

BAB 4

HASIL PENELITIAN

4.1 RINGKASAN HASIL PENELITIAN

Penelitian ini melakukan klastering dan klasifikasi terhadap data ulasan masyarakat pada Google Maps RS ABC yang ada di DIY, dengan total 7.231 ulasan yang berhasil diperoleh melalui teknik *scraping* menggunakan API SerpAPI, Python dan Apify *Web Scraper*. Proses *pre-processing* data yang terdiri dari data *cleaning*, *case folding*, *data normalization*, *tokenizing*, *filtering*, *stemming*, *labelling* data dan TF-IDF menghasilkan data bersih sebanyak 5.352 ulasan. Setelah data diproses, melakukan klastering tiap cabang RS ABC di DIY menggunakan algoritma K-Means. Hasil klastering menunjukkan jumlah klaster yang berbeda di tiap cabang, yaitu: RST 7 klaster, RSV 7 klaster, RSW 4 klaster, RSX 5 klaster, RSY 5 klaster, dan RSZ 4 klaster. Ulasan masyarakat terhadap tiap cabang RS ABC di DIY menunjukkan persepsi yang beragam.

Setelah topik pembahasan pada setiap klaster teridentifikasi, selanjutnya melakukan analisis sentimen pada setiap klaster menggunakan algoritma SVM untuk mengkategorikan ulasan ke dalam sentimen positif dan negatif. Hasil klasifikasi menunjukkan bahwa mayoritas ulasan memiliki sentimen positif, yaitu sebanyak 4.340 ulasan, sedangkan sentimen negatif berjumlah 1.012 ulasan. Model SVM diterapkan secara terpisah pada masing-masing rumah sakit. Hasil evaluasi menunjukkan bahwa nilai akurasi berkisar antara 91-97%, ini menunjukkan performa yang baik dalam mengkategorikan sentimen ulasan masyarakat. Analisis ini mampu mengidentifikasi aspek layanan yang perlu dipertahankan maupun diperbaiki di masing-masing cabang, serta memberikan gambaran menyeluruh mengenai persepsi masyarakat terhadap mutu layanan RS ABC.

4.2 PENGUMPULAN DATA

Pengumpulan data pada penelitian ini menggunakan data ulasan seluruh RS ABC yang ada di DIY yang diambil dari *platform* Google Maps. Data ini dikumpulkan menggunakan teknik *scrapping*, dengan API yang didapat dari *platform* SerpAPI, bahasa pemrograman Python dan Apify *Web Scraper*. Data ulasan yang digunakan terdiri dari 2.327 data untuk RST, 285 data untuk RSV, 2.127 data untuk RSW, 2.050 data untuk RSX, 159 data untuk RSY dan 283 data untuk RSZ. Semua data ulasan disimpan dalam bentuk csv. Data ulasan yang dianalisis hanya mencakup ulasan yang berisi teks (tidak hanya berisi *rating*) serta terdiri dari kolom *name*, *publishAt*, *publishedAtDate*, *rating*, *review* dan *location*. Hasil ulasan yang terambil dapat dilihat pada Tabel 4.1.

Tabel 4.1 Sampel Hasil *Scrapping*

name	publish At	published AtDate	rating	review	location
Faaizahbros Niar	sebulan lalu	2025-03-05T06:03:15.904Z	5	Layanan medis yang bagus. Dokter dan perawat yg berpengalaman. Tempat nya bersih dan nyaman. Serasa tdk berada di rumah sakit. Meskipun tidak memiliki tempat yg luas tp sudah sesuai standart. Melayani bpjs, dan berbagai macam asuransi tanpa ribet. Lancar dan cepat.	RST
Sumur Jeru	setahun lalu	2024-04-07T01:48:37.416Z	1	Pelayanan kurang maksimal ,rumah sakit besar tapi pelayanan rasa puskesmas.. Gak rekomendasi	RSW
nianopa 1109	5 bulan lalu	2024-11-25T02:22:10.241Z	3	Proses pendaftaran nya terlalu lama... jadi antrinya tambah lama... loket BPJS hanya 2 yg buka sedangkan pasiennya banyak... mohon dibenahi	RSX
septianita nita	6 bulan lalu	2024-10-17T02:03:24Z	4	Pelayanan bagus... Dokter perawat ramah semua...	RSV
Widy Ayulestari	9 bulan lalu	2024-07-03T13:42:23Z	1	Nganter mertua berobat dari mulai habis magrib sampai dengan 8.41 belum dapet pemeriksaan dokter. Pelayanan sangat lambat dan perlu banyak pembenahan dalam pelayanan nya..	RSY
Niken Tusiana	3 hari lalu	2025-04-21T10:03:34Z	2	Dokternya selalu molor tiap kontrol. Jadwal jam 16.30 sampai jam 5 sore lebih belum datang	RSZ

4.3 PRE-PROCESSING DATA

Sebelum melakukan tahap *modelling*, perlu dilakukan *pre-processing* data untuk meningkatkan kualitas data. Tahap *pre-processing* dilakukan dengan menggabungkan data ulasan dari seluruh cabang RS ABC di DIY. Kolom yang digunakan pada *pre-processing* hanya kolom *rating*, *review* dan *location*. Tahapan ini berguna untuk membersihkan data dan melakukan *labelling* data. Selain itu, berperan dalam menyaring data yang tidak relevan serta menyusun format data agar lebih teratur dan selaras dengan kebutuhan. Dengan adanya tahapan ini data menjadi lebih bersih dan lebih siap untuk diproses dalam analisis lanjutan.

4.3.1 Data Cleaning

Tahapan pertama dalam *pre-processing* data adalah data *cleaning*. Data *cleaning* digunakan untuk menghapus karakter yang tidak diperlukan seperti tanda baca, angka, tautan, *hashtag*, *mention*, emoji, dan lainnya. Berikut ini **Kode Program 4.1** untuk menjalankan proses data *cleaning*.

Kode Program 4.1 Data Cleaning

```
1. def clean_text(text):
2.     text = re.sub(r'http\S+', '', text)
3.     text = re.sub(r'@\w+|#\w+', '', text)
4.     text = re.sub(r'\d+', '', text)
5.     text = re.sub(r'^a-zA-Z\s', '', text)
6.     text = re.sub(r'\s+', ' ', text)
7.     text = text.replace('\n', ' ')
8.     text = re.sub(r'^\x00-\x7F+', '', text)
9.     return text.strip()
```

Pada proses ini menggunakan *library regular expression* yang digunakan untuk penggantian, pencocokan dan manipulasi teks berdasarkan pola tertentu yang telah dibuat sesuai kebutuhan. Berikut ini beberapa penjelasan *function* kode yang digunakan pada proses data *cleaning*.

1. `re.sub(r'http\S+', '', text)` → menghapus URL atau *link*.
2. `re.sub(r'@\w+|#\w+', '', text)` → menghapus *mention* (@) dan *hashtags* (#).
3. `re.sub(r'\d+', '', text)` → menghapus angka dari teks.
4. `re.sub(r'^a-zA-Z\s', '', text)` → menghapus karakter yang bukan huruf dan spasi.
5. `re.sub(r'\s+', ' ', text)` → menghapus spasi yang berlebih.

6. `text.replace('\n', ' ') → menghapus enter atau baris baru sehingga menjadi satu baris.`
7. `re.sub(r'[^\x00-\x7F]+' , "", text) → menghapus karakter non-ASCII seperti emoji.`
8. `return text.strip() → mengembalikan teks yang sudah dibersihkan.`

Tahap data *cleaning* ini menjadi tahap awal yang penting karena dapat mempengaruhi akurasi dan hasil analisis. Maka data harus dipastikan benar-benar bersih dari karakter yang tidak diperlukan. Hasil dari proses data *cleaning* dapat dilihat pada Gambar 4.1.

	rating	review	location	clean_data
0	5	RST ★ 5 Dari satpam, reservasi, dokter perawat...	RST	RST Dari satpam reservasi dokter perawat semua...
1	1	UDAH NUNGGU LAMA2 TAUNYA DOKTERNYA DAH BALIK. ...	RST	UDAH NUNGGU LAMA TAUNYA DOKTERNYA DAH BALIK DI...
2	5	Layanan medis yang bagus. Dokter dan perawat y...	RST	Layanan medis yang bagus Dokter dan perawat yg...
3	1	dokternya ga tau penyakitnya cm asal dikasih o...	RST	dokternya ga tau penyakitnya cm asal dikasih o...
4	2	Ketus bgt yg di penetapan yg lain nya oke	RST	Ketus bgt yg di penetapan yg lain nya oke

Gambar 4.1 Hasil Data *Cleaning*

4.3.2 Case Folding

Pada tahap *case folding*, semua huruf akan diubah menjadi huruf kecil (*lowercase*). Berikut ini **Kode Program 4.2** yang menjalankan proses *case folding*.

Kode Program 4.2 Case Folding

```
1. def case_folding(text):
2.     return text.lower()
```

Function `.lower()` untuk mengubah semua huruf besar menjadi huruf kecil dan hasilnya dikembalikan (*return*) sebagai *output* fungsi. Hasil dari proses *case folding* dapat dilihat pada Gambar 4.2.

	rating	review	location	clean_data	casefold_data
0	5	RST ★ 5 Dari satpam, reservasi, dokter perawat...	RST	RST Dari satpam reservasi dokter perawat semua...	rst dari satpam reservasi dokter perawat semua...
1	1	UDAH NUNGGU LAMA2 TAUNYA DOKTERNYA DAH BALIK. ...	RST	UDAH NUNGGU LAMA TAUNYA DOKTERNYA DAH BALIK DI...	udah nunggu lama taunya dokternya dah balik di...
2	5	Layanan medis yang bagus. Dokter dan perawat y...	RST	Layanan medis yang bagus Dokter dan perawat yg...	layanan medis yang bagus dokter dan perawat yg...
3	1	dokternya ga tau penyakitnya cm asal dikasih o...	RST	dokternya ga tau penyakitnya cm asal dikasih o...	dokternya ga tau penyakitnya cm asal dikasih o...
4	2	Ketus bgt yg di penetapan yg lain nya oke	RST	Ketus bgt yg di penetapan yg lain nya oke	ketus bgt yg di penetapan yg lain nya oke

Gambar 4.2 Hasil *Case Folding*

4.3.3 Data Normalization

Tahap data *normalization* berfungsi untuk mengubah kata yang semula tidak baku menjadi bentuk kata baku sehingga lebih mudah untuk dipahami. Proses normalisasi menggunakan kamus *slangwords* diperlukan karena banyak ditemukan penggunaan kata-kata tidak baku, singkatan, serta penulisan yang tidak sesuai ejaan resmi. Kamus *slangwords* yang digunakan pada tahap ini disusun secara manual yang berisi daftar kata *slang* dan padanannya dalam bentuk baku. Kamus ini disimpan dalam file csv dan terdiri dari 420 kata slang. Kamus *slangwords* yang digunakan pada penelitian ini dapat dilihat pada Tabel 4.2.

Tabel 4.2 Sampel Kamus *Slangwords* Indonesia

Slang	Formal
cls	kelas
lgsg	langsung
yg	yang
terimakasih	terima kasih
nggak	tidak
ngga	tidak
ga	tidak
gak	tidak
tdk	tidak
gk	tidak

Kamus *slangwords* dapat digunakan dengan cara memuat terlebih dahulu ke dalam kode program. Setiap kata dalam teks ulasan dicocokkan dengan kamus. Jika ditemukan kata slang, maka kata tersebut akan digantikan dengan bentuk bakunya. **Kode Program 4.3** untuk menjalankan proses data *normalization*.

Kode Program 4.3 Data *Normalization*

```

1. def load_slangwords(file_path):
2.     slangwords = {}
3.     with open(file_path, mode='r', encoding='utf-8', newline='') as
4.         file:
5.             reader = csv.reader(file)
6.             for row in reader:
7.                 if len(row) >= 2:
8.                     slang, correct = row[0].strip(), row[1].strip()
9.                     slangwords[slang] = correct
10.    return slangwords

```

```

11. slangwords_file = '/content/drive/My Drive/Dataset/List
12. Slangwords.csv'
13. slangwords_dict = load_slangwords(slangwords_file)
14. def normalize_slang_words(text):
15.     words = text.split()
16.     fixed_words = [slangwords_dict[word.lower()] if word.lower() in
17.         slangwords_dict else word for word in words]
18.     return ' '.join(fixed_words)

```

Penjelasan *function* kode yang digunakan pada proses data *normalization* sebagai berikut:

1. `def load_slangwords(file_path):` digunakan untuk membaca file csv yang berisi kata slang dan padanannya, lalu menyimpannya ke dalam bentuk *dictionary*.
2. `slangwords = {}` → *dictionary* kosong yang akan diisi dengan pasangan slang: kata_baku.
3. Fungsi untuk membuka dan membaca isi file csv ditunjukkan pada baris 3-5.
4. Perulangan *for* pada baris 6-9 berfungsi untuk membaca tiap baris dengan kolom pertama sebagai slang (`row[0]`), kolom kedua sebagai kata baku (`row[1]`). Lalu disimpan dalam *dictionary slangwords*.
5. Baris 11-13 untuk membuka dan memuat file csv yang ada pada google drive disimpan dalam *dictionary slangwords*.
6. `def normalize_slang_words(text):` untuk mengganti kata slang menjadi kata baku.
7. `words = text.split()` digunakan untuk memecah kalimat menjadi *list* kata.
8. Baris 16-17 untuk melakukan pengecekan apakah kata terdapat dalam kamus *slangwords*, jika ada maka akan diganti dengan kata baku. Namun jika tidak ada akan diabaikan.
9. Baris 18 untuk menggabungkan kembali kata-kata hasil normalisasi menjadi teks utuh.

Proses data *normalization* penting karena dapat mengurangi variasi kata serta membantu menyamakan bentuk kata yang memiliki makna sama. Hasil dari proses data *normalization* dapat dilihat pada Gambar 4.3.

	rating	review	location	casefold_data	slangwords_data
0	5	RST ★5nDari satpam, reservasi, dokter perawat...	RST	rst dari satpam reservasi dokter perawat semua...	rst dari satpam reservasi dokter perawat semua...
1	1	UDAH NUNGGU LAMA2 TAUNYA DOKTERNYA DAH BALIK ...	RST	udah nunggu lama taunya dokternya dah balik di...	sudah menunggu lama taunya dokternya sudah bal...
2	5	Layanan medis yang bagus. Dokter dan perawat y...	RST	layanan medis yang bagus dokter dan perawat yg...	layanan medis yang bagus dokter dan perawat ya...
3	1	dokternya ga tau penyakitnya cm asal dikasih o...	RST	dokternya ga tau penyakitnya cm asal dikasih o...	dokternya tidak tau penyakitnya cuma asal dika...
4	2	Ketus bgt yg di penetapan yg lain nya oke	RST	ketus bgt yg di penetapan yg lain nya oke	ketus banget yang di penetapan yang lain nya oke

Gambar 4.3 Hasil Data Normalization

4.3.4 Tokenizing

Tahap *tokenizing* digunakan untuk membagi kalimat menjadi kata-kata penyusunnya. Berikut ini **Kode Program 4.4** yang menjalankan proses *tokenizing*.

Kode Program 4.4 Tokenizing

```
1. def tokenizing(text):
2.     text = word_tokenize(text)
3.     return text
```

Function `word_tokenize()` untuk memecah teks menjadi token berupa kata per kata. Karena data telah diuraikan menjadi kata per kata secara rapi dan terstruktur, maka akan mempermudah proses *filtering*. Hasil dari proses *tokenizing* dapat dilihat pada Gambar 4.4.

	rating	review	location	slangwords_data	tokenizing_data
0	5	RST ★5nDari satpam, reservasi, dokter perawat...	RST	rst dari satpam reservasi dokter perawat semua...	[rst, dari, satpam, reservasi, dokter, perawat...
1	1	UDAH NUNGGU LAMA2 TAUNYA DOKTERNYA DAH BALIK ...	RST	sudah menunggu lama taunya dokternya sudah bal...	[sudah, menunggu, lama, taunya, dokternya, sud...
2	5	Layanan medis yang bagus. Dokter dan perawat y...	RST	layanan medis yang bagus dokter dan perawat ya...	[layanan, medis, yang, bagus, dokter, dan, per...
3	1	dokternya ga tau penyakitnya cm asal dikasih o...	RST	dokternya tidak tau penyakitnya cuma asal dika...	[dokternya, tidak, tau, penyakitnya, cuma, asa...
4	2	Ketus bgt yg di penetapan yg lain nya oke	RST	ketus banget yang di penetapan yang lain nya oke	[ketus, banget, yang, di, penetapan, yang, lai...

Gambar 4.4 Hasil Tokenizing

4.3.5 Filtering

Tahap *filtering* berfungsi untuk menghapus kata-kata yang dianggap tidak memiliki makna seperti kata ‘dan’, ‘yang’, ‘sih’, ‘di’, ‘nya’, ‘dll’ dan sebagainya. Daftar kata yang digunakan pada proses ini dari *library* Sastrawi yang secara khusus

disusun untuk bahasa Indonesia. Daftar ini kemudian ditambah dengan beberapa kata yang ditemukan pada data ulasan. Berikut ini **Kode Program 4.5** yang menjalankan proses *filtering*.

Kode Program 4.5 Filtering

```

1. def stopwords(text):
2.     factory = StopWordRemoverFactory()
3.     stopwords = factory.get_stop_words()
4.     keep_words = {'tidak', 'bukan', 'belum', 'jangan', 'tanpa',
5.                  'sangat', 'cukup', 'agak', 'lebih', 'terlalu',
6.                  'paling', 'setuju', 'bisa', 'banget', 'senang',
7.                  'bahagia', 'sebab', 'puas', 'baik', 'buruk',
8.                  'tolong', 'kecewa', 'sedih', 'jelek', 'namun',
9.                  'tapi', 'sedangkan', 'pikir', 'harus', 'pasti',
10.                 'menurut', 'pendapat', 'rasa'}
11.     tambahan_stopwords = {'sih', 'nya', 'lah', 'ber', 'kan', 'e', 'an',
12.                           'po', 's'}
13.     stopwords = [word for word in stopwords if word not in keep_words]
14.     stopwords.extend(tambahan_stopwords)
15.     stopwords_set = set(stopwords)
16.     filtered = [word for word in text if word not in stopwords_set]
17.     return filtered

```

Penjelasan *function* kode yang digunakan pada proses *filtering* sebagai berikut:

1. Langkah awal pada baris 2-3 yaitu mengambil daftar *stopword remover* dari *library* Sastrawi.
2. *keep_words* pada baris 4-10 merupakan daftar kata yang tidak boleh dihapus meskipun termasuk ke dalam *stopwords*.
3. Terdapat tambahan *stopwords* seperti pada baris 11-12 karena kata-kata tersebut sering muncul dalam teks ulasan.
4. Baris 13 berfungsi untuk menghapus kata-kata yang terdapat dalam daftar *stopwords* kecuali yang termasuk dalam *keep_words*.
5. Menambahkan *stopwords* tambahan ke dalam daftar *stopwords* menggunakan fungsi pada baris 14.
6. Pada baris 15 mengubah daftar menjadi set agar pencariannya lebih cepat.
7. Baris 16 melakukan *filtering* untuk setiap kata dalam *input* teks, lalu hanya menyimpan kata yang tidak ada di dalam *stopwords_set*.
8. *return filtered* untuk mengembalikan *list* kata-kata yang sudah bebas dari *stopwords* namun tetap mempertahankan kata-kata penting.

Pada proses *filtering* ini penulis juga membuat daftar kata yang tidak boleh dihapus atau tetap harus dipertahankan karena dianggap memiliki pengaruh pada proses *modelling* dan analisis. Hasil dari *filtering* dapat dilihat pada Gambar 4.5.

	rating	review	location	tokenizing_data	filtering_data
0	5	RST ★5nDari satpam, reservasi, dokter perawat...	RST	[rst, dari, satpam, reservasi, dokter, perawat...	[rst, satpam, reservasi, dokter, perawat, semu...
1	1	UDAH NUNGGU LAMA2 TAUNYA DOKTERNYA DAH BALIK ...	RST	[sudah, menunggu, lama, taunya, dokternya, sud...	[menunggu, lama, taunya, dokternya, balik, jad...
2	5	Layanan medis yang bagus. Dokter dan perawat y...	RST	[layanan, medis, yang, bagus, dokter, dan, per...	[layanan, medis, bagus, dokter, perawat, berpe...
3	1	dokternya ga tau penyakitnya cm asal dikasih o...	RST	[dokternya, tidak, tau, penyakitnya, cuma, asa...	[dokternya, tidak, tau, penyakitnya, cuma, asa...
4	2	Ketus bgt yg di penetapan yg lain nya oke	RST	[ketus, banget, yang, di, penetapan, yang, lai...	[ketus, banget, penetapan, oke]

Gambar 4.5 Hasil *Filtering*

4.3.6 Stemming

Pada tahap *stemming*, imbuhan yang terdapat pada setiap kata dihapus untuk memperoleh kata dasar. Contohnya kata ‘perawatan’ menjadi ‘rawat’, dan ‘memuaskan’ menjadi ‘puas’. Kode Program untuk menjalankan proses *stemming* dapat dilihat pada **Kode Program 4.6**.

Kode Program 4.6 Stemming

```

1. def stemming(text):
2.     factory = StemmerFactory()
3.     stemmer = factory.create_stemmer()
4.     stem_correction = {
5.         "layan": "layan", "pelayanannya": "layan", "dilayani":
6.         "layan", "layani": "layan", "layanannya": "layan",
7.         "layananya": "layan", "pelayananya": "layan", "pelayan":
8.         "layan", "ujung": "kunjung", "karyawannya": "karyawan",
9.         "karyawanya": "karyawan", "perawatnya": "rawat", "awat":
10.        "rawat", "pelayanan": "layan", "perawat": "rawat",
11.        "lanjutin": "lanjut", "naruh": "taruh", "diilangin": "hilang",
12.        "ilangin": "hilang", "ngajar": "ajar", "asa": "rasa", "bagai":
13.        "sebagai", "ngaco": "kacau", "perawatan": "rawat", "lingkung":
14.        "lingkungan", "ngelayanin": "layan", "muas": "puas",
15.        "langgan": "pelanggan", "lakukan": "laku", "terkait": "kait",
16.        "alas": "alasan", "dar": "sekedar", "menyepelekan": "sepele"
17.    }
18.     stemmed_words = []
19.     for word in text:
20.         stemmed = stemmer.stem(word)
21.         corrected = stem_correction.get(stemmed, stemmed)
22.         stemmed_words.append(corrected)
23.     stemmed_text = ' '.join(stemmed_words)
24.     return stemmed_text

```

Penjelasan *function* kode yang digunakan pada proses *stemming* sebagai berikut:

1. Fungsi `def stemming(text)`: untuk menerima teks yang sudah ditokenisasi
2. Pada baris 2 dan 3 membuat objek stemmer dari *library* Sastrawi yang digunakan untuk mengubah kata berimbuhan menjadi bentuk dasar.
3. *Dictionary* `stem_correction` untuk mengoreksi hasil *stemming*. Karena hasil *stemming* dari Sastrawi kadang tidak tepat misalnya “pelayanan” menjadi “layan”, maka disini diperbaiki secara manual.
4. `stemmed_words = []` untuk menyimpan hasil *stemming* dan koreksi dari setiap kata dalam teks yang diproses.
5. Perulangan *for* pada baris 19-22 berfungsi untuk melakukan *stemming*, jika hasil *stemming* tidak sesuai maka ganti menggunakan *dictionary*. Lalu dimasukkan ke dalam *list* `stemmed_words`.
6. Baris 23 dan 24 digunakan untuk menggabungkan kata-kata hasil *stemming* ke bentuk teks biasa dan mengembalikan hasil akhir sebagai *string*.

Proses *stemming* dilakukan agar dapat menyeragamkan berbagai bentuk kata turunan ke dalam satu bentuk kata dasar yang sama. Hasil dari proses *stemming* dapat dilihat pada Gambar 4.6.

rating	review	location	filtering_data	stemming_data
0	5	RST ★5 Dari satpam, reservasi, dokter perawat...	RST [rst, satpam, reservasi, dokter, perawat, semu...	rst satpam reservasi dokter rawat semua perila...
1	1	UDAH NUNGGU LAMA2 TAUNYA DOKTERNYA DAH BALIK ...	RST [menunggu, lama, taunya, dokternya, balik, jad...	tunggu lama tau dokter balik jadwal jam siang ...
2	5	Layanan medis yang bagus. Dokter dan perawat y...	RST [layanan, medis, bagus, dokter, perawat, berpe...	layanan medis bagus dokter rawat alam tempat b...
3	1	dokternya ga tau penyakitnya cm asal dikasih o...	RST [dokternya, tidak, tau, penyakitnya, cuma, asa...	dokter tidak tau sakit cuma asal kasih obat bu...
4	2	Ketus bgt yg di penetapan yg lain nya oke	RST [ketus, banget, penetapan, oke]	ketus banget tetap oke

Gambar 4.6 Hasil *Stemming*

4.3.7 Labelling Data

Tahap selanjutnya setelah data sudah bersih yaitu melakukan *labelling* data. Proses *labelling* data dilakukan menggunakan skema *average labeling* berdasarkan *rating* yang terdapat pada Google Maps yaitu *rating* 1-5. Berikut ini **Kode Program 4.7** yang menjalankan proses *labelling* data.

Kode Program 4.7 Labelling Data

```

1. def label_by_rating (rating):
2.     if rating in [4, 5]:
3.         return 'Positif'
4.     elif rating in [1, 2, 3]:
5.         return 'Negatif'

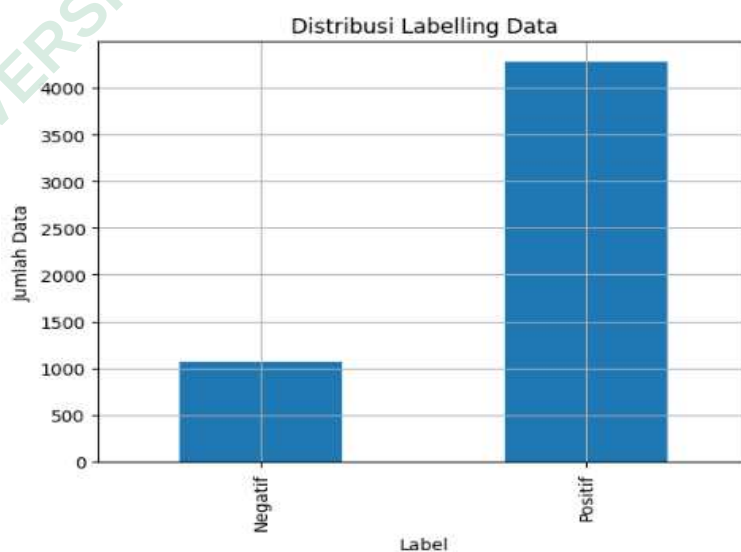
```

Rating 1, 2 dan 3 dikategorikan sebagai sentimen negatif yang menunjukkan terdapat pelayanan yang perlu diperbaiki. *Rating* 4 dan 5 dikategorikan sebagai sentimen positif yang menunjukkan bahwa pelayanan yang diberikan sudah bagus dan perlu dipertahankan. Hasil *labelling* data ditunjukkan pada Gambar 4.7.

	rating	review	location	stemming_data	labelling
0	5	RST ★5 Dari satpam, reservasi, dokter perawat...	RST	rst satpam reservasi dokter rawat semua perila...	Positif
1	1	UDAH NUNGGU LAMA2 TAUNYA DOKTERNYA DAH BALIK. ...	RST	tunggu lama tau dokter balik jadwal jam siang ...	Negatif
2	5	Layanan medis yang bagus. Dokter dan perawat y...	RST	layanan medis bagus dokter rawat alam tempat b...	Positif
3	1	dokternya ga tau penyakitnya cm asal dikasih o...	RST	dokter tidak tau sakit cuma asal kasih obat bu...	Negatif
4	2	Ketus bgt yg di penetapan yg lain nya oke	RST	ketus banget tetap oke	Negatif

Gambar 4.7 Hasil *Labelling* Data

Visualisasi hasil *labelling* data yang dapat dilihat pada Gambar 4.8 menunjukkan jumlah label sentimen positif sebanyak 4.340 dan label sentimen negatif sebanyak 1.012.



Gambar 4.8 Distribusi Hasil *Labelling* Data

4.3.8 TF-IDF

Sebelum menerapkan model analisis sentimen, data ulasan perlu diubah terlebih dahulu ke dalam bentuk numerik. Metode TF-IDF biasanya sering digunakan untuk melakukan proses ini. Proses TF-IDF ini diterapkan secara terpisah pada ulasan masing-masing cabang RS ABC di DIY. Proses TF-IDF ditunjukkan pada **Kode Program 4.8** dibawah ini.

Kode Program 4.8 TF-IDF

```
1. vectorizer = TfidfVectorizer()
2. X = vectorizer.fit_transform(df['stemming_data'])
3. feature_names = vectorizer.get_feature_names_out()
4. features_df = pd.DataFrame(X.toarray(), columns=feature_names)
```

Penjelasan *function* kode yang digunakan pada proses TF-IDF sebagai berikut:

1. *Vectorizer* digunakan untuk membuat objek TF-IDF *vectorizer* dari scikit-learn.
2. Baris kedua untuk melakukan *fit* dan *transform* secara langsung terhadap teks di kolom “stemming_data”. *Fit* digunakan untuk mempelajari semua kata unik dan menghitung IDF-nya. Sedangkan *transform* akan menghitung nilai TF-IDF dari masing-masing kata di setiap dokumen.
3. Baris ketiga digunakan untuk mengambil daftar fitur atau kolom atau kata-kata unik yang digunakan dalam representasi TF-IDF. *Feature names* merupakan *list* dari semua kata yang muncul di semua dokumen setelah *pre-processing*.
4. Baris keempat mengubah hasil TF-IDF (yang awalnya dalam bentuk *sparse matrix*) menjadi bentuk dataframe agar lebih mudah untuk dilihat.

Tahap TF-IDF digunakan untuk mengetahui seberapa besar peran sebuah kata dalam suatu ulasan dibandingkan dengan semua ulasan yang ada. Dengan begitu, kata-kata yang dianggap penting akan diberi nilai yang lebih tinggi. Hasil dari proses TF-IDF dapat dilihat pada Gambar 4.9.

alhamdulillah	bagus	baik	banget	beri	bersih	bisa	cepat	dokter
0.000000	0.000000	0.000000	0.000000	0.0	0.000000	0.0	0.000000	0.000000
0.000000	0.000000	0.000000	0.000000	0.0	0.000000	0.0	0.458950	0.000000
0.085037	0.069895	0.110178	0.000000	0.0	0.000000	0.0	0.079982	0.000000
0.000000	0.000000	0.071723	0.000000	0.0	0.000000	0.0	0.104132	0.095211
0.000000	0.000000	0.190623	0.000000	0.0	0.313619	0.0	0.000000	0.000000

Gambar 4.9 Hasil TF-IDF

4.4 MODEL KLASSTERING MENGGUNAKAN K-MEANS

Menerapkan model klastering menggunakan algoritma K-Means dari Sklearn. Implementasi model klastering menggunakan data yang sebelumnya telah direpresentasikan dalam bentuk vektor TF-IDF. Pada tahap klastering ini menerapkan metode *elbow* untuk menentukan jumlah klaster yang optimal seperti yang ditunjukkan pada **Kode Program 4.9**.

Kode Program 4.9 Metode *elbow*

```

1. K = range(2, 10)
2. wcss = []
3. for k in K:
4.     kmeans = KMeans(n_clusters=k, random_state=42)
5.     kmeans.fit(X)
6.     wcss.append(kmeans.inertia_)
7. kl = Kneelocator(K, wcss, curve="convex", direction="decreasing")
8. optimal_k = kl.elbow
9.
10. plt.figure(figsize=(8, 5))
11. plt.plot(K, wcss, 'bo-', label='WCSS per k')
12. plt.xlabel('Number of clusters (k)')
13. plt.ylabel('Within-cluster Sum of Squares (WCSS)')
14. plt.title('Elbow Method')
15. plt.legend()
16. plt.grid(True)
17. plt.show()

```

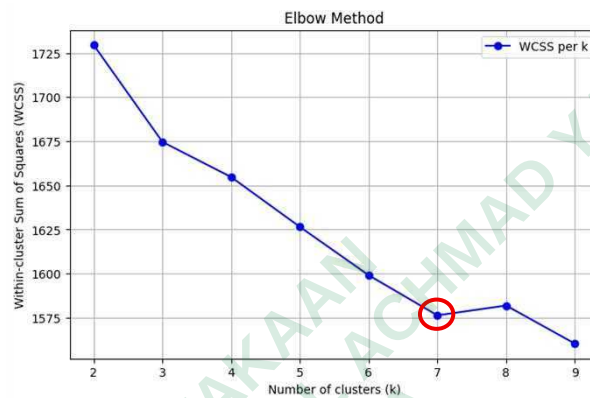
Penjelasan *function* kode yang digunakan pada proses menentukan jumlah klaster optimal (k) sebagai berikut:

1. *Looping* nilai k dari 2 hingga 10 → untuk setiap nilai k, dilakukan K-Means dilatih dan dihitung nilai WCSS (jumlah kuadrat dalam klaster). Mencoba jumlah klaster dari 2 sampai 9.
2. WCSS disimpan dalam *list* wcss sebagai ukuran seberapa baik data terkelompok.
3. *Kneelocator* digunakan untuk mencari "*elbow point*", yaitu titik di mana penurunan WCSS mulai melambat, titik ini dianggap jumlah klaster terbaik.

4. Plot *Elbow Curve* → menampilkan grafik WCSS dan k, dengan titik biru dan garis penghubung.

Metode *elbow* ini diterapkan secara terpisah pada ulasan masing-masing cabang RS ABC di DIY. Hasil metode *elbow* tiap cabang RS ABC di DIY dapat dilihat sebagai berikut:

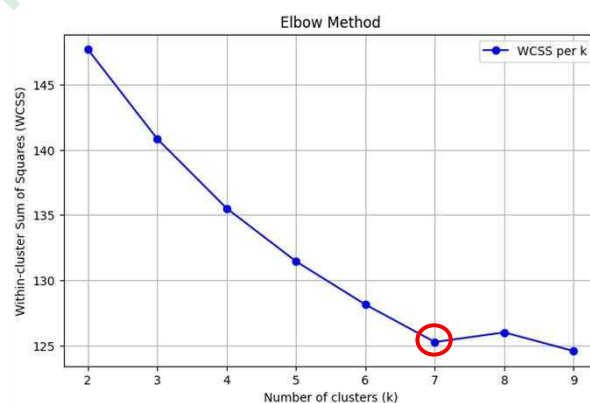
1. Hasil Metode *Elbow* RST



Gambar 4.10 Hasil Metode *Elbow* RST

Berdasarkan Gambar 4.10 dapat diketahui bahwa jumlah kluster optimal adalah 7, karena pada titik tersebut terjadi “*elbow*” atau siku. Maka analisis menggunakan 7 kluster.

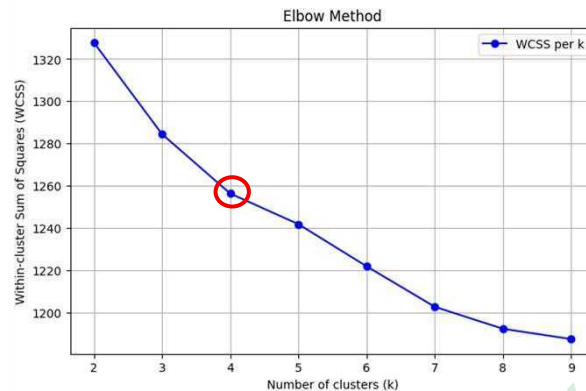
2. Hasil Metode *Elbow* RSV



Gambar 4.11 Hasil Metode *Elbow* RSV

Berdasarkan Gambar 4.11, jumlah kluster yang optimal adalah 7. Hal ini dapat dilihat dari bentuk grafik metode *elbow* yang menunjukkan adanya titik siku (*elbow*) pada $k = 7$. Maka analisis menggunakan 7 kluster.

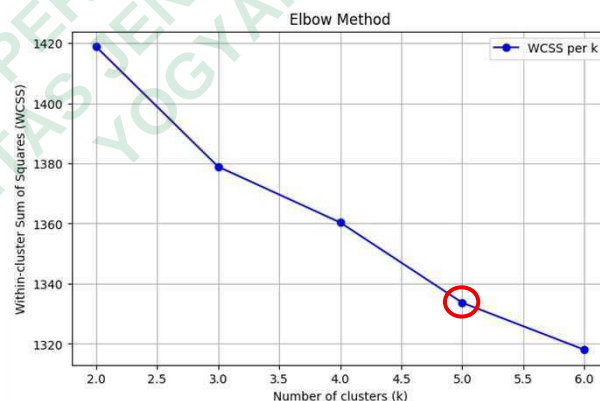
3. Hasil Metode *Elbow* RSW



Gambar 4.12 Hasil Metode *Elbow* RSW

Dari grafik *elbow* pada Gambar 4.12, terlihat bahwa WCSS turun drastis sampai $k=4$, lalu penurunannya mulai melambat. Ini berarti jumlah kluster yang paling optimal adalah 4, karena setelah itu penambahan kluster tidak terlalu mengurangi WCSS secara signifikan. Maka analisis menggunakan 4 kluster.

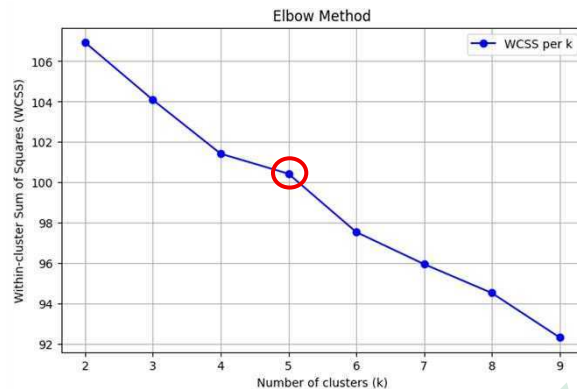
4. Hasil Metode *Elbow* RSX



Gambar 4.13 Hasil Metode *Elbow* RSX

Berdasarkan Gambar 4.13, jumlah kluster optimal yang ditentukan adalah 5, karena penurunan WCSS paling tajam terjadi sampai $k=5$. Setelah itu grafik mulai melandai atau datar. Maka analisis menggunakan 5 kluster.

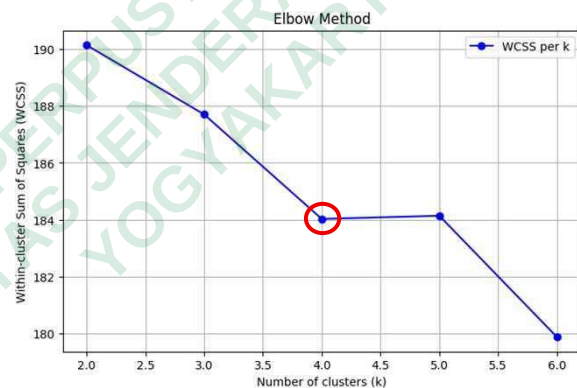
5. Hasil Metode *Elbow* RSY



Gambar 4.14 Hasil Metode *Elbow* RSY

Berdasarkan Gambar 4.14, jumlah kluster optimal adalah $k=5$, karena terlihat adanya perubahan sudut yang mencolok pada $k=5$ dimana titik ini dianggap sebagai "*elbow*". Maka analisis menggunakan 5 kluster.

6. Hasil Metode *Elbow* RSZ



Gambar 4.15 Hasil Metode *Elbow* RSZ

Berdasarkan pola penurunan WCSS pada Gambar 4.15, titik *elbow* tampak terlihat tajam pada $k=4$, karena setelah titik ini, penurunan WCSS melambat secara signifikan setelah titik tersebut. Maka analisis menggunakan 4 kluster.

Setelah mengetahui jumlah kluster optimal untuk masing-masing RS, selanjutnya menerapkan model klustering menggunakan K-Means dengan kode program yang dapat dilihat pada **Kode Program 4.10** berikut ini:

Kode Program 4.10 Model K-Means

```
1. best_k = 4
2. kmeans = KMeans(n_clusters=best_k, random_state=42)
3. df['cluster'] = kmeans.fit_predict(X) + 1
4. display(df[['rating', 'review', 'location', 'labelling',
               'cluster']])
```

Penjelasan *function* kode yang digunakan pada proses implementasi model K-Means sebagai berikut:

1. Menetapkan jumlah kluster terbaik (*best_k*), berdasarkan hasil dari grafik metode *elbow* sebelumnya.
2. Baris kedua membuat objek model K-Means dengan jumlah kluster 4 dan *random_state* untuk memastikan hasilnya konsisten setiap kali dijalankan.
3. Baris 3 dan 4, melatih model K-Means dan langsung memprediksi kluster untuk data X, hasil prediksi kluster disimpan ke kolom baru “*cluster*” di dataframe. Secara *default*, nilai hasil prediksi dimulai dari 0, sehingga +1 digunakan agar kluster dimulai dari angka 1, supaya lebih mudah dibaca, lebih konsisten dan tidak membingungkan karena angka 0 bisa disalahartikan sebagai belum masuk kluster. Lalu menampilkan hasil kluster yang telah ditentukan oleh K-Means beserta kolom *rating*, *review*, *location*, *labelling* dan *cluster*.

Model K-Means ini juga diterapkan pada ulasan masing-masing cabang RS ABC di DIY secara terpisah. Hasil klastering dan kata utama tiap kluster ditunjukkan pada Tabel 4.3 sampai Tabel 4.8. RST memiliki 7 kluster, dengan kata utama tercantum pada Tabel 4.3.

Tabel 4.3 Hasil Klastering RST

Cluster	Kata Utama
1	layanan, tidak, rawat, dokter, baik, ramah, pasien, rst, sakit, sangat
2	puas, layanan, sangat, ramah, rst, cepat, baik, sakit, bagus, tugas
3	ramah, layanan, baik, tugas, sangat, cepat, bersih, rawat, nyaman, rst
4	islami, layanan, ramah, nyaman, rst, rumah, sakit, baik, rs, bagus
5	bagus, layanan, ramah, cepat, rst, sangat, rumah, sakit, baik, fasilitas
6	rst, layanan, terima, baik, kasih, selalu, ramah, periksa, alhamdulillah, nyaman
7	baik, layanan, sangat, ramah, nyaman, rst, tempat, cukup, cepat, rawat

Berdasarkan Gambar 4.11, ulasan untuk RSV terbagi ke dalam 7 klaster, dengan kata utama dari masing-masing klaster ditampilkan pada Tabel 4.4.

Tabel 4.4 Hasil Klastering RSV

Cluster	Kata Utama
1	mantap, layanan, bersih, tempat, sakit, pulang, baru, tapi, tugas, ramah
2	baik, layanan, ramah, bagus, sangat, rs, rawat, cepat, terima, kasih
3	rsv, layanan, moga, rs, ramah, sehat, baik, sangat, gigi, anak
4	bagus, layanan, bersih, ramah, nyaman, tempat, rsv, dokter, rawat, sangat
5	dokter, ramah, rawat, layanan, moga, daftar, tidak, rsv, sangat, baik
6	layanan, ramah, sangat, puas, bersih, tempat, nyaman, cepat, dokter, rsv
7	rumah, sakit, bersih, ramah, layanan, cepat, nyaman, rsv, baru, moga

Gambar 4.12 menunjukkan bahwa ulasan RSW terbagi dalam 4 klaster, dengan kata utama tiap klaster tercantum pada Tabel 4.5.

Tabel 4.5 Hasil Klastering RSW

Cluster	Kata Utama
1	bagus, layanan, ramah, rsw, bersih, cepat, rumah, sakit, sangat, moga
2	luas, nyaman, bersih, layanan, parkir, tempat, ramah, rumah, sakit, bagus
3	layanan, baik, sangat, ramah, puas, rsw, rawat, kasih, terima, cepat
4	layanan, tidak, dokter, rawat, ramah, pasien, sakit, rsw, baik, sangat

Berdasarkan Gambar 4.13, ulasan untuk RSX terbagi ke dalam 5 klaster, dengan kata utama dari masing-masing klaster ditunjukkan pada Tabel 4.6.

Tabel 4.6 Hasil Klastering RSX

Cluster	Kata Utama
1	rsx, layanan, moga, makin, selalu, baik, maju, ramah, jaya, mantap
2	tidak, layanan, dokter, pasien, jam, rawat, antre, sangat, ramah, baik
3	bagus, layanan, ramah, sangat, bersih, cepat, banget, rumah, sakit, rawat
4	puas, layanan, sangat, rsx, ramah, baik, terima, kasih, cepat, alhamdulillah
5	layanan, ramah, baik, sangat, rawat, rsx, cepat, tugas, bagus, rumah

Pada Gambar 4.14 menunjukkan bahwa ulasan untuk RSY dikelompokkan ke dalam 5 klaster, dengan kata utama tiap klaster disajikan pada Tabel 4.7.

Tabel 4.7 Hasil Klastering RSY

Cluster	Kata Utama
1	rumah, sakit, bagi, tidak, suruh, daftar, hari, jam, satu, rs
2	nilai, layanan, tidak, beri, buat, sangat, kamar, air, sedikit, semua
3	layanan, baik, makin, luas, ramah, tempat, moga, parkir, rsy, maju
4	tidak, dokter, rawat, layanan, ramah, pasien, obat, kalau, sini, sama
5	layanan, bagus, sangat, ramah, puas, obat, cepat, kali, cukup, baik

Gambar 4.15 menunjukkan ulasan RSZ terbagi dalam 4 klaster, dengan kata utama tiap klaster ditunjukkan pada Tabel 4.8.

Tabel 4.8 Hasil Klastering RSZ

Cluster	Kata Utama
1	layanan, rsz, ramah, bpjs, tidak, terima, cepat, kasih, baik, pasien
2	tidak, layanan, pasien, tapi, baik, rs, jam, tau, dokter, sangat
3	ramah, layanan, rawat, bagus, baik, bersih, moga, makin, rsz, pasien
4	layanan, baik, sangat, ramah, rawat, rsz, pasien, cepat, dokter, puas

4.5 MODEL SENTIMEN MENGGUNAKAN SVM

Pembagian data ulasan menjadi dua kelompok yaitu data pelatihan (*training*) dan data pengujian (*testing*) merupakan langkah awal yang diperlukan sebelum menerapkan model sentimen. *Splitting* Data ini juga diterapkan pada klaster masing-masing cabang RS ABC di DIY. Pada *splitting* data menggunakan *library* sklearn dengan *function* `train_test_split` yang dapat dilihat pada **Kode Program 4.11** berikut:

Kode Program 4.11 *Splitting* Data

```
1. X_train, X_test, y_train, y_test = train_test_split(
2.     X, y, test_size=0.2, random_state=42, stratify=y)
```

Dalam penelitian ini X merupakan fitur hasil dari TF-IDF dan y berupa label atau target dari data. Penentuan proporsi data yang digunakan sebagai data *testing* menggunakan perintah `test_size=0.2` yaitu 20% dari total data. Sisanya 80% data digunakan untuk data *training*. `random_state=42` digunakan untuk memastikan hasil pembagian data selalu sama setiap dijalankan (*reproducible*), `stratify=y` berfungsi untuk mempertahankan proporsi label dalam data *train* dan data *test* agar distribusi kelas tetap seimbang di kedua set.

4.5.1 Model Support Vector Machine

Setelah masing-masing klaster berhasil diidentifikasi topik pembahasannya, tahap berikutnya adalah melakukan analisis menggunakan SVM. Model sentimen ini diterapkan pada data yang sebelumnya telah diubah menjadi bentuk vektor menggunakan TF-IDF dan telah melakukan proses *splitting* data. Berikut ini **Kode Program 4.12** untuk implementasi model sentimen menggunakan SVM.

Kode Program 4.12 Model Support Vector Machine

```
1. svm_model = LinearSVC()
2. svm_model.fit(X_train, y_train)
3. y_pred_train = svm_model.predict(X_train)
4. y_pred_test = svm_model.predict(X_test)
```

Penjelasan *function* kode yang digunakan pada proses implementasi model SVM sebagai berikut:

1. Baris pertama digunakan untuk membuat objek model klasifikasi SVM dengan menggunakan LinearSVC. Kernel linear ini akan mencoba memisahkan data menggunakan garis lurus atau *hyperplane* dalam ruang fitur.
2. Baris kedua melatih model SVM menggunakan data *train* dan labelnya. Model akan mempelajari pola dari data *train* untuk memetakan ke label *y_train*.
3. Baris ketiga menggunakan model yang sudah dilatih untuk memprediksi label dari data *train* untuk melihat seberapa baik model belajar dari data *training*.
4. Baris keempat memprediksi label dari data *test* untuk mengevaluasi kinerja model pada data yang belum pernah dilihat atau digunakan saat *training*.

4.5.2 Model Evaluation

Tahap selanjutnya yaitu melakukan evaluasi untuk menilai kinerja model. Model dievaluasi berdasarkan akurasi pada data *train* dan data *test* untuk mengetahui seberapa baik model mengenali pola dari data baru. *Classification report* digunakan untuk menggambarkan kemampuan model dalam mengidentifikasi ulasan positif dan negatif. Proses evaluasi model dapat dilihat pada **Kode Program 4.13** dan **Kode Program 4.14**.

Kode Program 4.13 Evaluasi Model Pada Data *Train*

```

1. accuracy_train = accuracy_score(y_train, y_pred_train)
2. print("Accuracy Train:", accuracy_train)
3. print("Classification Report Train:\n",
4. classification_report(y_train, y_pred_train))

```

Kode Program 4.14 Evaluasi Model Pada Data *Test*

```

1. accuracy_test = accuracy_score(y_test, y_pred_test)
2. print("Accuracy Test:", accuracy_test)
3. print("Classification Report Test:\n",
4. classification_report(y_test, y_pred_test))

```

Penjelasan *function* kode yang digunakan pada proses evaluasi model SVM sebagai berikut:

1. Fungsi `accuracy_score` digunakan untuk menghitung akurasi model pada data *training* dan data *testing* yang menghitung rasio jumlah prediksi yang benar dibandingkan dengan total jumlah data. Label yang sebenarnya yaitu `y_train` dan `y_test`, sedangkan `y_pred_train` dan `y_pred_test` merupakan hasil prediksi model.
2. Pada baris kedua digunakan untuk menampilkan hasil akurasi pada data *train* dan data *test*.
3. Lalu baris ketiga dan keempat untuk menampilkan laporan evaluasi *classification report*. Metrik yang dihasilkan seperti *precision*, *recall*, *F1-score* dan *support*.

Evaluasi model klasifikasi sentimen dilakukan secara terpisah untuk masing-masing rumah sakit. Berikut ini hasil evaluasi model pada data *training* dan data *testing* pada setiap rumah sakit yang ditunjukkan pada Tabel 4.9.

Tabel 4.9 Tabel Hasil Evaluasi Model Sentimen per RS

Nama RS	Akurasi Train	Akurasi Test	Precision	Recall	F1-Score
RST	0.99	0.92	0.89	0.80	0.84
RSV	1.0	0.97	0.49	0.50	0.49
RSW	0.98	0.91	0.91	0.83	0.86
RSX	0.99	0.94	0.93	0.92	0.92
RSY	0.99	0.92	0.95	0.86	0.89
RSZ	1.0	0.91	0.95	0.71	0.77

Berdasarkan Tabel 4.9, hasil evaluasi model klasifikasi sentimen bervariasi di tiap rumah sakit. Akurasi data train untuk semua rumah sakit berada di angka tinggi (0.98–1.0), yang menunjukkan bahwa model sangat baik mengenali pola dari data latih. Namun, perbedaan mulai terlihat pada akurasi test, yang menunjukkan seberapa baik model bekerja pada data baru. Akurasi test tertinggi dimiliki oleh RSV (0.97), diikuti oleh RSX (0.94), sementara yang lainnya berada di kisaran 0.91–0.92. Hal ini mengindikasikan bahwa performa model secara umum cukup baik.

Namun, jika dilihat dari metrik keseluruhan, RSX menunjukkan performa paling konsisten dan unggul di semua aspek. Akurasi test tinggi (0.94), precision (0.93), recall (0.92), dan F1-Score (0.92), menandakan model sangat seimbang dalam mengklasifikasi sentimen positif dan negatif. Sebaliknya, meskipun RSV memiliki akurasi test tertinggi, metrik lainnya sangat rendah (precision dan recall hanya 0.49–0.50). Selanjutnya melakukan prediksi menggunakan model SVM yang telah diterapkan untuk mengetahui sentimen dari umpan balik masyarakat. Proses prediksi data ulasan ditunjukkan pada **Kode Program 4.15** dibawah ini.

Kode Program 4.15 Prediksi Data dengan Model SVM

```
1. df['predicted_sentiment'] = svm_model.predict(X)
2. print(df['predicted_sentiment'].value_counts().sort_index())
```

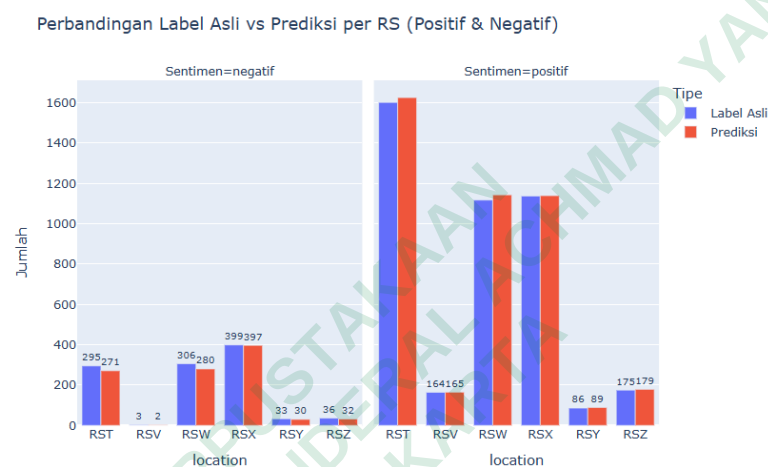
Prediksi sentimen untuk seluruh data X yang merupakan data input yang telah diolah dengan TF-IDF. Pada proses ini menggunakan perintah `predict()` dimana hasil prediksi disimpan sebagai kolom baru di dataframe dengan nama 'predicted_sentiment'. Kemudian menampilkan jumlah masing-masing label hasil prediksi menggunakan perintah `value_counts()`, `sort_index()` akan mengurutkan hasil berdasarkan nilai labelnya. Rincian hasil prediksi dari model SVM yang telah diterapkan ditunjukkan pada Tabel 4.10.

Tabel 4.10 Hasil Prediksi SVM

Nama RS	Jumlah Data	Label Positif Prediksi	Label Negatif Prediksi
RST	1896	1625	271
RSV	167	165	2

Nama RS	Jumlah Data	Label Positif Prediksi	Label Negatif Prediksi
RSW	1423	1143	280
RSX	1536	1139	397
RSY	119	89	30
RSZ	211	179	32

Perbandingan distribusi sentimen pada data aktual dan dan prediksi ditunjukkan pada Gambar 4.16.



Gambar 4.16 Perbandingan Label Asli dan Prediksi per RS

Berdasarkan Gambar 4.16 memperlihatkan bahwa hasil prediksi model memiliki kemiripan pola distribusi dengan label asli di sebagian besar rumah sakit, terutama pada RSX dan RSW. Dominasi ulasan bersentimen positif juga terlihat secara konsisten, baik pada label manual maupun hasil prediksi. Selisih jumlah antara keduanya tergolong kecil, yang mengindikasikan bahwa model memiliki tingkat kesesuaian dan akurasi yang cukup baik dalam memetakan sentimen masyarakat. Meskipun terdapat perbedaan mencolok pada beberapa rumah sakit, seperti RSV, secara umum model klasifikasi dapat diandalkan sebagai dasar untuk analisis lanjutan mengenai persepsi masyarakat terhadap layanan rumah sakit.

Setelah proses klastering dan klasifikasi berhasil diterapkan pada data ulasan, terbentuk beberapa klaster yang menggambarkan topik utama, seperti pelayanan medis, kenyamanan fasilitas dan lain-lain. Visualisasi *wordcloud* digunakan untuk menampilkan kata-kata dominan dimana ukuran kata menunjukkan frekuensi kemunculannya. Setiap klaster memiliki fokus pembahasan

yang berbeda mulai dari kepuasan hingga keluhan pasien. Analisis sentimen dilakukan untuk mengetahui persepsi masyarakat terhadap masing-masing rumah sakit. Dari hasil analisis sentimen menunjukkan bahwa arah sentimen di tiap kluster pada satu rumah sakit bervariasi, ada yang cenderung positif, ada pula yang dominan negatif, tergantung topik ulasannya. Hasil klastering dan klasifikasi setiap cabang RS ABC di DIY sebagai berikut:

1. RST

Hasil analisis pada Tabel 4.11 dibawah ini menunjukkan bahwa sebagian besar pasien memberikan respon yang sangat positif terhadap pelayanan RST. Namun, terdapat keluhan utama yang muncul, yaitu ketidaksesuaian jadwal dan ketidakhadiran dokter, yang menjadi sumber ketidakpuasan.

Tabel 4.11 Hasil Analisis RST

Cluster	Kata Utama	Analisis	Hasil Sentimen
1	layanan, tidak, rawat, dokter, baik, ramah, pasien, rst, sakit, sangat	Pasien mengapresiasi kualitas pemeriksaan namun mengeluhkan terkait kehadiran dokter yang tidak sesuai jadwal praktek.	Negatif
2	puas, layanan, sangat, ramah, rst, cepat, baik, sakit, bagus, tugas	Berfokus pada kepuasan pasien terhadap layanan RST baik terhadap sikap tenaga medis maupun kecepatan layanan.	Positif
3	ramah, layanan, baik, tugas, sangat, cepat, bersih, rawat, nyaman, rst	Penilaian positif terhadap kebersihan, kecepatan dan kenyamanan fasilitas rumah sakit, dan sikap petugas.	Positif
4	islami, layanan, ramah, nyaman, rst, rumah, sakit, baik, rs, bagus	Menyoroti penilaian positif terhadap lingkungan rumah sakit yang bernuansa Islami, yang mendukung kenyamanan selama berobat.	Positif
5	bagus, layanan, ramah, cepat, rst, sangat, rumah, sakit, baik, fasilitas	Fokus pada layanan dan fasilitas rumah sakit yang bagus, dengan apresiasi terhadap kecepatan dan keramahan staff.	Positif
6	rst, layanan, terima, baik, kasih, selalu, ramah, periksa, alhamdulillah, nyaman	Ekspresi rasa syukur dan terima kasih atas pelayanan yang diterima, pasien merasa dilayani dengan baik.	Positif
7	baik, layanan, sangat, ramah, nyaman, rst, tempat, cukup, cepat, rawat	Menekankan kenyamanan tempat dan kecepatan layanan selama perawatan yang dianggap cukup baik.	Positif

Dari hasil analisis tersebut, insight yang dapat diambil yaitu selain kompetensi medis, kepuasan pasien juga dipengaruhi oleh aspek sosial seperti keramahan, empati, serta kecepatan respons staf. Meskipun hanya satu klaster yang negatif, keluhan tersebut menyasar aspek krusial yang bisa berdampak besar terhadap kepercayaan pasien, sehingga harus segera diperbaiki

2. RSV

Pada Tabel 4.12 menunjukkan hasil analisis RSV yang memperlihatkan bahwa banyak pasien mengungkapkan kepuasan terhadap kualitas layanannya. Di sisi lain, beberapa pasien mengalami kendala dalam proses administratif. Ini menandakan adanya kebutuhan mendesak untuk meningkatkan kecepatan dalam proses administratif.

Tabel 4.12 Hasil Analisis RSV

Cluster	Kata Utama	Analisis	Hasil Sentimen
1	mantap, layanan, bersih, tempat, sakit, pulang, baru, tapi, tugas, ramah	Meskipun tergolong RS baru namun layanannya mantap, petugas ramah dan fasilitasnya bersih.	Positif
2	baik, layanan, ramah, bagus, sangat, rs, rawat, cepat, terima, kasih	Apresiasi atas layanan cepat, ramah dan perawatan yang baik disertai ungkapan terima kasih dari pasien.	Positif
3	rsv, layanan, moga, rs, ramah, sehat, baik, sangat, gigi, anak	Menyoroti layanan perawatan gigi dan anak, serta harapan agar kualitas tetap terjaga.	Positif
4	bagus, layanan, bersih, ramah, nyaman, tempat, rsv, dokter, rawat, sangat	Berfokus pada kondisi fisik rumah sakit, terutama pada kebersihan dan kenyamanan lingkungan RSV.	Positif
5	dokter, ramah, rawat, layanan, moga, daftar, tidak, rsv, sangat, baik	Menyoroti pengalaman pasien yang positif namun ada harapan perbaikan terkait proses administratif.	Negatif
6	layanan, ramah, sangat, puas, bersih, tempat, nyaman, cepat, dokter, rsv	Pasien sangat puas terhadap pelayanan di RSV, dari keramahan hingga kebersihan dan kenyamanan.	Positif
7	rumah, sakit, bersih, ramah, layanan, cepat, nyaman, rsv, baru, moga	Citra sebagai rumah sakit baru mendapatkan kesan positif terutama soal kebersihan dan kenyamanan.	Positif

Insight yang bisa dimanfaatkan oleh pihak manajemen RSV. Pertama, citra positif sebagai rumah sakit baru menunjukkan bahwa fondasi layanan sudah kuat, kekuatan ini perlu dijaga dengan terus memastikan konsistensi standar

pelayanan dan fasilitas. Kedua, layanan perawatan anak dan gigi perlu menjadi fokus berkelanjutan, karena pasien menaruh harapan besar pada keberlanjutan dan peningkatan kualitas di area ini. Ketiga, administrasi menjadi titik rawan, meski hanya satu klaster yang negatif, isu ini bersifat fundamental dan dapat memengaruhi kepuasan secara menyeluruh jika dibiarkan. Oleh karena itu, RSV perlu memperbaiki sistem antrean, transparansi informasi jadwal, serta memberikan pelatihan komunikasi bagi staf administratif dan tenaga medis agar interaksi dengan pasien lebih efektif.

3. RSW

Hasil analisis RSW seperti pada Tabel 4.13 secara keseluruhan menunjukkan kualitas layanan yang bagus, kecepatan layanan, keramahan tenaga medis dan fasilitas fisik yang bersih, dengan banyak pasien yang sangat puas dan menyampaikan rasa terima kasih. Namun, terdapat satu klaster yang menunjukkan keluhan terhadap pelayanan rawat inap atau interaksi dengan dokter yang perlu menjadi fokus perbaikan di RSW.

Tabel 4.13 Hasil Analisis RSW

Cluster	Kata Utama	Analisis	Hasil Sentimen
1	bagus, layanan, ramah, rsw, bersih, cepat, rumah, sakit, sangat, moga	Didominasi dengan kepuasan pasien terhadap layanan yang ramah dan prosesnya cepat, disertai harapan kualitas ini tetap terjaga.	Positif
2	luas, nyaman, bersih, layanan, parkir, tempat, ramah, rumah, sakit, bagus	Fokus pada kenyamanan fasilitas rumah sakit, seperti tempat yang luas, bersih dan area parkir yang memadai.	Positif
3	layanan, baik, sangat, ramah, puas, rsw, rawat, kasih, terima, cepat	Pasien puas dengan pelayanan RSW selama proses perawatan, serta ungkapan rasa syukur setelah dirawat.	Positif
4	layanan, tidak, dokter, rawat, ramah, pasien, sakit, rsw, baik, sangat	Ketidakpuasan saat berinteraksi dengan dokter atau perawatan yang dianggap kurang maksimal.	Negatif

Dari hasil analisis tersebut, insight yang dapat diambil yaitu rumah sakit disarankan untuk memperkuat standar operasional prosedur pelayanan medis, meningkatkan pelatihan komunikasi tenaga medis, dan menjaga kualitas fasilitas fisik yang telah mendapatkan respon baik. Selain itu, rumah sakit juga

bisa memanfaatkan kekuatan pada aspek kenyamanan dan keramahan sebagai citra positif dalam materi promosi. Dengan perbaikan berkelanjutan pada aspek yang dikritik dan menjaga performa yang sudah baik, RSW dapat terus meningkatkan kepercayaan dan kepuasan masyarakat terhadap pelayanannya.

4. RSX

Hasil analisis terhadap ulasan RSX menunjukkan mayoritas ulasan berkategori positif seperti yang ditunjukkan pada Tabel 4.14. Banyak pasien menyampaikan harapan dan dukungan agar rumah sakit ini terus maju dan berkembang. Namun demikian, munculnya kata “tidak” dalam ulasan menunjukkan keluhan terhadap waktu tunggu, antrean panjang, dan keterlambatan dokter yang berpengaruh pada pengalaman pasien.

Tabel 4.14 Hasil Analisis RSX

Cluster	Kata Utama	Analisis	Hasil Sentimen
1	rsx, layanan, moga, makin, selalu, baik, maju, ramah, jaya, mantap	Pasien menilai layanan baik dan ramah disertai harapan agar RSX terus maju dan jaya.	Positif
2	tidak, layanan, dokter, pasien, jam, rawat, antre, sangat, ramah, baik	Terdapat keluhan yang berkaitan dengan jam layanan, antrean dan keterlambatan dokter.	Negatif
3	bagus, layanan, ramah, sangat, bersih, cepat, banget, rumah, sakit, rawat	Pengalaman positif pada kualitas layanan secara menyeluruh, terutama pada kebersihan fasilitas.	Positif
4	puas, layanan, sangat, rsx, ramah, baik, terima, kasih, cepat, alhamdulillah	Didominasi rasa syukur dan kepuasan atas pelayanan yang cepat dan ramah dari RSX.	Positif
5	layanan, ramah, baik, sangat, rawat, rsx, cepat, tugas, bagus, rumah	Fokus pada keramahan dan kecepatan tenaga medis dalam menangani pasien.	Positif

Meskipun sentimen dominan positif, ekspektasi masyarakat terhadap pelayanan kesehatan semakin tinggi. RSX dapat mempertahankan citra positifnya dengan menjaga konsistensi layanan dan melakukan perbaikan di titik-titik rawan keluhan, seperti waktu tunggu dan respons tenaga medis. Ulasan juga menunjukkan bahwa pasien tak hanya menilai dari segi medis, tetapi juga dari pengalaman interaksi dan sistem layanan yang dijalani.

5. RSY

Hasil analisis RSY seperti pada Tabel 4.15 dibawah ini yang menunjukkan bahwa masalah utama terletak pada lamanya proses pelayanan awal dan kurangnya komunikasi antar petugas. Klaster yang berisi kata seperti “suruh”, “daftar”, “jam”, dan “hari” menggambarkan kesulitan pasien dalam proses pendaftaran dan ketidakpastian waktu pelayanan. Selain itu, muncul juga keluhan terhadap dokter dan antrean obat yang lama.

Tabel 4.15 Hasil Analisis RSY

Cluster	Kata Utama	Analisis	Hasil Sentimen
1	rumah, sakit, bagi, tidak, suruh, daftar, hari, jam, satu, rs	Adanya keluhan tentang lambatnya pelayanan awal, khususnya di bagian pendaftaran, dan antrian yang lama.	Negatif
2	nilai, layanan, tidak, beri, buat, sangat, kamar, air, sedikit, semua	Menyoroti kenyamanan pada layanan rawat inap dan terdapat kritik terkait kualitas air mineral untuk pasien.	Positif
3	layanan, baik, makin, luas, ramah, tempat, moga, parkir, rsy, maju	Apresiasi terhadap peningkatan layanan, area parkir yang luas dan harapan untuk kemajuan RSY.	Positif
4	tidak, dokter, rawat, layanan, ramah, pasien, obat, kalau, sini, sama	Fokus utama keluhan adalah interaksi dengan dokter, layanan rawat inap dan antrean obat yang lama.	Negatif
5	layanan, bagus, sangat, ramah, puas, obat, cepat, kali, cukup, baik	Puas terhadap layanan RSY yang bagus dan ramah, termasuk kualitas pelayanan obat.	Positif

Meskipun ada beberapa kritik, sentimen positif tetap kuat terutama pada aspek kenyamanan dan pelayanan yang responsif. Ini menunjukkan bahwa pasien menilai rumah sakit bukan hanya dari hasil medis, tetapi juga dari pengalaman menyeluruh selama berada di fasilitas kesehatan. Serta terdapat harapan agar rumah sakit terus berkembang. Perbaikan kecil pada aspek teknis dan komunikasi dapat meningkatkan citra dan kepercayaan terhadap RSY.

6. RSZ

Hasil analisis RSZ dapat dilihat pada Tabel 4.16 dibawah ini, pelayanan RSZ dinilai cukup baik oleh mayoritas pasien terkait keramahan, kecepatan layanan, dan kemudahan akses BPJS. Namun, terdapat ketidakpuasan seperti yang tercermin dari kata “tidak”, “tau”, dan “jam” yang menunjukkan bahwa

pasien seringkali menghadapi masalah dalam hal kecepatan pelayanan dan waktu yang dihabiskan untuk menunggu. Keluhan lainnya berkaitan dengan fasilitas yang kurang memadai.

Tabel 4.16 Hasil Analisis RSZ

Cluster	Kata Utama	Analisis	Hasil Sentimen
1	layanan, rsz, ramah, bpjs, tidak, terima, cepat, kasih, baik, pasien	Menyoroti ekspresi syukur terhadap layanan yang cepat dan ramah khususnya untuk pasien BPJS.	Positif
2	tidak, layanan, pasien, tapi, baik, rs, jam, tau, dokter, sangat	Keluhan terkait lamanya antrean, keterlambatan dokter dan fasilitas yang tidak memadai.	Negatif
3	ramah, layanan, rawat, bagus, baik, bersih, moga, makin, rsz, pasien	Pelayanan saat ini dinilai baik terutama pada rawat inap, dan pasien berharap RSZ dapat terus meningkatkan kualitasnya.	Positif
4	layanan, baik, sangat, ramah, rawat, rsz, pSasien, cepat, dokter, puas	Mencerminkan kepuasan pasien terhadap layanan RSZ secara umum, serta keramahan dokter dan perawat.	Positif

Ulasan dari pasien menunjukkan bahwa pelayanan cepat dan sikap ramah tenaga medis sangat diapresiasi, bahkan menjadi penentu utama persepsi positif terhadap rumah sakit. Kepuasan pasien BPJS juga menjadi sorotan penting, menunjukkan bahwa pelayanan yang adil dan merata dapat meningkatkan kepercayaan publik terhadap RSZ.

Secara keseluruhan, permasalahan yang paling banyak dikeluhkan masyarakat terhadap RS diantaranya:

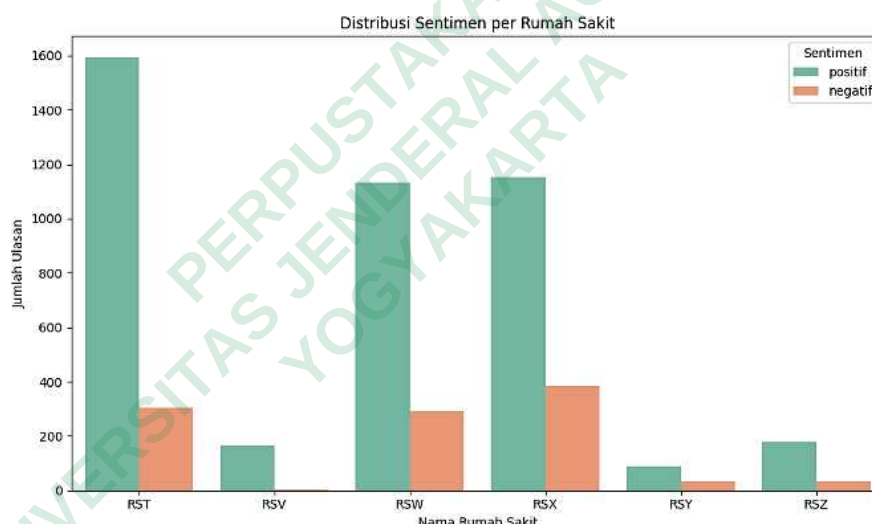
1. Waktu tunggu dan antrean yang lama, baik untuk antrean pendaftaran, pelayanan dokter, hingga pengambilan obat.
2. Dokter yang sering terlambat atau tidak hadir, ini berdampak langsung pada kepercayaan pasien.
3. Fasilitas fisik yang belum memadai, seperti ruang tunggu yang tidak nyaman dan parkir yang kurang luas.
4. Kurangnya informasi atau komunikasi yang jelas baik dari petugas pendaftaran, perawat maupun dokter

4.6 VISUALISASI DAN ANALISIS HASIL

Visualisasi dan analisis hasil merupakan tahapan akhir setelah mendapatkan hasil dari proses klustering dan klasifikasi. Visualisasi membantu menyajikan distribusi sentimen dan topik secara jelas, serta menggambarkan hal-hal yang sering dibahas masyarakat di tiap rumah sakit. Dengan menggunakan *dashboard*, akan mempermudah dalam melihat distribusi sentimen, topik utama per kluster, serta aspek pelayanan yang perlu ditingkatkan. Hasil visualisasi dianalisis guna memahami dan memperoleh wawasan dari topik-topik yang dihasilkan pada setiap klasternya, sehingga mendukung pengambilan keputusan berbasis data.

4.6.1 Visualisasi Hasil Sentimen

Visualisasi ini memberikan gambaran umum mengenai persepsi masyarakat terhadap masing-masing rumah sakit berdasarkan klasifikasi sentimen otomatis.



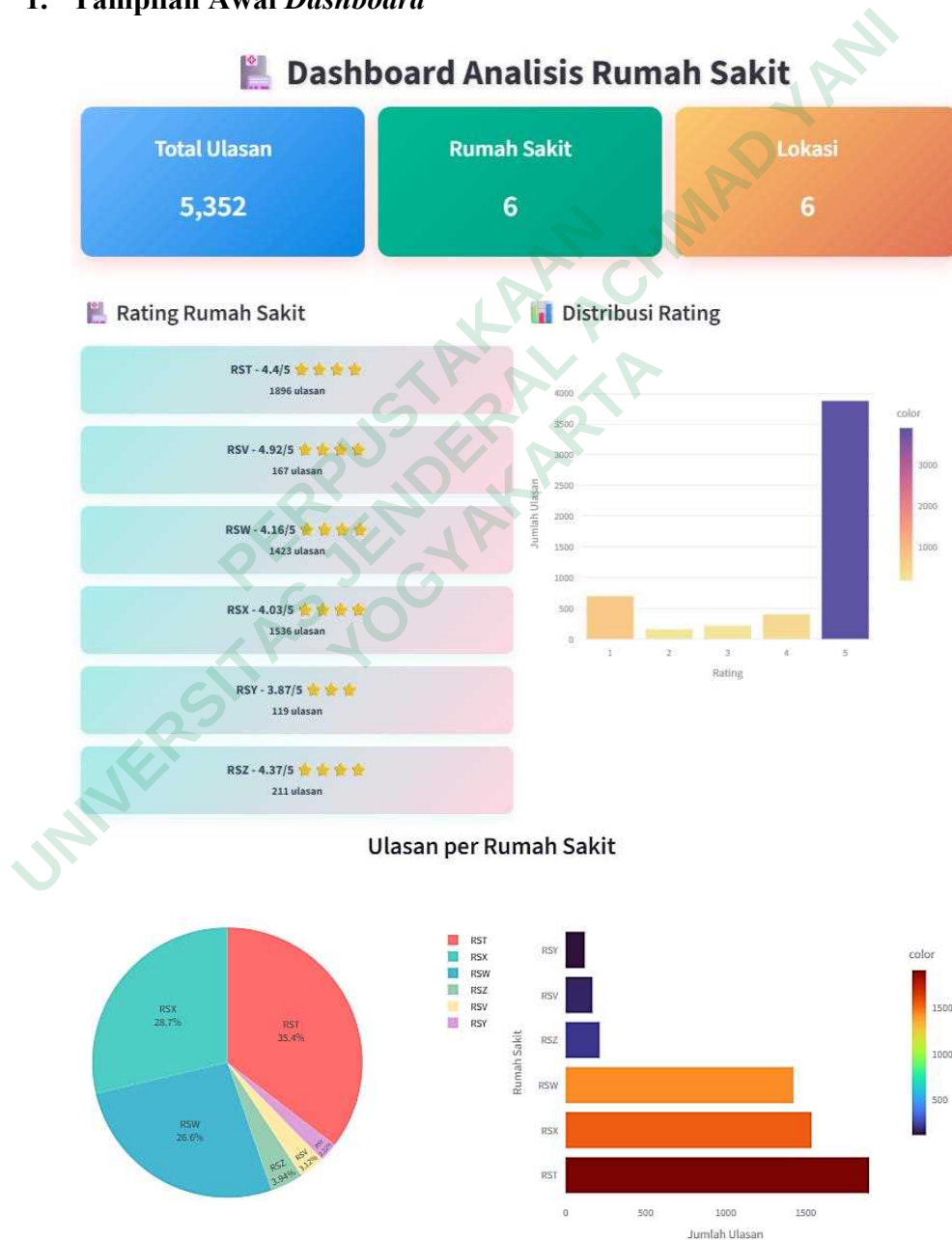
Gambar 4.17 Distribusi Sentimen Tiap RS

Berdasarkan Gambar 4.17 terlihat bahwa sebagian besar ulasan masyarakat bersentimen positif, terutama pada RST, RSX, dan RSW yang memiliki jumlah ulasan positif jauh lebih dominan dibanding ulasan negatif. Sebaliknya, beberapa rumah sakit seperti RSV dan RSY memiliki jumlah ulasan negatif yang cukup signifikan. Secara keseluruhan, hasil ini menunjukkan bahwa persepsi masyarakat terhadap layanan rumah sakit cenderung baik, meskipun masih terdapat persepsi negatif.

4.6.2 Visualisasi *Dashboard*

Visualisasi *dashboard* disusun untuk mempermudah pemahaman terhadap hasil analisis data secara menyeluruh dan terstruktur. Melalui tampilan visual yang informatif, *dashboard* menyajikan berbagai informasi penting seperti distribusi ulasan, hasil klastering topik, serta analisis sentimen dari masyarakat terhadap layanan rumah sakit.

1. Tampilan Awal *Dashboard*



Gambar 4.18 Tampilan Awal *Dashboard*

Dashboard pada Gambar 4.18 menampilkan total ulasan, jumlah rumah sakit, jumlah lokasi yang dianalisis dan rating tiap rumah sakit. Selain itu menampilkan distribusi rating secara keseluruhan, serta jumlah ulasan per rumah sakit dalam bentuk pie chart dan bar chart. Tampilan awal ini untuk memberikan gambaran umum mengenai sebaran ulasan dan penilaian masyarakat terhadap tiap rumah sakit.

2. Tampilan Menu *Clustering*



Gambar 4.19 Tampilan Menu *Clustering* (Visualisasi)

Analisis dan Insight Clustering

Analisis Clustering per Rumah Sakit:

RST:

- Total Ulasan: 1896 ulasan
- Rating Rata-rata: 4.40/5.0
- Jumlah Cluster: 7 cluster
- Distribusi Cluster: {1: 1026, 2: 96, 3: 243, 4: 91, 5: 109, 6: 204, 7: 127}

Analisis Detail per Cluster

Pilih cluster untuk analisis detail:

Cluster 1

Statistik Cluster 1:

- Total reviews: 1520
- Rating range: 1.0 - 5.0
- positif: 1247 (82.0%)
- negatif: 273 (18.0%)

Distribusi per Rumah Sakit:

- RST: 1026 ulasan
- RSX: 305 ulasan
- RSW: 110 ulasan
- RSZ: 52 ulasan
- RSV: 14 ulasan
- RSY: 13 ulasan

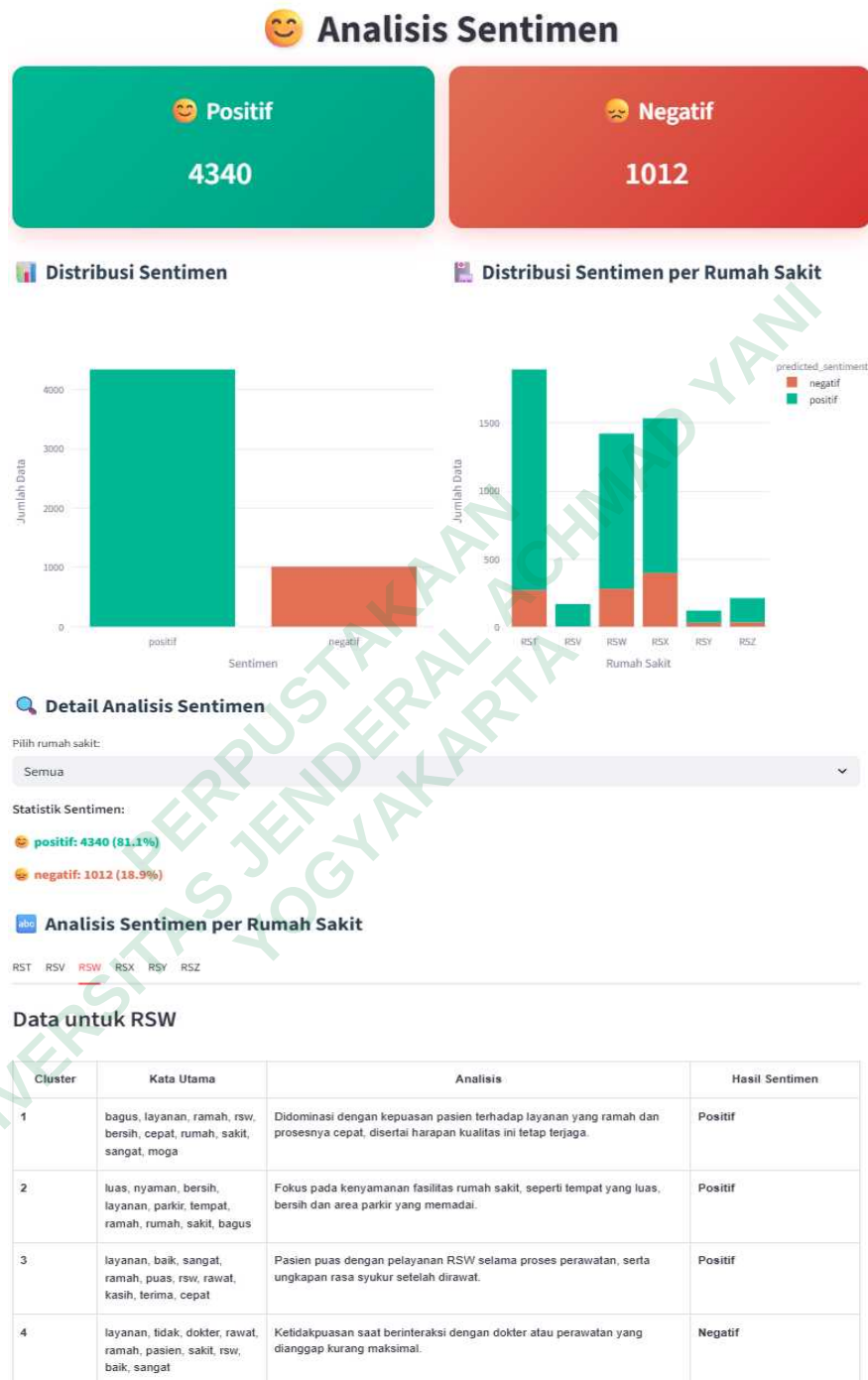
Sample Reviews:

1. alur pemeriksaan tidak jelas sama sekali, perawat / pegawai sangat tidak membantu
2. mohon maaf dengan sangat, saya dirawat dikelas 3 di RST dengan pembayaran umum. Sekedar saran saja jika pasien rawat inap harusnya mendapatkan air mineral setidaknya satu hari satu botol karena air mineral itu kan standar sekali ya dan penting, soalnya waktu dikasih makan pagi siang malam gak ada air itu rasanya..... sekian terima kasih
3. Luar biasa

Gambar 4.20 Tampilan Menu *Clustering* (Analisis)

Dashboard yang ditunjukkan pada Gambar 4.19 dan Gambar 4.20 ini menampilkan hasil analisis klustering dan total klaster. Visualisasi mencakup distribusi klaster secara keseluruhan dan distribusi klaster per rumah sakit. Terdapat juga detail isi dan contoh ulasan dari tiap klaster. Tujuannya adalah untuk mengidentifikasi topik utama yang menjadi perhatian masyarakat agar dapat fokus pada aspek-aspek pelayanan yang paling sering dibahas.

3. Tampilan Menu Analisis Sentimen



Gambar 4.21 Tampilan Menu Analisis Sentimen



Gambar 4.22 Tampilan Menu Analisis Sentimen (Contoh Ulasan)

Dashboard seperti pada Gambar 4.21 dan Gambar 4.22 ini menampilkan hasil analisis sentimen dari total 5.352 ulasan yang diklasifikasikan menjadi 4.340 ulasan positif dan 1.012 ulasan negatif. Visualisasi mencakup distribusi sentimen secara umum dan per rumah sakit, serta analisis kata kunci dan makna tiap klaster berdasarkan sentimennya. Terdapat juga contoh ulasan berdasarkan kategori sentimennya. Tujuannya adalah untuk memahami persepsi masyarakat terhadap layanan rumah sakit, mengidentifikasi aspek-aspek yang diapresiasi maupun yang dikeluhkan, serta sebagai dasar evaluasi dan perbaikan layanan.