

BAB III

ANALISIS DAN PERANCANGAN SISTEM

3.1 Analisis Sistem

Analisis sentimen, atau *opinion mining*, adalah proses otomatis untuk memahami dan menganalisis data teks guna menentukan sentimen dalam sebuah opini, baik positif maupun negatif. Dengan semakin berkembangnya teknologi, ulasan atau opini dari pengguna aplikasi dapat dengan mudah ditemukan di berbagai platform, salah satunya adalah Google Play Store. Ulasan pengguna ini mencerminkan pengalaman, kepuasan, atau kritik terhadap aplikasi tertentu, termasuk aplikasi TransJatim. Mengidentifikasi sentimen dari ulasan pengguna TransJatim menjadi penting untuk memahami persepsi masyarakat terhadap layanan yang diberikan, seperti keandalan aplikasi, fitur yang disediakan, dan kemudahan penggunaan. Dalam konteks ini, sentimen mencerminkan apakah ulasan tersebut bersifat positif, negatif, atau netral.

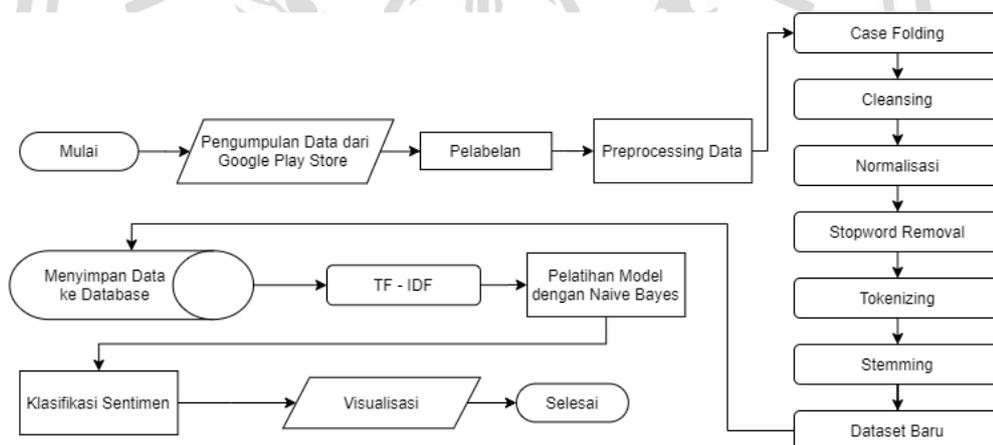
Melalui analisis sentimen, dapat diperoleh wawasan yang mendalam mengenai kekuatan dan kelemahan aplikasi TransJatim berdasarkan opini pengguna. Informasi ini dapat digunakan untuk memperbaiki layanan, meningkatkan kualitas aplikasi, dan menyusun strategi pengembangan yang lebih efektif. Dengan demikian, analisis sentimen tidak hanya bermanfaat untuk memahami opini publik tetapi juga sebagai dasar dalam pengambilan keputusan untuk meningkatkan pengalaman pengguna.

3.2 Hasil Analisis Sistem

Hasil analisis sentimen pada ulasan aplikasi TransJatim di Google Play Store dilakukan dengan metode *Naïve Bayes*. Sistem klasifikasi sentimen ini bertujuan untuk mengidentifikasi opini pengguna aplikasi TransJatim, yang didasarkan pada data ulasan di Google Play Store. Data ulasan yang digunakan telah diberi label sentimen positif, negatif, atau netral. Proses ini mencakup pengumpulan data ulasan dari pengguna untuk kemudian dijadikan data latih dalam membangun model klasifikasi.

Pengumpulan data dilakukan secara menyeluruh, menghasilkan sebanyak 459 komentar pengguna aplikasi TransJatim di Google Play Store. Setelah data diproses menggunakan metode *Naïve Bayes*, setiap ulasan diklasifikasikan ke dalam kelas sentimen positif, negatif, atau netral. Hasil klasifikasi ini memberikan informasi penting mengenai persepsi masyarakat terhadap aplikasi TransJatim. Melalui analisis ini, dapat diperoleh gambaran mengenai kekuatan dan kelemahan aplikasi berdasarkan opini pengguna, yang kemudian dapat digunakan untuk meningkatkan kualitas layanan dan pengalaman pengguna.

Sistem yang dirancang bertujuan untuk melakukan analisis sentimen terhadap ulasan aplikasi TransJatim yang tersedia di Google Play Store. Analisis ini menggunakan metode *Naïve Bayes* sebagai teknik klasifikasi sentimen untuk menentukan opini pengguna berdasarkan data komentar yang dikumpulkan. Sistem ini diharapkan dapat membantu pihak terkait, seperti pengembang aplikasi atau pemerintah, untuk memahami persepsi masyarakat terhadap layanan TransJatim. Dengan begitu, hasil analisis ini dapat digunakan untuk mendukung perbaikan layanan serta pengambilan keputusan yang lebih terarah. Deskripsi umum sistem yang akan dibangun dapat digambarkan seperti yang ditunjukkan pada Gambar 3.1.



Gambar 3. 1 *Flowchart* Sistem Analisis Sentimen Ulasan Aplikasi TransJatim di Google PlayStore

Berikut penjelasan dari *Flowchart* Sistem analisis sentiment pada ulasan Google Play Store pada Gambar 3.1:

1. Pengumpulan Data dari Google Play Store
2. Pelabelan

Ulasan data diklasifikasikan ke dalam sentimen positif, netral, atau negatif, melalui proses manual. Langkah ini bertujuan untuk menjamin bahwa setiap data memiliki penandaan sentimen yang jelas.

3. *Preprocessing* Data

Proses ini bertujuan untuk membersihkan dan mempersiapkan data teks agar dapat diproses lebih lanjut. Terdiri dari beberapa langkah:

1. *Case Folding*: Mengubah semua teks menjadi huruf kecil untuk konsistensi.
2. *Cleansing*: Menghapus karakter khusus, angka, tanda baca, atau elemen lain yang tidak relevan.
3. Normalisasi: Mengubah kata-kata tidak baku menjadi kata yang baku (misalnya, "gak" menjadi "tidak").
4. *Stopword Removal*: Menghapus kata-kata umum yang tidak memiliki makna penting (contoh: "dan", "atau").
5. *Tokenizing*: Memecah teks menjadi kata-kata atau token individual.
6. *Stemming*: Mengubah kata ke bentuk dasarnya (contoh: "berlari" menjadi "lari").
7. *Dataset Baru*: Hasil akhir *preprocessing* adalah *dataset* teks yang sudah bersih dan siap digunakan.

4. Menyimpan Data ke *Database*
5. Pelatihan Model dengan *Naïve Bayes*

Pada tahap Pelatihan Model dengan *Naïve Bayes*, data hasil *preprocessing* direpresentasikan menggunakan TF-IDF untuk mengubah teks menjadi format numerik. Model

kemudian dilatih untuk menghitung probabilitas setiap kata pada masing-masing kelas sentimen (positif, negatif, atau netral). Hasilnya adalah model yang siap digunakan untuk klasifikasi sentimen.

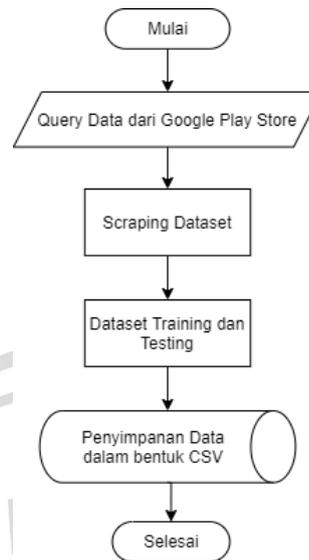
6. Klasifikasi Sentimen

Model *Naïve Bayes* digunakan untuk mengklasifikasikan sentimen data baru (apakah sentimen positif, negatif, atau netral).

7. Visualisasi

Pada tahap Visualisasi, hasil klasifikasi sentimen ditampilkan dalam bentuk grafik atau diagram. Visualisasi ini bertujuan untuk memudahkan analisis pola dan distribusi sentimen dari data yang telah diklasifikasikan. Visualisasi juga memberikan gambaran yang jelas mengenai proporsi sentimen tertentu serta mempermudah dalam mengevaluasi kualitas layanan berdasarkan tanggapan pengguna.

Selanjutnya tahap representasi *dataset* dilakukan pengumpulan data dengan cara scraping menggunakan *Python*. Data diambil dari komentar pengguna pada aplikasi TransJatim di Google Play Store. Komentar-komentar ini berisi opini pengguna terkait kualitas dan pengalaman mereka dalam menggunakan aplikasi tersebut. Data scraping dilakukan terhadap 459 komentar pengguna tanpa ada filter tertentu, sehingga seluruh komentar dimasukkan ke dalam *dataset*.



Gambar 3. 2 Alur *Scraping Dataset*

a. Mengakses Komentar di Google Play Store

Menggunakan skrip Python, dilakukan proses scraping untuk mengambil semua komentar terkait aplikasi TransJatim. Skrip ini memastikan semua data teks, seperti isi komentar, tanggal, tersimpan dengan benar.

b. Proses Penyimpanan

Semua data komentar yang berhasil dikumpulkan disimpan dalam format file CSV untuk mempermudah pengolahan data pada tahap berikutnya. Berikut adalah contoh hasil *scraping dataset* dengan menggunakan *python*.

Tabel 3. 1 Contoh Hasil Scraping Menggunakan *Python*

username	full_text
Nasrullah Mazii	Antrian gila gila an. Bukan karena transportasi andalan warga jatim. Tapi karena murah dan affordable, dan cuma karena satu satunya yg bisa dinaiki. Armada terbatas, kecil.

username	full_text
	Sementara pengguna banyak. Aplikasi tidak banyak membantu. Pelayanan publik yg tak jua memberikan pelayanan yg humanis bagi publik
putri tsaabitah	Antara aplikasi dan aktualnya tidak sinkron. Di aplikasi tertulis masih terisi 1 penumpang, kenyataannya sudah full dan tinggal yg berdiri. Untuk aplikasi transportasi publik ini sedikit mengecewakan.
edy purnomo	bagus banget.. bisa real time posisi bus dengan semua rute dan jumlah penumpang.. top markotop
Alfiyan Wildan	Tolong aplikasinya di kondisikan, jumlah penumpang suka eror, kadang tulisannya 0 tapi malah full, kadang penumpangnya dikit tapi di aplikasinya berjumlah 64 , bikin bingung
Tridya Ayu R	Aplikasi sudah cukup bagus dan membantu, saran utk skrng adalah penambahan armada dan halte krn pengguna sangat amat banyak, kasian halte2 kecil yg sudah tdk mendapatkan jatah

username	full_text
	naik, dan utk aplikasinya mohon ditambahkan fitur detail orang yg sedang menunggu di tiap2 halte, jadi spt ada aplikasi cek in - cek out di tiap halte, terimakasih 🙏🙏»

c. Pelabelan

Pelabelan dilakukan untuk mengklasifikasikan setiap komentar ke dalam kategori positif, netral, atau negatif. Setiap komentar diberi label berdasarkan kandungan emosional atau opini yang terkandung di dalamnya, sehingga data dapat digunakan untuk proses analisis lebih lanjut.

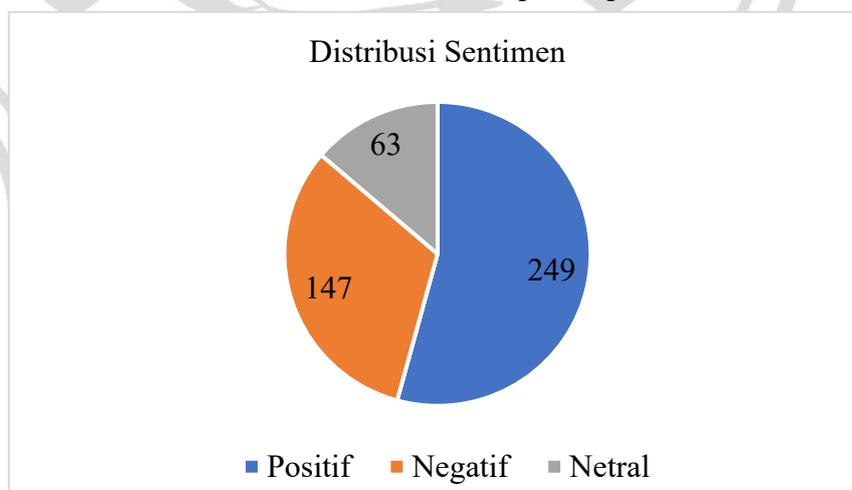
Tabel 3. 2 Hasil Pelabelan Awal

Username	Full_text	Pelabelan_awal
Nasrullah Mazii	Antrian gila gila an. Bukan karena transportasi andalan warga jatim. Tapi karena murah dan affordable, dan cuma karena satu satunya yg bisa dinaiki. Armada terbatas, kecil. Sementara pengguna banyak. Aplikasi tidak banyak membantu. Pelayanan publik yg tak jua memberikan	Negatif

Username	Full_text	Pelabelan_awal
	pelayanan yg humanis bagi publik	
putri tsaabitah	Antara aplikasi dan aktualnya tidak sinkron. Di aplikasi tertulis masih terisi 1 penumpang, kenyataannya sudah full dan tinggal yg berdiri. Untuk aplikasi transportasi publik ini sedikit mengecewakan.	Negatif
edy purnomo	bagus banget.. bisa real time posisi bus dengan semua rute dan jumlah penumpang.. top markotop	Positif
Alfiyan Wildan	Tolong aplikasinya di kondisikan, jumlah penumpang suka eror, kadang tulisannya 0 tapi malah full, kadang penumpangnya dikit tapi di aplikasinya berjumlah 64 , bikin bingung	Negatif
Tridya Ayu R	Aplikasi sudah cukup bagus dan membantu, saran utk skrng adalah	Positif

Username	Full_text	Pelabelan_awal
	penambahan armada dan halte krn pengguna sangat amat banyak, kasian halte2 kecil yg sudah tdk mendapatkan jatah naik, dan utk aplikasinya mohon ditambahkan fitur detail orang yg sedang menunggu di tiap2 halte, jadi spt ada aplikasi cek in - cek out di tiap halte, terimakasih ðŸ™¼ðŸŒŸ»	

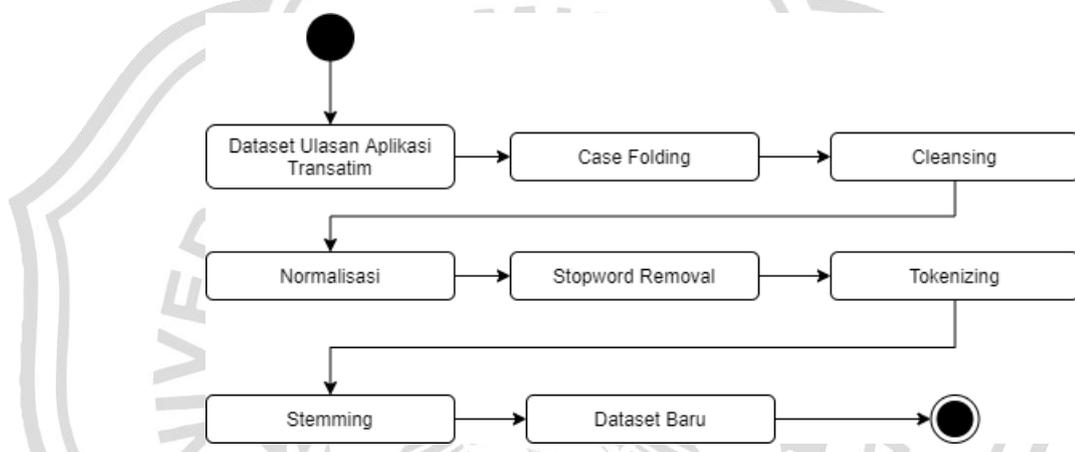
Pada tahap analisis data, setiap komentar dianalisis untuk mengidentifikasi sentimennya. Sentimen tersebut kemudian diklasifikasikan ke dalam tiga kategori, positif, netral, dan negatif. Distribusi klasifikasi sentimen ini ditampilkan pada Gambar 3.3.



Gambar 3.3 Distribusi Sentimen Terhadap Aplikasi TransJatim

1. *Preprocessing*

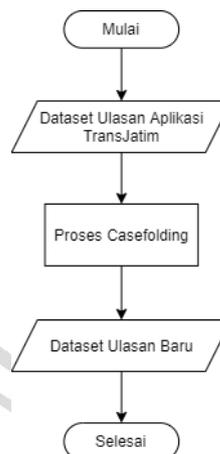
Pada tahap *preprocessing* sangat penting untuk mendukung langkah-langkah selanjutnya, terutama dalam mengurangi atribut-atribut yang kurang relevan dalam proses klasifikasi. Pada tahap ini, data yang digunakan masih berupa data mentah yang belum terolah. *Preprocessing* bertujuan untuk menghasilkan data yang lebih berkualitas sehingga proses klasifikasi dapat dilakukan dengan lebih mudah. Proses *preprocessing* ini melibatkan beberapa langkah yang dijelaskan pada Gambar 3.4 berikut.



Gambar 3. 4 Alur *preprocessing*

a. *Casefolding*

Pada tahap *casefolding*, semua huruf kapital dalam ulasan pengguna aplikasi TransJatim diubah menjadi huruf kecil. Proses ini bertujuan untuk menyeragamkan teks sehingga tidak ada perbedaan antara huruf besar dan kecil selama analisis. Diagram alur untuk proses case folding dapat dilihat pada Gambar 3.5.



Gambar 3. 5 Alur Proses *Casefolding*

Sebagai ilustrasi dari proses *cleansing*, contoh ulasan pengguna aplikasi TransJatim yang telah melalui tahap ini ditampilkan pada Tabel 3.3.

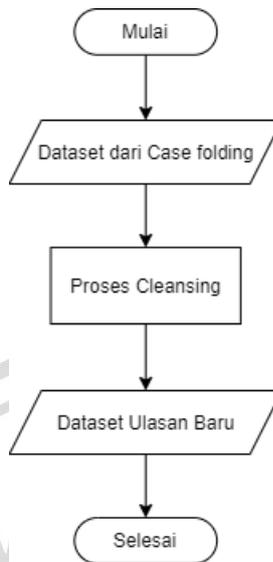
Tabel 3. 3 Contoh Hasil Proses *Casefolding*

Sebelum	Sesudah
Untuk GRESIK-LAMONGAN harap di tambah lagi halte nya. Jarak penempatan halte Dari sekapuk ke dalegan cukup jauh. Kami masyarakat yg tinggal sekitar desa Wotan,petung sampai ke selatan cukup keberatan	untuk gresik-lamongan harap di tambah lagi halte nya. jarak penempatan halte dari sekapuk ke dalegan cukup jauh kami masyarakat yg tinggal sekitar desa wotanpetung sampai ke selatan cukup keberatan
Antrian gila gila an. Bukan karena transportasi andalan warga jatim. Tapi karena murah dan affordable, dan cuma karena satu satunya yg bisa dinaiki. Armada terbatas, kecil. Sementara	antrian gila gila an. bukan karena transportasi andalan warga jatim. tapi karena murah dan affordable, dan cuma karena satu satunya yg bisa dinaiki. armada terbatas, kecil. sementara

Sebelum	Sesudah
pengguna banyak. Aplikasi tidak banyak membantu. Pelayanan publik yg tak jua memberikan pelayanan yg humanis bagi publik	pengguna banyak. aplikasi tidak banyak membantu. pelayanan publik yg tak jua memberikan pelayanan yg humanis bagi publik
Gila Ngeleg amat nih Aplikasi. Jaringan Wifi Lancar bnget buat buka yg lain,tpi buka ini Aplikasi dr tadi muterÂ² aja teros	gila ngeleg amat nih aplikasi. jaringan wifi lancar bnget buat buka yg lain,tpi buka ini aplikasi dr tadi muter Â² aja teros

b. *Cleansing*

Tahapan *cleansing* merupakan proses pembersihan data teks dengan menghilangkan elemen-elemen yang tidak relevan atau tidak diperlukan, seperti tanda baca, angka, simbol, URL, dan karakter khusus lainnya. Contoh elemen yang tidak penting meliputi tautan yang diawali dengan atribut seperti (http) atau (bit.ly), serta karakter khusus seperti (!@#\$%^&()_+{}[]<>?,:.). Tujuan dari proses *cleansing* adalah untuk meningkatkan kualitas data sehingga mempermudah analisis dan meningkatkan akurasi pada tahap pemrosesan berikutnya, seperti klasifikasi atau analisis sentimen. Diagram alir dari proses *cleansing* ditunjukkan pada Gambar 3.6.



Gambar 3. 6 Alur Proses *Cleansing*

Sebagai gambaran dari proses *cleansing* pada penelitian ini, contoh ulasan pengguna aplikasi TransJatim di Google Play Store yang telah melalui tahap pembersihan ditampilkan pada Tabel 3.4.

Tabel 3. 4 Contoh hasil proses *Celansing*

Sebelum	Sesudah
Aplikasi sudah cukup bagus dan membantu, saran utk skrng adalah penambahan armada dan halte krn pengguna sangat amat banyak, kasian halte2 kecil yg sudah tdk mendapatkan jatah naik, dan utk aplikasinya mohon ditambahkan fitur detail orang yg sedang menunggu di tiap2 halte, jadi spt ada	aplikasi sudah cukup bagus dan membantu saran utk skrng adalah penambahan armada dan halte krn pengguna sangat amat banyak kasian halte2 kecil yg sudah tdk mendapatkan jatah naik dan utk aplikasinya mohon ditambahkan fitur detail orang yg sedang menunggu di tiap2 halte jadi spt ada

Sebelum	Sesudah
aplikasi cek in - cek out di tiap halte, terimakasih	aplikasi cek in cek out di tiap halte terimakasih
Tolong diperbaiki lagi.. info perjalanan gak pas.. di aplikasi infonya di halte A, ditunggu di halte D ternyata sdh di halte F, jadinya ketinggalan bus.	tolong diperbaiki lagi info perjalanan gak pas di aplikasi infonya di halte a ditunggu di halte d ternyata sdh di halte f jadinya ketinggalan bus
Aplikasi sering eror . Di update gak bisa Pdhl sinyal full	aplikasi sering eror di update gak bisa pdhl sinyal full

c. Normalisasi

Normalisasi teks adalah proses menyamakan kata-kata tidak baku, slang, atau singkatan dalam ulasan pengguna menjadi bentuk standar sesuai KBBI. Dalam analisis sentimen pada aplikasi TransJatim, normalisasi bertujuan meningkatkan konsistensi data agar lebih mudah diproses oleh algoritma seperti *Naïve Bayes*. Misalnya, kata "gak" diubah menjadi "tidak," atau "bgt" menjadi "banget." Proses ini memastikan data lebih terstruktur dan akurat untuk analisis selanjutnya. Diagram alir dari proses normalisasi ditunjukkan pada Gambar 3.7.



Gambar 3. 7 Alur Proses Normalisasi

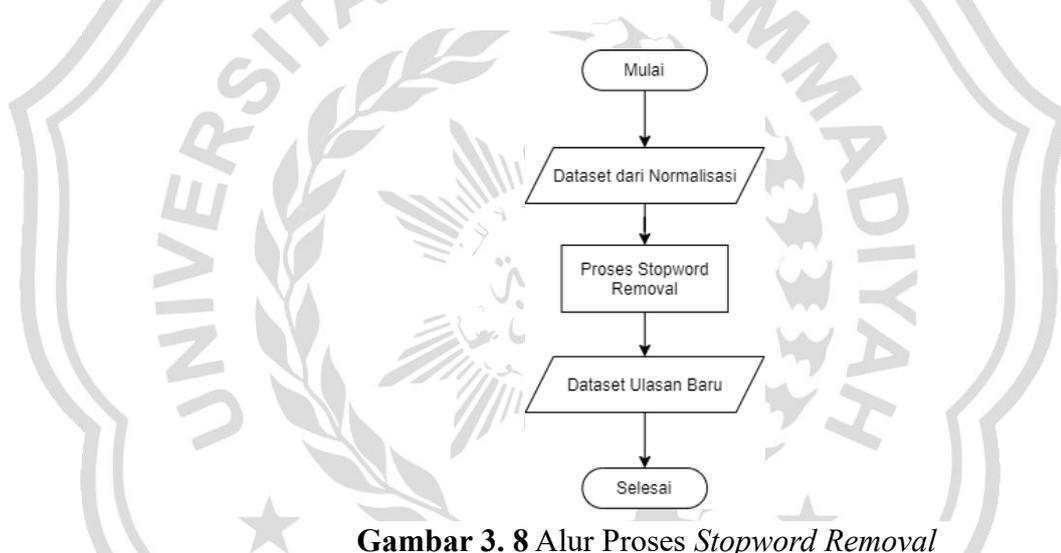
Sebagai gambaran dari proses Normalisasi pada penelitian ini, dapat ditampilkan pada Tabel 3.5.

Tabel 3. 5 Contoh Hasil Normalisasi

Sebelum	Sesudah
<p>aplikasi sudah cukup bagus dan membantu saran utk skrng adalah penambahan armada dan halte krn pengguna sangat amat banyak kasian halte2 kecil yg sudah tdk mendapatkan jatah naik dan utk aplikasinya mohon ditambahkan fitur detail orang yg sedang menunggu di tiap2 halte jadi spt ada aplikasi cek in cek out di tiap halte terimakasih</p>	<p>aplikasi cukup bagus dan membantu saran untuk sekarang penambahan armada halte karena pengguna sangat amat banyak kasian halte2 kecil yang sudah tidak mendapatkan jatah naik dan untuk aplikasinya mohon ditambahkan fitur detail orang yang sedang menunggu di tiap2 halte jadi seperti ada aplikasi cek in cek out di tiap halte terimakasih</p>
<p>tolong diperbaiki lagi info perjalanan gak pas di aplikasi infonya di halte a ditunggu di halte d ternyata sdh di halte f jadinya ketinggalan bus</p>	<p>tolong diperbaiki lagi info perjalanan tidak pas di aplikasi infonya di halte a ditunggu di halte d ternyata sudah di halte f jadinya ketinggalan bus</p>
<p>aplikasi sering eror di update gak bisa pdhl sinyal full</p>	<p>aplikasi sering eror di update tidak bisa padahal sinyal full</p>

d. *Stopword Removal*

Pada tahap *Stopword Removal*, dilakukan proses penghapusan kata-kata umum yang sering muncul tetapi tidak memiliki makna signifikan dalam analisis sentimen, seperti "dan," "yang," "adalah," dan "di." Kata-kata ini tidak memberikan kontribusi penting terhadap pemahaman konteks ulasan pengguna terhadap aplikasi TransJatim. Dengan menghapus *stopword*, proses analisis menjadi lebih efisien karena algoritma *Naïve Bayes* dapat lebih fokus pada kata-kata yang relevan dan informatif. Diagram alir proses *Stopword Removal* dapat dilihat pada Gambar 3.8.



Gambar 3. 8 Alur Proses *Stopword Removal*

Sebagai gambaran dari proses *Stopword Removal* pada penelitian ini, dapat ditampilkan pada Tabel 3.6.

Tabel 3. 6 Contoh Hasil *Stopword Removal*

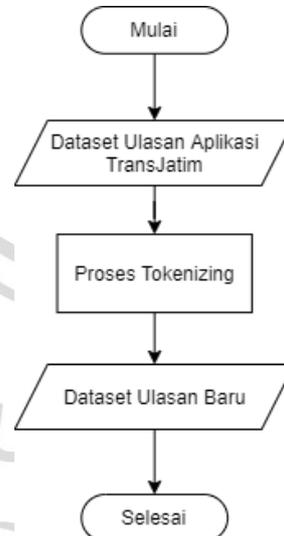
Sebelum	Sesudah
aplikasi cukup bagus dan membantu saran untuk sekarang penambahan armada halte karena pengguna sangat amat	aplikasi cukup bagus membantu saran sekarang penambahan armada halte karena pengguna sangat banyak kasian halte2 kecil

Sebelum	Sesudah
banyak kasian halte2 kecil yang sudah tidak mendapatkan jatah naik dan untuk aplikasinya mohon ditambahkan fitur detail orang yang sedang menunggu di tiap2 halte jadi seperti ada aplikasi cek in cek out di tiap halte terimakasih	mendapatkan jatah naik aplikasinya mohon ditambahkan fitur detail orang sedang menunggu tiap2 halte jadi seperti aplikasi cek in cek out tiap halte terimakasih
tolong diperbaiki lagi info perjalanan tidak pas di aplikasi infonya di halte a ditunggu di halte d ternyata sudah di halte f jadinya ketinggalan bus	tolong perbaiki info perjalanan pas aplikasi infonya halte a ditunggu halte ternyata halte f jadinya ketinggalan bus
aplikasi sering eror di update tidak bisa padahal sinyal full	aplikasi sering eror update padahal sinyal full

e. *Tokenizing*

Proses *tokenizing* adalah langkah memecah ulasan pengguna aplikasi TransJatim di Google Play Store menjadi unit-unit kecil yang disebut token. Token ini dapat berupa kata, frasa, atau simbol tergantung pada kebutuhan analisis. Langkah ini memungkinkan teks yang diproses menjadi lebih terstruktur, sehingga setiap token dapat dianalisis secara individual. Hal ini sangat penting untuk berbagai tugas pemrosesan bahasa alami NLP (Natural Language Processing), seperti analisis sentimen, klasifikasi teks, dan

ekstraksi informasi. Diagram alur dari proses *tokenizing* ditunjukkan pada Gambar 3.9.



Gambar 3. 9 Alur Proses *Tokenizing*

Sebagai ilustrasi dari proses *tokenizing*, contoh hasil pemecahan ulasan pengguna aplikasi TransJatim di Google Play Store menjadi token ditampilkan pada Tabel 3.7.

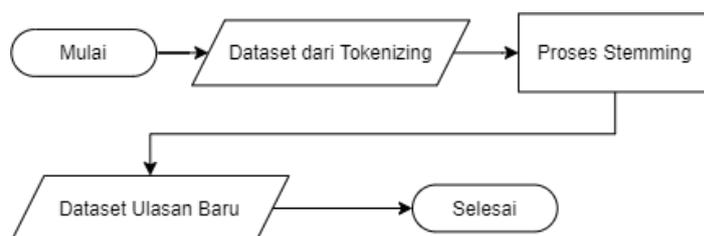
Tabel 3. 7 Contoh Hasil *Tokenizing*

Sebelum	Sesudah
aplikasi cukup bagus membantu saran sekarang penambahan armada halte karena pengguna sangat banyak kasian halte2 kecil mendapatkan jatah naik aplikasinya mohon ditambahkan fitur detail orang sedang menunggu tiap2 halte jadi seperti aplikasi cek in cek out tiap halte terimakasih	['aplikasi', 'cukup', 'bagus', 'membantu', 'saran', 'sekarang', 'penambahan', 'armada', 'halte', 'karena', 'pengguna', 'sangat', 'banyak', 'kasian', 'halte2', 'kecil', 'mendapatkan', 'jatah', 'naik', 'aplikasinya', 'mohon', 'ditambahkan', 'fitur', 'detail', 'orang', 'sedang', 'menunggu', 'tiap2', 'halte', 'jadi', 'seperti',

Sebelum	Sesudah
	'aplikasi', 'cek', 'in', 'cek', 'out', 'tiap', 'halte', 'terimakasih']
tolong perbaiki info perjalanan pas aplikasi infonya halte a ditunggu halte ternyata halte f jadinya ketinggalan bus	['tolong', 'diperbaiki', 'info', 'perjalanan', 'pas', 'aplikasi', 'infonya', 'halte', 'a', 'ditunggu', 'halte', 'ternyata', 'halte', 'f', 'jadinya', 'ketinggalan', 'bus']
aplikasi sering eror update padahal sinyal full	['aplikasi', 'sering', 'eror', 'update', 'padahal', 'sinyal', 'full']

f. *Stemming*

Proses *Stemming* dilakukan untuk mengubah kata-kata menjadi bentuk dasarnya atau akar katanya. Tujuan dari proses ini adalah menghilangkan infleksi kata seperti imbuhan dan akhiran sehingga berbagai bentuk dari satu kata dapat disederhanakan menjadi satu bentuk dasar. Sebagai contoh, kata "berjalan," "berlarian," dan "berlari-lari" akan direduksi menjadi "lari." Dalam analisis sentimen terhadap ulasan aplikasi TransJatim, *stemming* membantu menyederhanakan data dan mengurangi variasi kata yang harus dianalisis. Hal ini meningkatkan efisiensi dan akurasi algoritma *Naïve Bayes* dalam klasifikasi sentimen. Diagram alir proses *Stemming* dapat dilihat pada Gambar 3.10.



Gambar 3. 10 Alur proses *Stemming*

Sebagai gambaran dari proses *stemming* pada penelitian ini, dapat ditampilkan pada Tabel 3.8.

Tabel 3. 8 Contoh Hasil *Stemming*

Sebelum	Sesudah
['aplikasi', 'cukup', 'bagus', 'membantu', 'saran', 'sekarang', 'penambahan', 'armada', 'halte', 'karena', 'pengguna', 'sangat', 'banyak', 'kasian', 'halte2', 'kecil', 'mendapatkan', 'jatah', 'naik', 'aplikasinya', 'mohon', 'ditambahakan', 'fitur', 'detail', 'orang', 'sedang', 'menunggu', 'tiap2', 'halte', 'jadi', 'seperti', 'aplikasi', 'cek', 'in', 'cek', 'out', 'tiap', 'halte', 'terimakasih']	aplikasi cukup bagus bantu saran sekarang tambah armada halte karena guna sangat banyak kasi halte2 kecil dapat jatah naik aplikasi mohon ditambahakan fitur detail orang sedang tunggu tiap2 halte jadi seperti aplikasi cek in cek out tiap halte terimakasih
['tolong', 'diperbaiki', 'info', 'perjalanan', 'pas', 'aplikasi', 'infonya', 'halte', 'a', 'ditunggu', 'halte', 'ternyata', 'halte', 'f', 'jadinya', 'ketinggalan', 'bus']	baik info jalan pas aplikasi info halte a tunggu halte nyata halte f jadi tinggal bus
['aplikasi', 'sering', 'eror', 'update', 'padahal', 'sinyal', 'full']	aplikasi sering eror update padahal sinyal full

2. Ekstraksi Fitur

Pada tahap ekstraksi fitur dengan algoritma TF-IDF, teks ulasan aplikasi TransJatim diubah menjadi vektor numerik. Setiap kata dalam ulasan diberi bobot berdasarkan frekuensi kemunculannya di dokumen tersebut dan tingkat kepentingannya dalam keseluruhan kumpulan ulasan. Algoritma TF-IDF memungkinkan representasi teks yang lebih bermakna, sehingga mendukung proses analisis sentimen menggunakan *Naïve Bayes*. Bobot yang dihasilkan digunakan sebagai input dalam proses klasifikasi untuk menentukan sentimen positif, netral, atau negatif dari ulasan pengguna.

Tabel 3. 9 Contoh Data Ulasan Dari Data Uji

No_Dokumen	Hasil_Preprocessing
D1	aplikasi sangat bantu sekali sekarang semua posisi bus kota pantau cukup 1 aplikasi
D2	bis kurang banyak calon tumpang tolak terus kasihan nunggu cuma 1 jam lebih 3 jam naik aplikasi baik
D3	terimakasih pemprov jatim ada jalur sangat bantu dalam jalan murah mudah nyaman

Proses pembobotan kata diawali dengan mengambil kata-kata yang telah melewati tahap *preprocessing* dari *database*. Selanjutnya, dilakukan perhitungan frekuensi kemunculan setiap kata dalam dokumen untuk menentukan bobotnya. Perhitungan ini menggunakan kata-kata yang sudah melalui proses *preprocessing*, sehingga menghasilkan bobot untuk setiap kata. Hasil pembobotan kata dapat dilihat pada Tabel 3.10 berikut.

Tabel 3. 10 Menghitung TF-IDF

TERM	TF			DF	idf=log(N/DF)	TFIDF		
	D1	D2	D3			D1	D2	D3
aplikasi	2	1	0	3	0	0	0	0
sangat	1	0	1	2	0.176091259	0.17609	0	0.17609
bantu	1	0	1	2	0.176091259	0.17609	0	0.17609
sekali	1	0	0	1	0.477121255	0.47712	0	0
sekarang	1	0	0	1	0.477121255	0.47712	0	0
semua	1	0	0	1	0.477121255	0.47712	0	0
posisi	1	0	0	1	0.477121255	0.47712	0	0
bus	1	0	0	1	0.477121255	0.47712	0	0
kota	1	0	0	1	0.477121255	0.47712	0	0
pantau	1	0	0	1	0.477121255	0.47712	0	0
cukup	1	0	0	1	0.477121255	0.47712	0	0
bis	0	1	0	1	0.477121255	0	0.47712	0
kurang	0	1	0	1	0.477121255	0	0.47712	0
banyak	0	1	0	1	0.477121255	0	0.47712	0
calon	0	1	0	1	0.477121255	0	0.47712	0
tumpang	0	1	0	1	0.477121255	0	0.47712	0
tolak	0	1	0	1	0.477121255	0	0.47712	0
terus	0	1	0	1	0.477121255	0	0.47712	0
kasihan	0	1	0	1	0.477121255	0	0.47712	0
nunggu	0	1	0	1	0.477121255	0	0.47712	0
Cuma	0	1	0	1	0.477121255	0	0.47712	0
jam	0	2	0	2	0.176091259	0	0.35218	0
lebih	0	1	0	1	0.477121255	0	0.47712	0
naik	0	1	0	1	0.477121255	0	0.47712	0
baik	0	1	0	1	0.477121255	0	0.47712	0
terimakasih	0	0	1	1	0.477121255	0	0	0.47712
pemprov	0	0	1	1	0.477121255	0	0	0.47712
jatim	0	0	1	1	0.477121255	0	0	0.47712
ada	0	0	1	1	0.477121255	0	0	0.47712
jalur	0	0	1	1	0.477121255	0	0	0.47712
dalam	0	0	1	1	0.477121255	0	0	0.47712
jalan	0	0	1	1	0.477121255	0	0	0.47712
murah	0	0	1	1	0.477121255	0	0	0.47712
mudah	0	0	1	1	0.477121255	0	0	0.47712
nyaman	0	0	1	1	0.477121255	0	0	0.47712

Tabel TF-IDF (*Term Frequency-Inverse Document Frequency*) adalah metode untuk mengukur seberapa penting sebuah kata (term) dalam dokumen tertentu relatif terhadap kumpulan dokumen (corpus). Berikut adalah penjelasan detail tentang proses pada tabel 3.10:

1. *Term Frequency (TF)*

Proses penghitungan dimulai dengan menghitung *Term Frequency* (TF), yaitu frekuensi kemunculan suatu kata dalam dokumen tertentu. Sebagai contoh, kata "ada" muncul satu kali di dokumen D3, tetapi tidak muncul di dokumen D1 dan D2, sehingga nilai TF-nya adalah 1 untuk D3 dan 0 untuk D1 dan D2.

2. *DF (Document Frequency)*

DF mengukur jumlah dokumen yang mengandung kata tertentu. Misalnya, kata "ada" hanya muncul di satu dokumen (D3), sehingga nilai DF-nya adalah 1. Sebaliknya, kata "aplikasi" muncul di dokumen D1 sebanyak 2 kali dan pada dokumen D2, sehingga nilai DF-nya adalah 3.

3. *IDF (Inverse Document Frequency)*

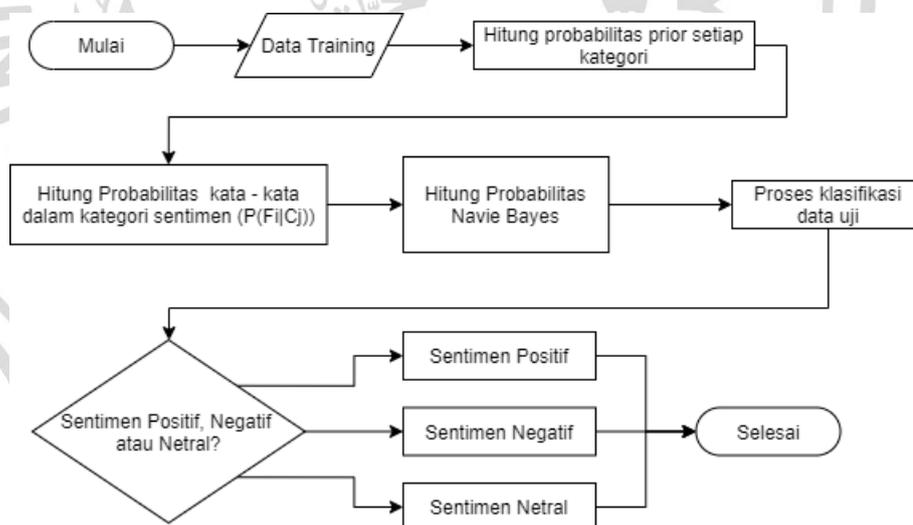
IDF digunakan untuk mengetahui seberapa unik suatu kata dalam kumpulan dokumen (corpus). Rumus IDF adalah $IDF(t) = \log\left(\frac{N}{df}\right)$, Dimana N adalah jumlah total dokumen, dan DF adalah jumlah dokumen yang mengandung kata tersebut. Contohnya, untuk kata "ada" yang memiliki $DF = 1$, nilai IDF dihitung sebagai $\log\left(\frac{3}{1}\right) = 0.4771$. Sebaliknya, untuk kata "aplikasi" yang memiliki $DF = 3$, nilai IDF-nya menjadi $\log\left(\frac{3}{3}\right) = 0$

4. TF-IDF

TF-IDF dihitung dengan mengalikan nilai TF dengan IDF untuk setiap kata dalam dokumen. Contohnya, untuk kata "ada" di dokumen D3, nilai TF adalah 1 dan IDF adalah 0.4771, sehingga nilai TF-IDF-nya adalah $1 \times 0.4771 = 0.4771$. Sementara itu, kata "aplikasi" di dokumen D1 memiliki $TF = 2$ dan $IDF = 0$, sehingga nilai TF-IDF-nya adalah $2 \times 0 = 0$. Nilai TF-IDF menunjukkan bobot atau relevansi suatu kata terhadap dokumen tertentu, di mana kata yang lebih unik dalam corpus memiliki bobot lebih tinggi.

3. Naïve Bayes

Berikut adalah gambar *flowchart* proses *Naïve Bayes* yang menjelaskan langkah-langkah pelatihan model dan pengujian data untuk klasifikasi sentimen. Proses ini dimulai dari pembagian *dataset*, pelatihan model, hingga klasifikasi data uji berdasarkan probabilitas tertinggi.



Gambar 3. 11 *Flowchart* Proses *Naïve Bayes*

Flowchart di atas menggambarkan alur proses analisis sentimen menggunakan metode *Naïve Bayes*. Berikut adalah penjelasan setiap langkahnya:

1. Mulai
 - a. Proses dimulai untuk melakukan analisis sentimen.
2. Data Training
 - a. Data pelatihan (training data) yang berisi ulasan beserta kategori sentimen (positif, negatif, atau netral) digunakan untuk melatih model.
3. Hitung Probabilitas Prior Setiap Kategori
 - a. Probabilitas awal (prior *probability*) dari setiap kategori sentimen dihitung berdasarkan distribusi kategori sentimen dalam data pelatihan.
4. Hitung Probabilitas Kata-Kata dalam Kategori Sentimen ($P(F|C)$)
 - a. Probabilitas kemunculan setiap kata dalam masing-masing kategori sentimen dihitung.
 - b. Proses ini membantu menentukan seberapa besar keterkaitan suatu kata dengan kategori sentimen tertentu.
5. Hitung Probabilitas *Naïve Bayes*
 - a. Menggunakan rumus *Naïve Bayes*, probabilitas suatu ulasan termasuk dalam kategori tertentu dihitung berdasarkan probabilitas prior dan probabilitas kata-kata dalam kategori.
6. Proses Klasifikasi Data Uji
 - a. Data uji (ulasan yang ingin diklasifikasikan) dimasukkan ke dalam model *Naïve Bayes* untuk memprediksi kategori sentimennya.

7. Keputusan: Sentimen Positif, Negatif, atau Netral?
 - a. Berdasarkan probabilitas yang dihitung, sistem menentukan apakah ulasan tersebut termasuk kategori positif, negatif, atau netral.
8. Sentimen Positif, Negatif, atau Netral
 - a. Hasil klasifikasi ditampilkan dalam tiga kategori:
 - 1 Sentimen Positif: Jika ulasan cenderung memiliki kata-kata yang dikategorikan positif.
 - 2 Sentimen Negatif: Jika ulasan lebih banyak mengandung kata-kata negatif.
 - 3 Sentimen Netral: Jika ulasan tidak condong ke positif maupun negatif.
9. Selesai
 - a. Proses analisis sentimen selesai, dan hasilnya dapat digunakan untuk keperluan visualisasi atau laporan lebih lanjut.

3.3 Representasi Model

Setelah data ulasan diubah menjadi vektor, *dataset* dibagi menjadi dua bagian menggunakan fungsi *train_test_split*. Data latih digunakan untuk melatih model menggunakan algoritma *Naïve Bayes*. Selanjutnya, model diuji dengan data uji untuk mengklasifikasikan sentimen pengguna terhadap aplikasi TransJatim. Algoritma *Naïve Bayes* menghitung nilai teks terhadap topik dan sentimen berdasarkan kata-kata yang telah dipreproses. Setelah fitur kata dikumpulkan, bobot dihitung, dan nilai tertinggi diambil untuk menentukan topik dan sentimen. Proses pengujian dilakukan dengan menghitung probabilitas tiap kategori dan memilih nilai tertinggi. Berikut adalah perhitungan dari proses *Naïve Bayes*. Sebagai contoh perhitungan, digunakan tabel 3.9 dengan ID D1:

1. Hitung probabilitas prior untuk setiap kategori. Terdapat tiga kategori yang digunakan, yaitu kelas positif, negatif, dan netral.

$$P(\text{positif}) = \frac{1}{3} = 0.3333$$

$$P(\text{negatif}) = \frac{1}{3} = 0.3333$$

$$P(\text{netral}) = \frac{1}{3} = 0.3333$$

2. Hitung Probabilitas Kata-kata dalam Kategori Sentimen ($P(F_i|C_j)$)

Hitung *frekuensi* kemunculan setiap kata dalam setiap dokumen (TF) dan jumlah dokumen yang mengandung setiap kata (DF). Kemudian hitung TF-IDF dan probabilitas menggunakan *smoothing*. Diketahui jumlah total kata dalam C_j adalah $D_1 = 12$, $D_2 = 16$, $D_3 = 12$, dan total keseluruhan kata adalah 35.

Langkah Perhitungan:

3. Hitung Probabilitas *Naïve Bayes*

Gunakan persamaan (2.10) untuk menghitung probabilitas setiap kata dalam setiap kategori sentimen.

- a. Probabilitas kata “aplikasi”

$$P(\text{aplikasi}|\text{positif}) = \frac{2 + 1}{(12 + 35)} = \frac{3}{47} = 0.0638$$

$$P(\text{aplikasi}|\text{negatif}) = \frac{1 + 1}{(16 + 35)} = \frac{2}{51} = 0.0392$$

$$P(\text{aplikasi}|\text{netral}) = \frac{0 + 1}{(12 + 35)} = \frac{1}{47} = 0.0213$$

- b. Probabilitas kata “sangat”

$$P(\text{sangat}|\text{positif}) = \frac{1 + 1}{(12 + 35)} = \frac{2}{47} = 0.0426$$

$$P(\text{sangat}|\text{negatif}) = \frac{0 + 1}{(16 + 35)} = \frac{1}{51} = 0.0196$$

$$P(\text{sangat}|\text{netral}) = \frac{1 + 1}{(12 + 35)} = \frac{2}{47} = 0.0426$$

c. Probabilitas kata “bantu”

$$P(\text{bantu}|\text{positif}) = \frac{1 + 1}{(12 + 35)} = \frac{2}{47} = 0.0426$$

$$P(\text{bantu}|\text{negatif}) = \frac{0 + 1}{(16 + 35)} = \frac{1}{51} = 0.0196$$

$$P(\text{bantu}|\text{netral}) = \frac{1 + 1}{(12 + 35)} = \frac{2}{47} = 0.0426$$

d. Probabilitas kata “sekali”

$$P(\text{sekali}|\text{positif}) = \frac{1 + 1}{(12 + 35)} = \frac{2}{47} = 0.0426$$

$$P(\text{sekali}|\text{negatif}) = \frac{0 + 1}{(16 + 35)} = \frac{1}{51} = 0.0196$$

$$P(\text{sekali}|\text{netral}) = \frac{0 + 1}{(12 + 35)} = \frac{1}{47} = 0.0213$$

e. Probabilitas kata “sekarang”

$$P(\text{sekarang}|\text{positif}) = \frac{1 + 1}{(12 + 35)} = \frac{2}{47} = 0.0426$$

$$P(\text{sekarang}|\text{negatif}) = \frac{0 + 1}{(16 + 35)} = \frac{1}{51} = 0.0196$$

$$P(\text{sekarang}|\text{netral}) = \frac{0 + 1}{(12 + 35)} = \frac{1}{47} = 0.0213$$

f. Probabilitas kata “semua”

$$P(\text{semua}|\text{positif}) = \frac{1 + 1}{(12 + 35)} = \frac{2}{47} = 0.0426$$

$$P(\text{semua}|\text{negatif}) = \frac{0 + 1}{(16 + 35)} = \frac{1}{51} = 0.0196$$

$$P(\text{semua}|\text{netral}) = \frac{0 + 1}{(12 + 35)} = \frac{1}{47} = 0.0213$$

g. Probabilitas kata “posisi”

$$P(\text{posisi}|\text{positif}) = \frac{1 + 1}{(12 + 35)} = \frac{2}{47} = 0.0426$$

$$P(\text{posisi}|\text{negatif}) = \frac{0 + 1}{(16 + 35)} = \frac{1}{51} = 0.0196$$

$$P(\text{posisi}|\text{netral}) = \frac{0 + 1}{(12 + 35)} = \frac{1}{47} = 0.0213$$

h. Probabilitas kata “bus”

$$P(\text{bus}|\text{positif}) = \frac{1 + 1}{(12 + 35)} = \frac{2}{47} = 0.0426$$

$$P(\text{bus}|\text{negatif}) = \frac{0 + 1}{(16 + 35)} = \frac{1}{51} = 0.0196$$

$$P(\text{bus}|\text{netral}) = \frac{0 + 1}{(12 + 35)} = \frac{1}{47} = 0.0213$$

i. Probabilitas kata “kota”

$$P(\text{kota}|\text{positif}) = \frac{1 + 1}{(12 + 35)} = \frac{2}{47} = 0.0426$$

$$P(\text{kota}|\text{negatif}) = \frac{0 + 1}{(16 + 35)} = \frac{1}{51} = 0.0196$$

$$P(\text{kota}|\text{netral}) = \frac{0 + 1}{(12 + 35)} = \frac{1}{47} = 0.0213$$

j. Probabilitas kata “pantau”

$$P(\text{pantau}|\text{positif}) = \frac{1 + 1}{(12 + 35)} = \frac{2}{47} = 0.0426$$

$$P(\text{pantau}|\text{negatif}) = \frac{0 + 1}{(16 + 35)} = \frac{1}{51} = 0.0196$$

$$P(\text{pantau}|\text{netral}) = \frac{0 + 1}{(12 + 35)} = \frac{1}{47} = 0.0213$$

k. Probabilitas kata “cukup”

$$P(\text{cukup}|\text{positif}) = \frac{1 + 1}{(12 + 35)} = \frac{2}{47} = 0.0426$$

$$P(\text{cukup}|\text{negatif}) = \frac{0 + 1}{(16 + 35)} = \frac{1}{51} = 0.0196$$

$$P(\text{cukup}|\text{netral}) = \frac{0 + 1}{(12 + 35)} = \frac{1}{47} = 0.0213$$

4. Proses klasifikasi data uji

Dalam klasifikasi data uji, nilai probabilitas dikalikan. Nilai tertinggi menunjukkan kelas baru dari data tersebut. Pada data uji “aplikasi sangat bantu sekali sekarang semua posisi bus kota pantau cukup 1 aplikasi”.

a. $P(uji|positif) =$

$$\begin{aligned} & P(aplikasi|positif) \times P(sangat|positif) \times \dots \times P(cukup|positif) \\ & = 0.0638 \times 0.0426 \times 0.0426 \times 0.0426 \times 0.0426 \times \dots \times 0.0426 \\ & = 0.000000000000000414213 \end{aligned}$$

b. $P(uji|negatif) =$

$$\begin{aligned} & P(aplikasi|negatif) \times P(sangat|negatif) \times \dots \times P(cukup|negatif) \\ & = 0.0426 \times 0.0196 \times 0.0196 \times 0.0196 \times 0.0196 \times \dots \times 0.0196 \\ & = 0.000000000000000001098 \end{aligned}$$

c. $P(uji|netral) =$

$$\begin{aligned} & P(aplikasi|netral) \times P(sangat|netral) \times \dots \times P(cukup|netral) \\ & = 0.0213 \times 0.0426 \times 0.0426 \times 0.0213 \times 0.0213 \times \dots \times 0.0213 \\ & = 0.000000000000000005393 \end{aligned}$$

Berdasarkan perhitungan probabilitas *Naïve Bayes* yang telah dilakukan, dapat disimpulkan bahwa kalimat dalam dokumen D1 pada tabel 3.9 ("aplikasi sangat bantu sekali sekarang semua posisi bus kota pantau cukup 1 aplikasi") menghasilkan nilai tertinggi untuk kategori positif, yaitu sebesar 0.000000000000000414213. Dengan demikian, dokumen D1 memiliki sentimen positif. Hasil perhitungan tersebut ditampilkan pada Tabel 3.11.

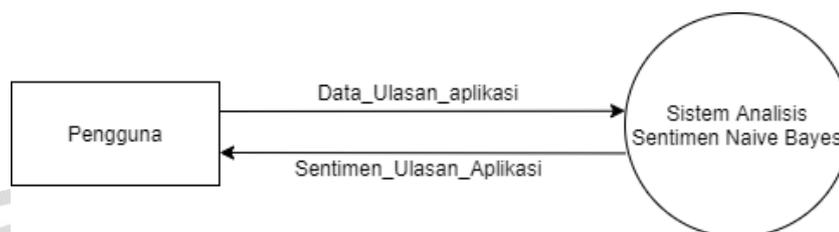
Tabel 3. 11 Contoh Hasil data uji

ID	Ulasan	Probabilitas Positif	Probabilitas Negatif	Probabilitas Netral	Prediksi Sentimen	Label Awal
D1	aplikasi sangat bantu sekali sekarang semua posisi bus kota pantau cukup 1 aplikasi	0.00000000 0000000414 213	0.00000000 0000000000 1098	0.00000000 0000000000 5393	Positif	Positif
D2	bis kurang banyak calon tumpang tolak terus kasihan nunggu cuma 1 jam lebih 3 jam naik aplikasi baik	0.00000000 0000000000 00000130	0.00000000 0000000000 01017203	0.00000000 0000000000 00000130	Negatif	Negatif
D3	terimakasih pemprov jatim ada jalur sangat bantu dalam jalan murah mudah nyaman	0.00000000 0000000006 34	0.00000000 0000000002 80	0.00000000 0000006489 33	Netral	Positif

3.4 Perancangan Sistem

3.4.1 Diagram Konteks

Diagram konteks adalah representasi visual yang menunjukkan satu proses utama dalam sebuah sistem beserta interaksinya dengan elemen-elemen terkait. Pada penelitian ini, diagram konteks yang digunakan ditampilkan pada Gambar 3.12.

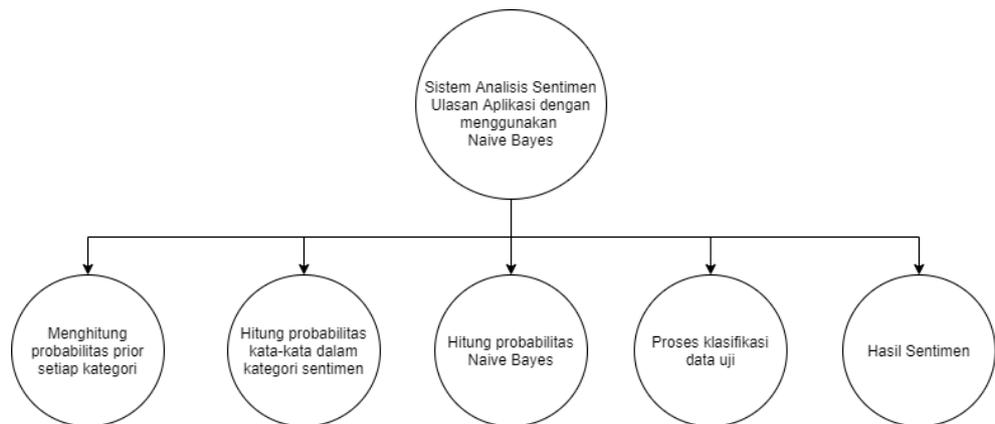


Gambar 3. 12 Diagram Konteks

Diagram konteks ini menggambarkan alur interaksi antara pengguna dan sistem analisis sentimen menggunakan metode *Naïve Bayes*. Pengguna memberikan input berupa *Data_Ulasan_Aplikasi* ke dalam sistem. Sistem kemudian memproses data tersebut melalui tahapan analisis sentimen untuk menghasilkan *Sentimen_Ulasan_Aplikasi* yang dikembalikan kepada pengguna. Diagram ini merangkum bagaimana sistem menerima input, memproses data ulasan, dan menyampaikan hasil analisis sentimen kepada pengguna.

3.4.2 Diagram Berjenjang

Diagram berjenjang merupakan runtutan proses yang terdapat dalam sistem. Dalam pengembangan sistem analisis sentimen, diagram berjenjang digunakan sebagai langkah awal dalam menggambarkan *Data Flow Diagram* (DFD) hingga ke level-level yang lebih rinci. Diagram berjenjang dari sistem analisis sentimen ulasan aplikasi TransJatim di Google Play Store menggunakan metode *Naïve Bayes* ditampilkan pada gambar 3.13.



Gambar 3. 13 Diagram Berjenjang Analisis Sentimen Ulasan

Aplikasi TransJatim di *Google PlayStore*

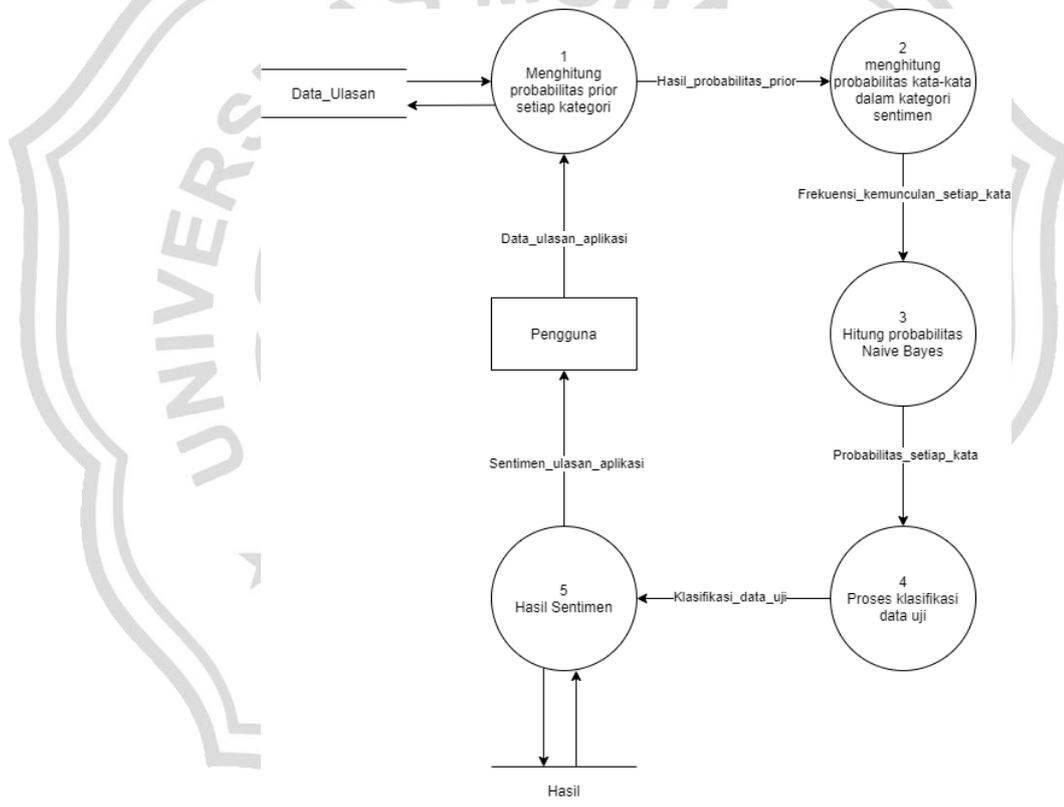
Pada Gambar 3.13 menggambarkan diagram berjenjang dari sistem dengan urutan sebagai berikut:

1. Top Level: Diagram menunjukkan Sistem Analisis Sentimen Ulasan Aplikasi dengan menggunakan *Naïve Bayes* sebagai inti proses untuk mengklasifikasikan sentimen ulasan aplikasi TransJatim.
2. Level 1: Diagram ini menggambarkan proses utama yang dilakukan dalam sistem:
 - a. Menghitung probabilitas prior setiap kategori: Menghitung probabilitas awal setiap kategori sentimen (positif, negatif, netral).
 - b. Menghitung probabilitas kata-kata dalam kategori sentimen: Menghitung probabilitas kata-kata berdasarkan distribusi dalam setiap kategori.
 - c. Menghitung probabilitas *Naïve Bayes*: Menggunakan rumus *Naïve Bayes* untuk menghitung peluang suatu ulasan termasuk dalam kategori tertentu.
 - d. Proses klasifikasi data uji: Melakukan klasifikasi sentimen berdasarkan probabilitas tertinggi dari model yang telah dilatih.

- e. Hasil Sentimen: Menyajikan hasil akhir klasifikasi sentimen kepada pengguna, berupa kategori sentimen (positif, negatif, netral).

3.4.3 Data Flow Diagram

Data Flow Diagram (DFD) untuk sistem Implementasi Metode *Naïve Bayes* dalam Melakukan Analisis Sentimen Ulasan Aplikasi TransJatim di Google Play Store menggambarkan aliran data dari proses awal hingga hasil akhir analisis sentimen. DFD ini bertujuan untuk memberikan gambaran proses yang terjadi dalam sistem. Berikut adalah gambar dari DFD Level 1 pada Gambar 3.14.



Gambar 3. 16 DFD Level 1 Analisis Sentimen Ulasan Aplikasi TransJatim di *Google PlayStore*

Berdasarkan Gambar 3.14, DFD Level 1 pada Proses Analisis Sentimen menjelaskan lima proses utama dalam implementasi analisis sentimen terhadap ulasan pengguna aplikasi TransJatim di Google Play Store, yaitu:

1. Proses 1.0 Menghitung Probabilitas Prior Setiap Kategori
 - a. Data ulasan pengguna diolah untuk menghitung probabilitas awal (*prior probability*) setiap kategori sentimen, yaitu positif, negatif, dan netral.
 - b. Proses ini menghasilkan nilai Hasil_probabilitas_prior yang akan digunakan untuk proses selanjutnya.
2. Proses 2.0 Menghitung Probabilitas Kata-Kata dalam Kategori Sentimen
 - a. Sistem menghitung frekuensi kemunculan setiap kata dalam masing-masing kategori sentimen.
 - b. Frekuensi ini digunakan untuk menghitung probabilitas setiap kata yang relevan dengan kategori sentimen tertentu.
 - c. Hasilnya adalah Frekuensi_kemunculan_setiap_kata, yang akan menjadi input untuk perhitungan probabilitas *Naïve Bayes*.
3. Proses 3.0 Hitung Probabilitas *Naïve Bayes*
 - a. Probabilitas setiap kata dari proses sebelumnya dipadukan dengan probabilitas prior untuk menghasilkan model *Naïve Bayes*.
 - b. Proses ini menghasilkan Probabilitas_setiap_kata, yang menjadi dasar untuk mengklasifikasikan data ulasan.

Pada gambar 3.15, terdapat diagram ERD yang menunjukkan struktur database untuk sistem analisis sentimen ulasan. Tabel *Dataset* berfungsi sebagai penyimpan data ulasan beserta hasil *preprocessing* seperti *casefolding*, *cleansing*, normalisasi, dan label sentimen. Tabel Admin menyimpan informasi pengguna sistem, sementara tabel *Stopword* dan Normalisasi digunakan untuk menyimpan daftar kata *stopword* dan aturan normalisasi yang diperlukan dalam proses *preprocessing*. Relasi antar tabel ini memastikan pengelolaan data ulasan dan analisis sentimen berjalan secara terstruktur dan efisien. Diagram ini mencakup 5 tabel utama yang mendukung proses analisis dari awal hingga visualisasi hasil. . Berikut rincian dari tiap tabel:

1. Tabel *User*

Tabel ini dibuat untuk secara khusus mengelola data pengguna yang memiliki akses ke sistem. Data pada tabel ini digunakan untuk melakukan autentikasi pengguna melalui login sebelum dapat menggunakan sistem. Struktur tabel *User* dapat dilihat pada tabel 3.12.

Tabel 3. 12 Tabel *User*

Nama	Tipe Data	Ukuran	Key
username	Varchar	16	PK
password	Varchar	116	-

2. Tabel *Dataset*

Tabel ini menyimpan data utama yang akan digunakan dalam proses analisis sentimen. Setiap *record* berisi data asli dan hasil *preprocessing* seperti *casefolding*, *cleansing*, normalisasi, *stopword removal*, dan label sentimen. Struktur tabel *Dataset* dapat dilihat pada tabel 3.13.

Tabel 3. 13 Tabel *Dataset*

Nama	Tipe Data	Ukuran	Key
id_dataset	Int	-	PK
data_asli	Text	-	-
casefolding	Text	-	-
cleaning	Text	-	-
normalisasi	Text	-	-
stopword	Text	-	-
preprocessing	Text	-	-
label	Varchar	25	-

3. Tabel Klasifikasi

Tabel ini digunakan untuk menyimpan pasangan kata yang akan dinormalisasi. Kata yang tidak baku (awal) akan diubah menjadi bentuk baku (akhir). Struktur tabel klasifikasi dapat dilihat pada tabel 3.14

Tabel 3. 14 Tabel Klasifikasi

Nama	Tipe Data	Ukuran	Key
id_normalisasi	Int	-	PK
awal	Varchar	100	-
akhir	Varchar	100	-

4. Tabel *Stopword*

Tabel ini digunakan untuk menyimpan data hasil evaluasi model klasifikasi sentimen dalam bentuk JSON. Nilai pada kolom *option_value* berisi informasi seperti *confusion matrix* (cm), *classification report* (cr), dan akurasi (acc). Tabel ini bersifat fleksibel untuk menyimpan berbagai jenis metrik evaluasi yang dihasilkan dari proses analisis sentimen. Struktur tabel *stopword* dapat dilihat pada tabel 3.15

Tabel 3. 15 Tabel *Stopword*

Nama	Tipe Data	Ukuran	Key
id_stopword	Int	-	PK
nama_stopword	varchar	100	-

5. Tabel *Option*

Tabel ini menyimpan konfigurasi sistem atau parameter yang digunakan dalam proses analisis, seperti pilihan-pilihan *preprocessing* atau pengaturan lainnya. Struktur tabel *option* dapat dilihat pada tabel 3.16

Tabel 3. 16 Tabel *Option*

Nama	Tipe Data	Ukuran	Key
id_option	Int	-	PK
option_name	varchar	100	-
option_value	Varchar	100	-

3.6 Perancangan Antar Muka

Desain antarmuka (*interface*) berperan sebagai media komunikasi antara sistem dan pengguna. Antarmuka ini menyajikan informasi dalam bentuk tampilan yang berisi data yang dibutuhkan oleh pengguna. Dalam penelitian ini, desain antarmuka digunakan untuk memasukkan data ulasan aplikasi Transjatim di Google Play Store dan menampilkan hasil analisis sentimen dari ulasan tersebut.

a. Antar Muka Halaman *Login Page*

Halaman *Login Page* merupakan halaman pertama yang harus diakses oleh pengguna untuk masuk ke dalam sistem. Halaman ini berfungsi sebagai pintu masuk yang memastikan hanya pengguna yang memiliki akses yang dapat menggunakan sistem. Antarmuka ini menyediakan *form* untuk mengisi *username* dan *password*.

Naive Bayes Text Mining	<p>Login</p> <input type="text" value="Username"/> <input type="password" value="Password"/> <input type="button" value="Masuk"/>
Dashboard	
Login	

Gambar 3. 20 Tampilan Halaman *Login*

b. Antar Muka *Dashboard*

Dashboard ini menampilkan analisis sentimen (Negatif, Netral, Positif) menggunakan metode *Naive Bayes*. Terdapat menu navigasi untuk mengelola *dataset*, *preprocessing* teks (*stopword*, normalisasi), klasifikasi, dan visualisasi. Tujuannya memudahkan pengguna dalam memantau hasil *text mining* secara efisien.

Naive Bayes Text Mining	<p>Home</p> <div style="display: flex; justify-content: space-around;"> <input type="button" value="Negatif"/> <input type="button" value="Netral"/> <input type="button" value="Positif"/> </div> <p>Naive Bayes adalah metode klasifikasi yang digunakan untuk memprediksi suatu data masuk ke kategori apa, berdasarkan data sebelumnya.</p>
Dashboard	
Dataset	
Stopword	
Normalisasi	
Preprocessing	
Klasifikasi Naive Bayes	
Visualisasi	

Gambar 3. 21 Tampilan *Dashboard*

c. Antar Muka Halaman *Dataset*

Halaman *Dataset* merupakan halaman yang digunakan untuk menampilkan dan mengelola data ulasan yang akan dianalisis. Pada halaman ini, pengguna dapat mengunggah *dataset* ulasan berupa file CSV atau Excel. Selain itu, halaman ini juga menampilkan *preview dataset* yang telah diunggah.

Naive Bayes Text Mining	Dataset																																														
Dashboard	Pencarian	Refresh	Tambah																																												
Dataset	Cetak																																														
Stopword	<table border="1"> <thead> <tr> <th>No</th> <th>Data Asli</th> <th>Pelabelan Awal</th> <th>Aksi</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Data Ulasan</td> <td>Positif</td> <td>Edit Hapus</td> </tr> <tr> <td>2</td> <td>Data Ulasan</td> <td>negatif</td> <td>Edit Hapus</td> </tr> <tr> <td>3</td> <td>Data Ulasan</td> <td>Netral</td> <td>Edit Hapus</td> </tr> <tr> <td>4</td> <td>Data Ulasan</td> <td>Positif</td> <td>Edit Hapus</td> </tr> <tr> <td>5</td> <td>Data Ulasan</td> <td>negatif</td> <td>Edit Hapus</td> </tr> <tr> <td>6</td> <td>Data Ulasan</td> <td>Netral</td> <td>Edit Hapus</td> </tr> <tr> <td>7</td> <td>Data Ulasan</td> <td>Positif</td> <td>Edit Hapus</td> </tr> <tr> <td>8</td> <td>Data Ulasan</td> <td>negatif</td> <td>Edit Hapus</td> </tr> <tr> <td>9</td> <td>Data Ulasan</td> <td>Netral</td> <td>Edit Hapus</td> </tr> <tr> <td>10</td> <td>Data Ulasan</td> <td>Positif</td> <td>Edit Hapus</td> </tr> </tbody> </table>			No	Data Asli	Pelabelan Awal	Aksi	1	Data Ulasan	Positif	Edit Hapus	2	Data Ulasan	negatif	Edit Hapus	3	Data Ulasan	Netral	Edit Hapus	4	Data Ulasan	Positif	Edit Hapus	5	Data Ulasan	negatif	Edit Hapus	6	Data Ulasan	Netral	Edit Hapus	7	Data Ulasan	Positif	Edit Hapus	8	Data Ulasan	negatif	Edit Hapus	9	Data Ulasan	Netral	Edit Hapus	10	Data Ulasan	Positif	Edit Hapus
No	Data Asli	Pelabelan Awal	Aksi																																												
1	Data Ulasan	Positif	Edit Hapus																																												
2	Data Ulasan	negatif	Edit Hapus																																												
3	Data Ulasan	Netral	Edit Hapus																																												
4	Data Ulasan	Positif	Edit Hapus																																												
5	Data Ulasan	negatif	Edit Hapus																																												
6	Data Ulasan	Netral	Edit Hapus																																												
7	Data Ulasan	Positif	Edit Hapus																																												
8	Data Ulasan	negatif	Edit Hapus																																												
9	Data Ulasan	Netral	Edit Hapus																																												
10	Data Ulasan	Positif	Edit Hapus																																												
Normalisasi																																															
Preprocessing																																															
Klasifikasi Naive Bayes																																															
Visualisasi																																															

Gambar 3. 22 Tampilan Halaman *Dataset*

d. Antar Muka *Stopword*

Halaman ini digunakan untuk mengelola daftar *stopword* dalam sistem *text mining*. Pengguna dapat menambah, mengedit, atau menghapus kata *stopword* melalui tabel yang menampilkan daftar kata beserta tombol aksi. Fitur pencarian dan *refresh* disediakan untuk kemudahan navigasi.

Naive Bayes Text Mining	Stopword																																			
Dashboard	Pencarian Refresh Tambah																																			
Dataset																																				
Stopword	<table border="1"> <thead> <tr> <th>No</th> <th>Nama Stopword</th> <th>Aksi</th> </tr> </thead> <tbody> <tr><td>1</td><td>Kata Stopword</td><td>Edit Hapus</td></tr> <tr><td>2</td><td>Kata Stopword</td><td>Edit Hapus</td></tr> <tr><td>3</td><td>Kata Stopword</td><td>Edit Hapus</td></tr> <tr><td>4</td><td>Kata Stopword</td><td>Edit Hapus</td></tr> <tr><td>5</td><td>Kata Stopword</td><td>Edit Hapus</td></tr> <tr><td>6</td><td>Kata Stopword</td><td>Edit Hapus</td></tr> <tr><td>7</td><td>Kata Stopword</td><td>Edit Hapus</td></tr> <tr><td>8</td><td>Kata Stopword</td><td>Edit Hapus</td></tr> <tr><td>9</td><td>Kata Stopword</td><td>Edit Hapus</td></tr> <tr><td>10</td><td>Kata Stopword</td><td>Edit Hapus</td></tr> </tbody> </table>			No	Nama Stopword	Aksi	1	Kata Stopword	Edit Hapus	2	Kata Stopword	Edit Hapus	3	Kata Stopword	Edit Hapus	4	Kata Stopword	Edit Hapus	5	Kata Stopword	Edit Hapus	6	Kata Stopword	Edit Hapus	7	Kata Stopword	Edit Hapus	8	Kata Stopword	Edit Hapus	9	Kata Stopword	Edit Hapus	10	Kata Stopword	Edit Hapus
No	Nama Stopword	Aksi																																		
1	Kata Stopword	Edit Hapus																																		
2	Kata Stopword	Edit Hapus																																		
3	Kata Stopword	Edit Hapus																																		
4	Kata Stopword	Edit Hapus																																		
5	Kata Stopword	Edit Hapus																																		
6	Kata Stopword	Edit Hapus																																		
7	Kata Stopword	Edit Hapus																																		
8	Kata Stopword	Edit Hapus																																		
9	Kata Stopword	Edit Hapus																																		
10	Kata Stopword	Edit Hapus																																		
Normalisasi																																				
Preprocessing																																				
Klasifikasi Naive Bayes																																				
Visualisasi																																				

Gambar 3. 23 Tampilan Halaman *Stopword*

e. Antar Muka Halaman Normalisasi

Halaman ini berfungsi untuk mengelola proses normalisasi teks dengan menampilkan daftar pasangan kata (kata awal dan kata baku). Pengguna dapat menambah, mengedit, atau menghapus aturan normalisasi melalui aksi yang tersedia.

Naive Bayes Text Mining	Normalisasi																																															
Dashboard	Pencarian Refresh Tambah																																															
Dataset																																																
Stopword	<table border="1"> <thead> <tr> <th>No</th> <th>Awal</th> <th>Akhir</th> <th>Aksi</th> </tr> </thead> <tbody> <tr><td>1</td><td>Kata Awal</td><td>Kata Akhir</td><td>Edit Hapus</td></tr> <tr><td>2</td><td>Kata Awal</td><td>Kata Akhir</td><td>Edit Hapus</td></tr> <tr><td>3</td><td>Kata Awal</td><td>Kata Akhir</td><td>Edit Hapus</td></tr> <tr><td>4</td><td>Kata Awal</td><td>Kata Akhir</td><td>Edit Hapus</td></tr> <tr><td>5</td><td>Kata Awal</td><td>Kata Akhir</td><td>Edit Hapus</td></tr> <tr><td>6</td><td>Kata Awal</td><td>Kata Akhir</td><td>Edit Hapus</td></tr> <tr><td>7</td><td>Kata Awal</td><td>Kata Akhir</td><td>Edit Hapus</td></tr> <tr><td>8</td><td>Kata Awal</td><td>Kata Akhir</td><td>Edit Hapus</td></tr> <tr><td>9</td><td>Kata Awal</td><td>Kata Akhir</td><td>Edit Hapus</td></tr> <tr><td>10</td><td>Kata Awal</td><td>Kata Akhir</td><td>Edit Hapus</td></tr> </tbody> </table>				No	Awal	Akhir	Aksi	1	Kata Awal	Kata Akhir	Edit Hapus	2	Kata Awal	Kata Akhir	Edit Hapus	3	Kata Awal	Kata Akhir	Edit Hapus	4	Kata Awal	Kata Akhir	Edit Hapus	5	Kata Awal	Kata Akhir	Edit Hapus	6	Kata Awal	Kata Akhir	Edit Hapus	7	Kata Awal	Kata Akhir	Edit Hapus	8	Kata Awal	Kata Akhir	Edit Hapus	9	Kata Awal	Kata Akhir	Edit Hapus	10	Kata Awal	Kata Akhir	Edit Hapus
No	Awal	Akhir	Aksi																																													
1	Kata Awal	Kata Akhir	Edit Hapus																																													
2	Kata Awal	Kata Akhir	Edit Hapus																																													
3	Kata Awal	Kata Akhir	Edit Hapus																																													
4	Kata Awal	Kata Akhir	Edit Hapus																																													
5	Kata Awal	Kata Akhir	Edit Hapus																																													
6	Kata Awal	Kata Akhir	Edit Hapus																																													
7	Kata Awal	Kata Akhir	Edit Hapus																																													
8	Kata Awal	Kata Akhir	Edit Hapus																																													
9	Kata Awal	Kata Akhir	Edit Hapus																																													
10	Kata Awal	Kata Akhir	Edit Hapus																																													
Normalisasi																																																
Preprocessing																																																
Klasifikasi Naive Bayes																																																
Visualisasi																																																

Gambar 3. 24 Tampilan Halaman Visualisasi

f. Antar Muka Halaman *Preprocessing*

Halaman ini menampilkan proses pengolahan teks secara lengkap, mulai dari data asli hingga tahapan *preprocessing* seperti *casefolding*, *cleansing*, normalisasi, *stopword removal*, tokenisasi, dan *stemming*. Tabel utama menunjukkan hasil setiap tahap untuk setiap

data, memungkinkan pengguna memantau transformasi teks secara sistematis.

Naive Bayes Text Mining	Preprocessing						
Dashboard	<input type="button" value="Pencarian"/> <input type="button" value="Refresh"/> <input type="button" value="Cetak"/>						
Dataset							
Stopword							
Normalisasi							
Preprocessing							
Klasifikasi Naive Bayes							
Visualisasi							

No	Data Asli	Casefolding	Cleaning	Norm	Stopword	Token	Stemming
1							
2							
3							
4							
5							
6							
7							
8							
9							
10							

Gambar 3. 25 Tampilan Halaman *Preprocessing*

g. Antar Muka Halaman Klasifikasi *Naïve Bayes*

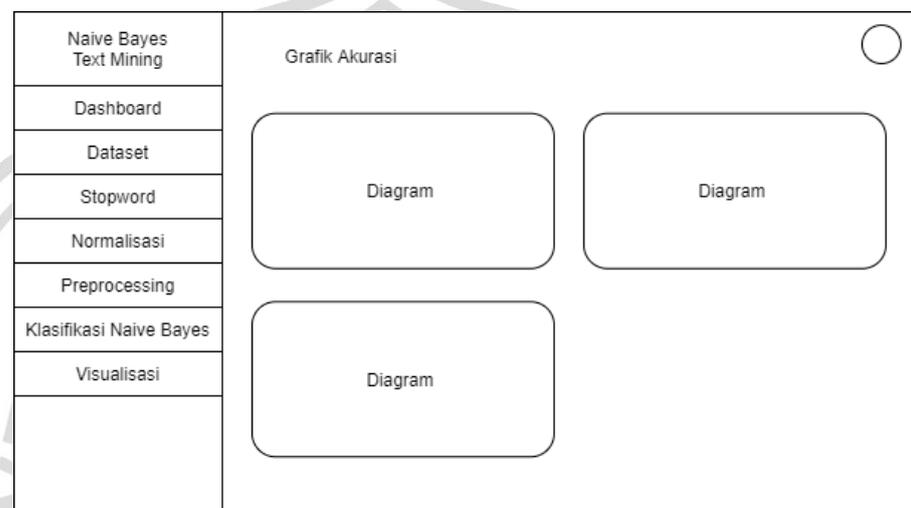
Halaman ini menampilkan proses dan hasil klasifikasi teks menggunakan algoritma *Naïve Bayes*. Pengguna dapat melakukan training model, testing data, serta melihat berbagai metrik evaluasi seperti *confusion matrix*, akurasi, dan *classification report*. Fitur TF-IDF digunakan untuk ekstraksi fitur teks.

Naive Bayes Text Mining	Klasifikasi Naive Bayes	
Dashboard	Presentase Testing	
Dataset	<input type="text"/>	
Stopword	<input type="button" value="Hitung"/>	
Normalisasi	Training	
Preprocessing	TF IDF	
Klasifikasi Naive Bayes	Testing	
Visualisasi	Confusion Matrix	
	Akurasi	
	Classification Report	

Gambar 3. 26 Tampilan Halaman Klasifikasi *Naïve Bayes*

h. Antar Muka Halaman Visualisasi

Halaman ini menampilkan berbagai diagram visual untuk mempresentasikan hasil analisis *text mining*, termasuk grafik klasifikasi dan metrik evaluasi lainnya. Pengguna dapat melihat representasi visual dari data yang telah diproses melalui antarmuka yang sederhana dan intuitif, tampilan grafis membantu pengguna mengevaluasi hasil secara lebih efektif.



Gambar 3. 27 Tampilan Halaman Visualisasi

3.7 Perancangan Pengujian Sistem

3.7.1 Pengujian Sistem

Perancangan pengujian sistem bertujuan untuk acuan dan memastikan setiap fitur pada sistem analisis sentimen ulasan aplikasi TransJatim berfungsi dengan baik. Pengujian meliputi input data, pengelolaan *dataset*, analisis sentimen, dan visualisasi hasil. Hasil pengujian dicocokkan dengan hasil yang diharapkan, sebagaimana ditunjukkan pada Tabel 3.17. Pengujian ini dilakukan secara bertahap dengan menggunakan metode *black box testing*, yang berfokus pada pengujian fungsionalitas sistem.

Tabel 3. 17 Perancangan Pengujian Sistem

No	Pengujian	Hasil yang Diharapkan	Status
1	<i>Login dengan username dan password yang valid</i>	Berhasil masuk ke dalam sistem	
2	<i>Login dengan username atau password yang salah</i>	Sistem menolak dan menampilkan pesan kesalahan	
3	<i>Login tanpa mengisi username dan password</i>	Sistem menolak dan menampilkan pesan bahwa input wajib diisi	
4	Import <i>dataset</i> dari file CSV	<i>Dataset</i> berhasil ditambahkan dan ditampilkan	
5	Menjalankan proses <i>preprocessing</i> (<i>casefolding, cleansing, normalisasi, tokenizing, stopword, stemming</i>)	Teks berhasil diproses dan ditampilkan dalam bentuk bersih	
6	Membagi data menjadi data latih dan data uji	Sistem berhasil membagi data sesuai input	
7	Melakukan perhitungan TF-IDF	Sistem menampilkan nilai TF-IDF untuk setiap dokumen	
8	Menjalankan klasifikasi menggunakan metode <i>Naïve Bayes</i>	Sistem menampilkan hasil klasifikasi beserta <i>confusion matrix</i> dan akurasi	
9	Menampilkan visualisasi hasil klasifikasi	Sistem menampilkan grafik distribusi	

No	Pengujian	Hasil yang Diharapkan	Status
		sentimen dan akurasi model	

3.7.2 Pengujian Metode *Naïve Bayes*

Penelitian ini menguji metode algoritma *Naïve Bayes* untuk menganalisis sentimen berdasarkan data ulasan pengguna aplikasi TransJatim yang diperoleh dari Google Play Store. Berikut merupakan skenario pengujian yang digunakan:

1. Data yang digunakan dalam penelitian ini adalah data ulasan aplikasi TransJatim sebanyak 459 data ulasan yang dikumpulkan dari kolom komentar di Google Play Store.
2. Ulasan tersebut telah melalui tahap *preprocessing* data, yang meliputi *cleansing*, *case folding*, *tokenizing*, *stopword removal*, dan *stemming* untuk mempersiapkan data sebelum klasifikasi.
3. Penelitian ini menggunakan algoritma *Naïve Bayes* untuk mengklasifikasikan sentimen ulasan ke dalam tiga kategori, yaitu positif, negatif, dan netral.
4. Pengujian dilakukan sebanyak tiga kali dengan rasio pembagian data training dan testing yang berbeda, yaitu 80:20, 70:30, dan 60:40, untuk melihat konsistensi dan kestabilan kinerja model dalam pembagian data yang berbeda.
5. Evaluasi performa model dilakukan menggunakan metrik akurasi, presisi, *recall*, dan *f1-score*.

3.8 Spesifikasi Pengembangan Sistem

Spesifikasi pengembangan sistem adalah rincian teknis dan kebutuhan perangkat keras, perangkat lunak, serta teknologi yang digunakan untuk merancang, membangun, dan mengimplementasikan sebuah sistem. Spesifikasi ini mencakup detail tentang komponen-

komponen yang dibutuhkan untuk memastikan sistem dapat berjalan dengan optimal sesuai tujuan yang telah ditetapkan.

Berikut adalah rincian spesifikasi pengembangan sistem:

1. Perangkat Keras

Perangkat keras adalah alat yang digunakan untuk menunjang dalam pembuatan sistem. Dalam pembuatan sistem ini perangkat keras yang digunakan yaitu laptop dengan spesifikasi:

- a. Processor Intel Core I5
- b. RAM 12 GB
- c. SSD 1000 GB
- d. Monitor 15,6"
- e. Mouse

2. Perangkat Lunak

Perangkat lunak adalah program atau aplikasi yang digunakan untuk membangun sistem. Perangkat lunak yang dibutuhkan dalam pembuatan sistem ini adalah:

- a. Windows 11
- b. *Web Server* Apache
- c. Bahasa pemrograman HTML, CSS, dan PHP
- d. *Database Server* MySql

3.9 Skenario Pengujian dan Evaluasi Model

Pengujian dalam pengembangan sistem analisis sentimen merupakan langkah penting untuk mengukur tingkat akurasi model yang digunakan. Dalam penelitian ini, pengujian dilakukan menggunakan metode *Naïve Bayes* untuk mengevaluasi performa model. Skenario pengujian dirancang untuk menentukan akurasi analisis sentimen dengan membagi *dataset* menjadi dua bagian berdasarkan rasio 80:20 ; 70:30 ; 60:40. *Dataset* yang digunakan berjumlah 459 ulasan. Pembagian *dataset* ini bertujuan untuk mengevaluasi akurasi model dalam mengklasifikasikan sentimen ulasan aplikasi. Rasio pembagian *dataset* tersebut dapat dilihat pada Tabel 3.18.

Tabel 3. 18 Tabel Pembagian *Dataset*

Rasio Pembagian % Data Latih : Data Uji	Jumlah Data Latih	Jumlah Data Uji
80 : 20	361	91
70 : 30	316	136
60 : 40	271	181

Evaluasi model dilakukan untuk menilai kinerja algoritma *Naïve Bayes* dalam menganalisis sentimen ulasan aplikasi TransJatim di Google Playstore. Proses ini mencakup pengujian model dengan data uji yang telah disembunyikan selama pelatihan. Prediksi kelas yang dihasilkan oleh model dibandingkan dengan label asli pada data uji, menghasilkan metrik evaluasi seperti akurasi, presisi, *recall*, dan *F1-score*. Metrik-metrik ini memberikan gambaran seberapa baik model mampu mengklasifikasikan sentimen ulasan. Hasil evaluasi ditampilkan dalam bentuk *confusion matrix*, yang menunjukkan perbandingan antara prediksi model dan label aktual untuk kategori sentimen positif, negatif, dan netral. *Confusion matrix* ini membantu dalam memahami performa model secara menyeluruh, termasuk jumlah prediksi benar dan salah, sehingga efektivitas metode *Naïve Bayes* dapat dievaluasi dengan baik. Perlu dicatat bahwa jumlah total data yang digunakan dalam evaluasi berkurang sebanyak 7 data karena data tersebut menjadi kosong setelah melalui proses *preprocessing*.