggunakan Algoritma Support Vector Machine. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, 3(2), 243–250.
- Vargiu, E., & Urru, M. (2013). Exploiting web scraping in a collaborative filtering-based approach to web advertising. *Artif. Intell. Research*, 2(1), 44–54.
- Wahyudi, D., Susyanto, T., & Nugroho, D. (2017). Implementasi dan analisis algoritma stemming nazief & adriani dan porter pada dokumen berbahasa indonesia. *Jurnal Ilmiah SINUS*, 15(2), 49–56.
- Wijayanto, A. (2009). *Uji Chi-Square*.

LAMPIRAN

Lampiran A Tabel Menentukan Banyaknya Fitur Yang Akan Digunakan Berdasarkan Hasil Rata-Rata Akurasi Tertinggi Pengujian Penggunaan 100 Fitur Sampai 2650 Fitur (Semua Fitur).

| Jumlah Fitur | Fold | Akurasi Chi Square | Akurasi Mutual Information | Rata-Rata Akurasi Chi Square | Rata-Rata Akurasi Mutual Information | Akurasi Tertinggi |
|--------------|------|--------------------|----------------------------|------------------------------|--------------------------------------|-------------------|
| 100 | 1 | 0.89896373 | 0.89896373 | 89.8157209 | 89.7414133 | 89.8157209 |
| | 2 | 0.94559585 | 0.94559585 | | | |
| | 3 | 0.83937824 | 0.83937824 | | | |
| | 4 | 0.90414508 | 0.90414508 | | | |
| | 5 | 0.87823834 | 0.88082902 | | | |
| | 6 | 0.91948052 | 0.91168831 | | | |
| | 7 | 0.9012987 | 0.9012987 | | | |
| 200 | 1 | 0.89896373 | 0.9015544 | 89.7787113 | 89.7417017 | 89.7787113 |
| | 2 | 0.94559585 | 0.94559585 | | | |
| | 3 | 0.83937824 | 0.83937824 | | | |
| | 4 | 0.9015544 | 0.90414508 | | | |
| | 5 | 0.87823834 | 0.87046632 | | | |
| | 6 | 0.91948052 | 0.91688312 | | | |
| | 7 | 0.9012987 | 0.9038961 | | | |
| 300 | 1 | 0.89896373 | 0.9015544 | 89.8157209 | 89.8157209 | 89.8157209 |
| | 2 | 0.94818653 | 0.94559585 | | | |
| | 3 | 0.83937824 | 0.83937824 | | | |
| | 4 | 0.9015544 | 0.90414508 | | | |
| | 5 | 0.87823834 | 0.87564767 | | | |
| | 6 | 0.91688312 | 0.91688312 | | | |
| | 7 | 0.9038961 | 0.9038961 | | | |
| 400 | 1 | 0.9015544 | 0.90414508 | 89.8898363 | 89.8898363 | 89.8898363 |
| | 2 | 0.94818653 | 0.94559585 | | | |
| | 3 | 0.84196891 | 0.84196891 | | | |
| | 4 | 0.9015544 | 0.89896373 | | | |
| | 5 | 0.87564767 | 0.87823834 | | | |
| | 6 | 0.91948052 | 0.91948052 | | | |
| | 7 | 0.9038961 | 0.9038961 | | | |
| 500 | 1 | 0.89637306 | 0.90673575 | 89.8528267 | 89.9268459 | 89.9268459 |
| | 2 | 0.94559585 | 0.94300518 | | | |
| | 3 | 0.84196891 | 0.84196891 | | | |
| | 4 | 0.9015544 | 0.89896373 | | | |
| | 5 | 0.88082902 | 0.88082902 | | | |
| | 6 | 0.91948052 | 0.91948052 | | | |
| | 7 | 0.9038961 | 0.9038961 | | | |
| 600 | 1 | 0.89896373 | 0.90673575 | 89.8157209 | 89.8898363 | 89.8898363 |
| | 2 | 0.94818653 | 0.94041451 | | | |
| | 3 | 0.84196891 | 0.84196891 | | | |
| | 4 | 0.9015544 | 0.89896373 | | | |
| | 5 | 0.87564767 | 0.88082902 | | | |
| | 6 | 0.91948052 | 0.91948052 | | | |
| | 7 | 0.9012987 | 0.9038961 | | | |
| 700 | 1 | 0.9015544 | 0.90673575 | 90.0008652 | 90.0378748 | 90.0378748 |
| | 2 | 0.94818653 | 0.94300518 | | | |
| | 3 | 0.84715026 | 0.84196891 | | | |
| | 4 | 0.90414508 | 0.89896373 | | | |
| | 5 | 0.87564767 | 0.88860104 | | | |
| | 6 | 0.91948052 | 0.91948052 | | | |

| | | | | | | |
|------|---|------------|------------|-------------------|-------------------|-------------------|
| | 7 | 0.9038961 | 0.9038961 | | | |
| 800 | 1 | 0.89896373 | 0.90932642 | 89.9268459 | 90.0378748 | 90.0378748 |
| | 2 | 0.94818653 | 0.94041451 | | | |
| | 3 | 0.84455959 | 0.84196891 | | | |
| | 4 | 0.9015544 | 0.89896373 | | | |
| | 5 | 0.87823834 | 0.88860104 | | | |
| | 6 | 0.91948052 | 0.91948052 | | | |
| | 7 | 0.9038961 | 0.9038961 | | | |
| 900 | 1 | 0.89896373 | 0.90932642 | 89.9638555 | 90.0379709 | 90.0379709 |
| | 2 | 0.94818653 | 0.94041451 | | | |
| | 3 | 0.84715026 | 0.84196891 | | | |
| | 4 | 0.9015544 | 0.9015544 | | | |
| | 5 | 0.87823834 | 0.88341969 | | | |
| | 6 | 0.91948052 | 0.91948052 | | | |
| | 7 | 0.9038961 | 0.90649351 | | | |
| 1000 | 1 | 0.9015544 | 0.90673575 | 90.0749805 | 90.0749805 | 90.0749805 |
| | 2 | 0.94818653 | 0.94041451 | | | |
| | 3 | 0.84455959 | 0.84455959 | | | |
| | 4 | 0.90414508 | 0.9015544 | | | |
| | 5 | 0.88082902 | 0.88601036 | | | |
| | 6 | 0.91948052 | 0.91948052 | | | |
| | 7 | 0.90649351 | 0.90649351 | | | |
| 1100 | 1 | 0.90414508 | 0.90673575 | 89.9637594 | 90.1119902 | 90.1119902 |
| | 2 | 0.94559585 | 0.94041451 | | | |
| | 3 | 0.84455959 | 0.84455959 | | | |
| | 4 | 0.89896373 | 0.90414508 | | | |
| | 5 | 0.88341969 | 0.88601036 | | | |
| | 6 | 0.91688312 | 0.91948052 | | | |
| | 7 | 0.9038961 | 0.90649351 | | | |
| 1200 | 1 | 0.9015544 | 0.90673575 | 89.9638555 | 90.0379709 | 90.0379709 |
| | 2 | 0.94559585 | 0.93782383 | | | |
| | 3 | 0.84455959 | 0.84455959 | | | |
| | 4 | 0.89896373 | 0.90414508 | | | |
| | 5 | 0.88341969 | 0.88341969 | | | |
| | 6 | 0.91688312 | 0.91688312 | | | |
| | 7 | 0.90649351 | 0.90909091 | | | |
| 1300 | 1 | 0.90414508 | 0.90673575 | 90.0378748 | 90.0749805 | 90.0749805 |
| | 2 | 0.94559585 | 0.93782383 | | | |
| | 3 | 0.84455959 | 0.84715026 | | | |
| | 4 | 0.9015544 | 0.90414508 | | | |
| | 5 | 0.88341969 | 0.88341969 | | | |
| | 6 | 0.91688312 | 0.91688312 | | | |
| | 7 | 0.90649351 | 0.90909091 | | | |
| 1400 | 1 | 0.90414508 | 0.90414508 | 90.0378748 | 90.0379709 | 90.0379709 |
| | 2 | 0.94300518 | 0.93782383 | | | |
| | 3 | 0.84455959 | 0.84715026 | | | |
| | 4 | 0.9015544 | 0.9015544 | | | |
| | 5 | 0.88601036 | 0.88601036 | | | |
| | 6 | 0.91688312 | 0.91688312 | | | |
| | 7 | 0.90649351 | 0.90909091 | | | |
| 1500 | 1 | 0.90414508 | 0.90414508 | 90.1119902 | 90.0379709 | 90.1119902 |
| | 2 | 0.94300518 | 0.93782383 | | | |
| | 3 | 0.84715026 | 0.84715026 | | | |
| | 4 | 0.9015544 | 0.9015544 | | | |
| | 5 | 0.88601036 | 0.88601036 | | | |
| | 6 | 0.91688312 | 0.91688312 | | | |
| | 7 | 0.90909091 | 0.90909091 | | | |

| | | | | | | |
|------|---|------------|------------|-------------------|-------------------|-------------------|
| 1600 | 1 | 0.90414508 | 0.90414508 | 90.0749805 | 90.0009613 | 90.0749805 |
| | 2 | 0.94300518 | 0.93782383 | | | |
| | 3 | 0.84715026 | 0.84715026 | | | |
| | 4 | 0.9015544 | 0.89896373 | | | |
| | 5 | 0.88341969 | 0.88601036 | | | |
| | 6 | 0.91688312 | 0.91688312 | | | |
| | 7 | 0.90909091 | 0.90909091 | | | |
| 1700 | 1 | 0.90414508 | 0.90414508 | 90.0749805 | 90.0379709 | 90.0749805 |
| | 2 | 0.94041451 | 0.93782383 | | | |
| | 3 | 0.84715026 | 0.84715026 | | | |
| | 4 | 0.9015544 | 0.89896373 | | | |
| | 5 | 0.88601036 | 0.88860104 | | | |
| | 6 | 0.91688312 | 0.91688312 | | | |
| | 7 | 0.90909091 | 0.90909091 | | | |
| 1800 | 1 | 0.90414508 | 0.90414508 | 90.1119902 | 90.1119902 | 90.1119902 |
| | 2 | 0.94041451 | 0.94041451 | | | |
| | 3 | 0.84715026 | 0.84715026 | | | |
| | 4 | 0.90414508 | 0.9015544 | | | |
| | 5 | 0.88601036 | 0.88860104 | | | |
| | 6 | 0.91688312 | 0.91688312 | | | |
| | 7 | 0.90909091 | 0.90909091 | | | |
| 1900 | 1 | 0.90414508 | 0.90414508 | 90.0749805 | 90.0748844 | 90.0749805 |
| | 2 | 0.93782383 | 0.94041451 | | | |
| | 3 | 0.84715026 | 0.84715026 | | | |
| | 4 | 0.90414508 | 0.9015544 | | | |
| | 5 | 0.88601036 | 0.88860104 | | | |
| | 6 | 0.91688312 | 0.91688312 | | | |
| | 7 | 0.90909091 | 0.90649351 | | | |
| 2000 | 1 | 0.90414508 | 0.90414508 | 90.038067 | 90.0378748 | 90.038067 |
| | 2 | 0.93782383 | 0.94041451 | | | |
| | 3 | 0.84715026 | 0.84715026 | | | |
| | 4 | 0.89896373 | 0.89896373 | | | |
| | 5 | 0.88601036 | 0.88860104 | | | |
| | 6 | 0.91948052 | 0.91688312 | | | |
| | 7 | 0.90909091 | 0.90649351 | | | |
| 2100 | 1 | 0.90414508 | 0.90414508 | 90.0750767 | 90.1119902 | 90.1119902 |
| | 2 | 0.93782383 | 0.94041451 | | | |
| | 3 | 0.84715026 | 0.84715026 | | | |
| | 4 | 0.90414508 | 0.9015544 | | | |
| | 5 | 0.88341969 | 0.88860104 | | | |
| | 6 | 0.91948052 | 0.91688312 | | | |
| | 7 | 0.90909091 | 0.90909091 | | | |
| 2200 | 1 | 0.90414508 | 0.90414508 | 90.1120863 | 90.0749805 | 90.1120863 |
| | 2 | 0.93782383 | 0.94041451 | | | |
| | 3 | 0.84715026 | 0.84715026 | | | |
| | 4 | 0.90414508 | 0.89896373 | | | |
| | 5 | 0.88601036 | 0.88860104 | | | |
| | 6 | 0.91948052 | 0.91688312 | | | |
| | 7 | 0.90909091 | 0.90909091 | | | |
| 2300 | 1 | 0.90414508 | 0.90414508 | 90.1490959 | 90.2231152 | 90.2231152 |
| | 2 | 0.93782383 | 0.94041451 | | | |
| | 3 | 0.84715026 | 0.84974093 | | | |
| | 4 | 0.90414508 | 0.9015544 | | | |
| | 5 | 0.88860104 | 0.89119171 | | | |
| | 6 | 0.92207792 | 0.91948052 | | | |
| | 7 | 0.90649351 | 0.90909091 | | | |
| 2400 | 1 | 0.90414508 | 0.90414508 | 90.1861055 | 90.2972305 | 90.2972305 |

| | | | | | | |
|--------------------------------------------|---|------------|------------|-------------------|-------------------|-------------------|
| | 2 | 0.93782383 | 0.94041451 | | | |
| | 3 | 0.84715026 | 0.84974093 | | | |
| | 4 | 0.9015544 | 0.9015544 | | | |
| | 5 | 0.89378238 | 0.89378238 | | | |
| | 6 | 0.92207792 | 0.92207792 | | | |
| | 7 | 0.90649351 | 0.90909091 | | | |
| 2500 | 1 | 0.90414508 | 0.90414508 | 90.2602209 | 90.2602209 | 90.2602209 |
| | 2 | 0.94041451 | 0.94041451 | | | |
| | 3 | 0.84715026 | 0.84715026 | | | |
| | 4 | 0.9015544 | 0.9015544 | | | |
| | 5 | 0.89378238 | 0.89378238 | | | |
| | 6 | 0.92207792 | 0.92207792 | | | |
| | 7 | 0.90909091 | 0.90909091 | | | |
| 2600 | 1 | 0.90414508 | 0.90414508 | 90.2602209 | 90.2602209 | 90.2602209 |
| | 2 | 0.94041451 | 0.94041451 | | | |
| | 3 | 0.84715026 | 0.84715026 | | | |
| | 4 | 0.9015544 | 0.9015544 | | | |
| | 5 | 0.89378238 | 0.89378238 | | | |
| | 6 | 0.92207792 | 0.92207792 | | | |
| | 7 | 0.90909091 | 0.90909091 | | | |
| 2650 (semua fitur) | 1 | 0.90414508 | 0.90414508 | 90.2602209 | 90.2602209 | 90.2602209 |
| | 2 | 0.94041451 | 0.94041451 | | | |
| | 3 | 0.84715026 | 0.84715026 | | | |
| | 4 | 0.9015544 | 0.9015544 | | | |
| | 5 | 0.89378238 | 0.89378238 | | | |
| | 6 | 0.92207792 | 0.92207792 | | | |
| | 7 | 0.90909091 | 0.90909091 | | | |
| Rata-rata akurasi tertinggi sebesar | | | | | 90.2972305 | |

Lampiran B Contoh Data Labeling Ulasan

| Data Ulasan Ke- | Nama Responden |
|-----------------|----------------------------------|
| 1-600 | Nadia Mutia Khansa (R1) |
| | Moch. Rizky Bayuarga (R2) |
| | Khaeruz Zaman (R3) |
| | Selva Mahlinasha Arindani (R4) |
| | Nur Khaliza Irlansda (R5) |
| 700-1200 | Awang Suria Trisakti (R1) |
| | Yahdi Indrawan (R2) |
| | Maudia Khumaida Savitri (R3) |
| | Nandha Juniaroesita Peksi (R4) |
| | Ahmad Dzakiyyul Fuad (R5) |
| 1300-1700 | Asa Aolada Akhira (R1) |
| | Fanindita Widyantoro Putri (R2) |
| | Claudy Theresia Siregar (R3) |
| | Alis Tiyaning Tiyas (R4) |
| | Efrilia Mahdilah Nurhidayah (R5) |
| 1800-2200 | Renita Aswadi (R1) |
| | Evariyanti (R2) |
| | Crussita Devi (R3) |
| | Ninditya Salma Nur Aini (R4) |
| | Nur Isna Melati Putri (R5) |
| 2300-2700 | Niken Oktaviana (R1) |
| | Clarita Magdalena Marpaung (R2) |
| | Risma Stela Putri (R3) |
| | Muhammad Fahmi Adam (R4) |
| | Andre Saputra (R5) |

Keterangan :

1. Diberi label 1 jika merupakan sentimen positif.
2. Diberi label 0 jika merupakan sentimen negatif.

| No | Id Wisata | Id_User | Ulasan | Sentimen | | | | | |
|----|-----------|----------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------|----|----|----|----|-------|
| | | | | R1 | R2 | R3 | R4 | R5 | Hasil |
| 1 | 1 | Irfian Way | Salah satu peninggalan sejarah kerajaan nusantara yg masih terjaga hingga sekarang. Cuma syang sebagian propertinya sudah tidak lengkap. dan juga tidak semua foto di pajang hanya di taruh di etalase | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | 1 | Sunarto Antok | Tujuan wisata yg tidak boleh dilewatkan saat berwisata ke jogja... disini kita bisa tahu dan belajar sejarah dan adat serta budaya jogjakarta | 1 | 1 | 1 | 1 | 1 | 1 |
| 3 | 1 | Thomas Herdian | Bersih, Bagus. Pelayanan pemandunya sangat Mantap, Ramah2 Orang jualan di Luar | 1 | 1 | 1 | 1 | 1 | 1 |
| 4 | 1 | avri lita | tempat yg perlu d kunjungi klo ke jogja,, apalgi bawa anak usia | 1 | 1 | 1 | 1 | 1 | 1 |

| | | | | | | | | |
|----|---|---------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---|---|---|---|---|
| | | | sekolah seru dan nambah ilmu juga | | | | | |
| 5 | 1 | Isa Maulana | Tempat bersejarah, pas kesana kondisinya kotor banyak daun2 kering.. | 0 | 0 | 0 | 0 | 0 |
| 6 | 1 | Alfian B.s | Tempatnya bersih orang orangnya juga ramah, keratonnya juga megah | 1 | 1 | 1 | 1 | 1 |
| 7 | 1 | Nanda Budi | Tempatnya bersih , ketika datang langsung disambut dan dipandu mengelilingi keraton sehingga dapat menambah ilmu dan wawasan | 1 | 1 | 1 | 1 | 1 |
| 8 | 1 | Arif Nasiruddin | Suasana sekitar keraton Jogja selalu bikin kangen, saya yg tinggal di Jogja saja kadang kangen untuk sekedar jalan atau bersepeda di sekitar alun-alun. | 1 | 1 | 1 | 1 | 1 |
| 9 | 1 | Taufik Maulana | AGAK KECEWA SIH KARENA WAKTU SAYA MAMPIR KESINI KATANYA TANGGAL 8 JULI BARU DIBUKA UNTUK UMUM LAGI. YA UDAH LAH MEMANG NGGAK REJEKINYA KESITU... | 0 | 0 | 0 | 0 | 0 |
| 10 | 1 | Viedie Ananda | Tempat yang sangat penuh sejarah, setelah new normal wisata harus di pandu pemandu, | 1 | 1 | 1 | 1 | 1 |
| 11 | 1 | Nabila El Fatikha | Bagi kawan-kawan yang berkunjung ke Yogyakarta, tentu jangan lewatkan salah satu tempat penting ini. Ya.. Keraton Yogyakarta akan selalu mencuri hatimu untuk selalu datang lagi. Tempat tumbuh dan lestari budaya Jawa. | 1 | 1 | 1 | 1 | 1 |
| 12 | 1 | Qiandra Aeleasha | Tempat yang paling wajib didatangi ketika sedang berada di jogja tempat yang masih melestarikan budaya keraton dan tempat untuk foto foto rekomen deh | 1 | 1 | 1 | 1 | 1 |
| 13 | 1 | TK MULIA 1 Kaligambir | saya sering banget main ke keraton jogja.. kotanya ngangenin.. makanan murah kayak di kotaku sendiri | 1 | 1 | 1 | 1 | 1 |
| 14 | 1 | yulianita sulistyaningrum | Sekarang tempatnya sudah lebih baik, nyaman, bagus untuk liburan anak sekolah tambah pengalaman, pengetahuan belajar sejarah juga.. | 1 | 1 | 1 | 1 | 1 |
| 15 | 1 | Rini Kartikasari | Destinasi wisata wajib kalau anda datang ke Jogja suasana dan bangunan yang klasik membuat anda seakan akan berada di zaman jawa kuno | 1 | 1 | 1 | 1 | 1 |
| 16 | 1 | mfarhan pramono | Tempat yang sangat cocok saat berkunjung ke jogja karena di | 1 | 1 | 1 | 1 | 1 |

| | | | | | | | | |
|----|---|-------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---|---|---|---|---|
| | | | kraton kita dapat menambah wawasan tentang kerajaan Yogyakarta | | | | | |
| 17 | 1 | GS Adi Surabaya | Tempat leluhur suku Jawa, harus di jaga dan di lestarikan | 1 | 1 | 1 | 1 | 1 |
| 18 | 1 | Sajiman Prawiro Sasmito | Sedang direnovasi... pemandunya sopan dan ramah sekali.... | 1 | 1 | 1 | 1 | 1 |
| 19 | 1 | Diah Ayulenii | Saya datang setelah acara seni tari selesai, tempatnya hening dan tenang, mungkin saat saya kesini jamnya sangat tepat belum ada rombongan kunjungan jadi lebih leluasa untuk mempelajari sejarah maupun berfoto foto. | 1 | 1 | 1 | 1 | 1 |
| 20 | 1 | Budi Martono | Istana favorit, tempat belajar, tempat wisata, tempat menulis diary. | 1 | 1 | 1 | 1 | 1 |
| 21 | 1 | Herman Sumarno | Saat kami sekeluarga berkunjung , pemandunya ramah dan penjelasannya detail sekali, sempat mampir ke pengrajin lukisan dan batik, senang bangga walau bukan orang yogy. Yogyakarta kamu emang istimewa | 1 | 1 | 1 | 1 | 1 |
| 22 | 1 | Victricius Aditya | Keraton merupakan tempat ternyaman dan tempat indah. Wajib dikunjungi bagi wisatawan dalam negeri dan wisatawan luar negeri. Sangat mencerminkan Yogyakarta. Kerajaan Warga Yogyakarta yang indah dan menawan. | 1 | 1 | 1 | 1 | 1 |
| 23 | 1 | Nazwa fildzah | tempat yang eksotik untuk berswafoto | 1 | 1 | 1 | 1 | 1 |
| 24 | 1 | bansol tv | Sangat bersejarah, dan sangat perlu dilestarikan, semangat dan sejarah yang harus disatukan | 1 | 1 | 1 | 1 | 1 |
| 25 | 1 | Hadlir Husaeni | Banyak tempat bersejarah sebagai wahana wisata | 1 | 1 | 1 | 1 | 1 |
| 26 | 1 | Raden Tejo | Tempat bersejarah yang masih terjaga dengan baik sampai sekarang | 1 | 1 | 1 | 1 | 1 |
| 27 | 1 | Mutia Dwi Zulfana | Suasana khas keraton.. bagus menyenangkan | 1 | 1 | 1 | 1 | 1 |
| 28 | 1 | Cica I | Liburan ke Yogyakarta? Belum lengkap kalau gak mampir kesini. Jalan jalan sambil tambah ilmu juga, jadi tau banyak juga kan... Tempatnya bagus dan sangat luas | 1 | 1 | 1 | 1 | 1 |
| 29 | 1 | Freya Hutter | untuk turis disarankan jangan bawa tas dg isi yg banyak kesini, karna tempat wisata ini butuh jalan melewati rumah warga supaya bisa sampai di beberapa lokasi lain yg juga termasuk dalam wisata ini | 1 | 1 | 1 | 1 | 1 |

| | | | | | | | | | |
|----|---|-------------------------|---------------------------------------------------------------------------------------------------------|---|---|---|---|---|---|
| 30 | 1 | Agustin Purwaningsih | Tmpt untuk belajar sejarah. ada pemandu yg sangat ramah2. dg sabar siap menjawab semua pertanyaan2 kita | 1 | 1 | 1 | 1 | 1 | 1 |
|----|---|-------------------------|---------------------------------------------------------------------------------------------------------|---|---|---|---|---|---|

Hasil data labeling untuk setiap pariwisata

| Id_Wisata | Positif | Negatif | Jumlah |
|------------------|----------------|----------------|---------------|
| 1 | 90 | 10 | 100 |
| 2 | 90 | 10 | 100 |
| 3 | 79 | 21 | 100 |
| 4 | 96 | 4 | 100 |
| 5 | 94 | 6 | 100 |
| 6 | 90 | 10 | 100 |
| 7 | 99 | 1 | 100 |
| 8 | 79 | 21 | 100 |
| 9 | 88 | 12 | 100 |
| 10 | 75 | 25 | 100 |
| 11 | 83 | 17 | 100 |
| 12 | 80 | 20 | 100 |
| 13 | 90 | 10 | 100 |
| 14 | 99 | 1 | 100 |
| 15 | 85 | 15 | 100 |
| 16 | 79 | 21 | 100 |
| 17 | 76 | 24 | 100 |
| 18 | 86 | 14 | 100 |
| 19 | 92 | 8 | 100 |
| 20 | 91 | 9 | 100 |
| 21 | 90 | 10 | 100 |
| 22 | 91 | 9 | 100 |
| 23 | 86 | 14 | 100 |
| 24 | 95 | 5 | 100 |
| 25 | 75 | 25 | 100 |
| 26 | 95 | 5 | 100 |
| 27 | 93 | 7 | 100 |
| Jumlah | 2366 | 334 | 2700 |