BAB III
METODE PENELITIAN

3.1 Objek Penelitian

Studi metode ini mengandalkan data yang diperoleh melalui TANGKAR dengan mengunduh file yang relevan guna mendukung proses analisis. Pemilihan sumber data dari pemerintah daerah Karawang ini dilakukan berdasarkan keandalannya dalam menyediakan informasi yang sesuai dengan ruang lingkup. Penelitian ini menggunakan metode klasifikasi yang didasarkan pada penerapan algoritma *Random Forest* dan *Support Vector Machine* (SVM). Keduanya algoritma tersebut digunakan dalam proses analisis keluhan masyarakat, dengan tujuan utama mengukur tingkat urgensi setiap keluhan secara akurat. Selain itu, penelitian ini juga mengevaluasi keluhan berdasarkan isu dan permasalahan yang diidentifikasi, serta memetakan status penyelesaian keluhan, baik yang telah diterima maupun yang belum diterima oleh pihak berwenang.

Seluruh proses penelitian dilakukan secara daring melalui *Google Colab*, dengan memanfaatkan dataset. Analisis dilakukan secara terstruktur untuk memastikan bahwa keluhan masyarakat dapat diklasifikasikan secara tepat sesuai tingkat urgensinya. Penelitian ini dilaksanakan berdasarkan jadwal yang telah dirancang, dengan demikian proses pengumpulan dan pengolahan data dapat dilakukan secara terstruktur, tepat, dan sesuai dengan arah serta tujuan penelitian yang telah ditetapkan.

Penelitian ini dilaksanakan dalam kurun waktu 7 bulan, dimulai dari Oktober 2024 hingga april 2025. Rangkaian kegiatan penelitian dirancang secara sistematis untuk memastikan semua tahapan Pelaksanaan penelitian dapat berjalan selaras dengan sasaran yang telah dirumuskan sebelumnya. Pada Tabel 3.1 menjelaskan tentang waktu pelaksanaan penelitian :

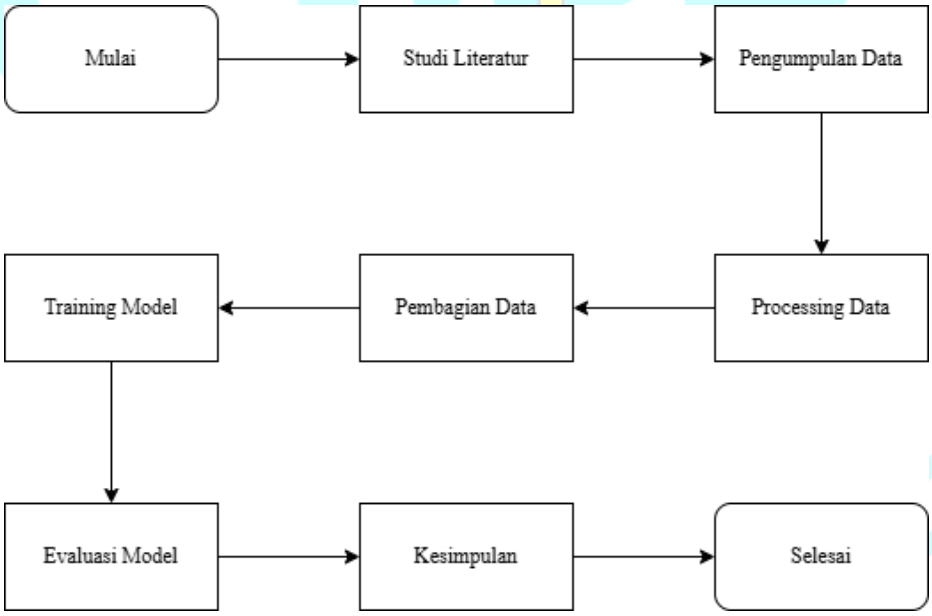
Tabel 3. 1 Waktu pelaksanaan penelitian

NO	Kegiatan	Bulan						
		1	2	3	4	5	6	7
1.	Studi literatur							
2.	Pengumpulan Data							

NO	Kegiatan	Bulan						
		1	2	3	4	5	6	7
3.	Preprocessing Data							
4.	Pembagian Data							
5.	Pelatihan Model							
6.	Evaluasi Model							
7.	Kesimpulan							

3.2 Prosedur Penelitian

Langkah-langkah penelitian mencakup tahapan mulai dari persiapan hingga analisis hasil, yang dirancang untuk memastikan pelaksanaan penelitian berlangsung secara terstruktur dan akurat. Rangkaian proses yang dilalui dapat ditinjau pada Gambar 3.1 *flowmap* penelitian.



Gambar 3. 1 *Flowmap* penelitian

3.2.1 Studi Literatur

Studi ini dimulai dengan melakukan studi pustaka untuk memperoleh landasan teoretis yang kuat terkait pengolahan data keluhan masyarakat dan analisis tingkat urgensi. Kajian ini berfokus pada algoritma *Random Forest*

dan *Support Vector Machine* (SVM) yang dipilih sebagai metode analisis utama dalam penelitian ini. Studi literatur bertujuan untuk memahami karakteristik algoritma tersebut, mengidentifikasi kelemahan dan kelebihannya, serta menemukan studi kasus yang relevan dengan data keluhan masyarakat. Sumber literatur meliputi jurnal ilmiah, buku referensi, dan dokumen dari basis data akademik yang terpercaya.

3.2.2 Pengumpulan Data

Penelitian ini menggunakan data utama yang terkumpul melalui Tanggap Karawang. Data yang diambil berupa file *Excel* yang berjumlah 6380 memuat informasi tentang keluhan Masyarakat, NO, ID Pengaduan, Tanggal Pengaduan, Sumber, Kategori, Isi Pengaduan, Unit Organisasi, Tanggapan, Kendala, Status. Pada Gambar 3. 2 memuat tentang data keluhan Masyarakat.

	NO	ID Pengaduan	Tanggal Pengaduan	Sumber	Kategori	Isi Pengaduan	Unit Organisasi	Tanggapan	Kendala	Status
0	1	14269	2023-12-31 14:24:21	APLIKASI	AKTA KELAHIRAN	Pengajuan	DINAS KEPENDUDUKAN DAN PENCATATAN SIPIL	OPERATOR 2024-01-13 08:38:35 : Hallo kak Untuk...	NaN	Selesai
1	2	14268	2023-12-31 05:55:16	APLIKASI	KTP	selamat pagi tim tangkar,salam sejahtera. moho...	DINAS KEPENDUDUKAN DAN PENCATATAN SIPIL	OPERATOR 2024-01-13 08:37:33 : Selamat pagi ka...	NaN	Selesai
2	3	14267	2023-12-30 22:17:25	APLIKASI	banjir	Banjir di perumahan shanaya bintang residence ...	BADAN PENANGGULANGAN BENCANA DAERAH	OPERATOR 2024-01-11 17:12:24 : terimakasih tel...	NaN	Selesai
3	4	14266	2023-12-30 20:17:59	APLIKASI	#LAIN-LAIN	Memvalidasi ktp	DINAS KEPENDUDUKAN DAN PENCATATAN SIPIL	OPERATOR 2024-01-13 08:36:13 : Selamat pagi ka...	NaN	Selesai
4	5	14265	2023-12-30 10:27:23	APLIKASI	#LAIN-LAIN	Code QR KTP digitl	DINAS KEPENDUDUKAN DAN PENCATATAN SIPIL	OPERATOR 2024-01-13 08:35:47 : Hallo kak Untuk...	NaN	Selesai

Gambar 3. 2 Memuat data keluhan masyarakat

Sumber : (Dokumen pribadi)

3.2.3 *Preprocessing*

Tahapan *preprocessing* bertujuan guna membersihkan, mempersiapkan, dan mengonversi data mentah ke dalam format yang siap untuk diproses oleh model *machine learning*. Proses ini mencakup beberapa langkah penting untuk memastikan kualitas data dan meningkatkan performa model dalam mempelajari pola dari dataset. Berikut tahapan dari *preprocessing* :

1. Penanganan nilai kosong

   Data yang memiliki nilai kosong di kolom penting, seperti permasalahan, isu, dan status, dihapus untuk memastikan dataset konsisten. Langkah ini mencegah error selama proses pelatihan model. Tujuan nya adalah menghindari bias akibat data yang tidak lengkap dan Memastikan data yang digunakan sudah bersih.

2. Transformasi Data Kategori

   Data kategori pada kolom isu dan status diubah menjadi bentuk numerik menggunakan teknik Label *Encoding*. Transformasi ini penting agar model *machine learning* dapat memahami informasi yang terkandung dalam data kategori.

   Contoh :

   a. Isu : " Infrastruktur Jalan" → 0, "Kesehatan" → 1.
   b. Status : "diterima" → 0, "belum diakomodir" → 1.

   Tujuan nya adalah mengubah data kategori menjadi format yang dapat diproses oleh model.

3. *Preprocessing* Teks Pada Kolom

   Permasalahan Data teks dalam kolom permasalahan diproses lebih lanjut agar menjadi representasi numerik yang optimal. Langkah-langkahnya meliputi :

   a. *Lowercasing* : Semua teks diubah menjadi format huruf kecil agar memastikan konsistensi.

      Contoh : "Sering Terjadi Banjir" menjadi " sering terjadi banjir"

   b. *Tokenizing* : Teks dibagi menjadi kata-kata terpisah atau token.

      Contoh : "sering terjadi banjir" menjadi ["sering", "terjadi", "banjir"].

   c. *Filtering Stopwords* : Kata-kata biasa yang tidak relevan, misalnya "dan", "di", "adalah", dihapus menggunakan pustaka *stopwords* untuk Bahasa Indonesia.

      Contoh : ["sering", "terjadi", "banjir", "di"] menjadi ["sering", "terjadi", "banjir"].

   d. *Stemming* : Setiap kata diubah menjadi bentuk dasarnya menggunakan algoritma *stemming*.

      Contoh : ["terjadi", "banjir"] menjadi ["jadi", "banjir"].

c. Transformasi TF-IDF : Hasil teks yang telah diproses digunakan untuk menghitung bobot TF-IDF, yang memberikan nilai numerik pada setiap kata berdasarkan relevansinya dalam *corpus*.

4. Gabungan Fitur Hasil transformasi TF-IDF dari teks digabungkan dengan encoding kategori dari kolom isu.
   Contoh : Representasi teks sebagai vektor [0.12, 0.05, 0.08, ...] digabungkan dengan *encoding* isu, misalnya 1.

5. Pembagian dataset : Kumpulan data dipisahkan menjadi data latih dan data uji, umumnya dengan rasio 80:20. Data latih dimanfaatkan untuk membangun model, sedangkan data uji dipakai guna menilai kinerja model.

3.2.4 Pembagian Data

Menurut Pokhrel (2024), Pada percobaan awal dengan pemisahan data 90:10, kami berhasil memperoleh nilai akurasi mencapai 0,786 untuk metode TF-IDF dan 0.766 untuk metode Sastrawi. Hasil tersebut mengindikasikan bahwa model dapat mengklasifikasikan ulasan dengan tingkat ketepatan yang memadai, walaupun tetap ada peluang dalam rangka peningkatan selanjutnya.

Percobaan kedua dengan pembagian data 80:20 memberikan akurasi yang hampir sama, yaitu 0.788 untuk TF-IDF dan 0.773 untuk Sastrawi. Hal ini menunjukkan bahwa pembagian data ini memberikan hasil yang stabil dan konsisten dalam mempertahankan performa model.

Sementara itu, pada percobaan dengan pembagian data 70:30, meskipun akurasi tetap tinggi dengan nilai 0.766 untuk TF-IDF dan 0.753 untuk Sastrawi, terlihat sedikit penurunan dibandingkan dengan percobaan sebelumnya. Ini menyarankan bahwa proporsi data latih yang lebih kecil dapat mempengaruhi kemampuan model dalam mengeneralisasi data uji.

Pembagian data dengan rasio 80:20 merupakan pilihan yang paling sering dipakai karena menyediakan keseimbangan optimal antara data pelatihan dan pengujian. Sebanyak 80% dari data digunakan untuk proses pelatihan model, memastikan model memiliki cukup data untuk mempelajari

pola, sementara 20% data uji cukup untuk mengevaluasi performa model. Rasio ini sangat cocok untuk dataset dengan ukuran kecil hingga sedang, seperti dataset keluhan masyarakat dengan 700 sampel.

3.2.5 Pelatihan Model

Proses pelatihan model dilakukan memanfaatkan dua algoritma machine learning, yaitu *Random Forest* dan *Support Vector Machine* (SVM), karena keduanya mampu menangani tugas klasifikasi dengan baik. Data latih yang sudah diproses terlebih dahulu digunakan sebagai bahan pelatihan model. Berikut tahapan *training* model yang akan dilakukan :

1. Menyiapkan data latih
   a. Data latih (80%) yang telah diproses digunakan untuk melatih model.
   b. Kolom permasalahan ditransformasi menggunakan TF-IDF, sedangkan kolom isu *diencoding*. Keduanya digabungkan menjadi matriks fitur.

2. Pelatihan model *Random Forest*
   a. Model dilatih dengan membentuk sejumlah pohon keputusan berdasarkan subset data yang dipilih secara acak.
   b. Prediksi akhir diperoleh berdasarkan suara terbanyak dari seluruh pohon keputusan.
   c. Parameter seperti jumlah pohon (*n_estimators*) disesuaikan untuk mengoptimalkan performa.

3. Pelatihan model SVM
   a. Model SVM dilatih menggunakan *kernel Radial Basis Function* (RBF) untuk memisahkan kelas target secara non-linear.
   b. Parameter seperti C (pengaturan *margin*) dan gamma (pengaruh data) dioptimasi agar model bekerja lebih baik.

4. Validasi dan Evaluasi
   a. Model diuji pada data uji (20%) untuk mengevaluasi performa.

b. Prediksi model dibandingkan dengan target asli, dan metrik seperti akurasi, *precision*, *recall*, dan *F1-score* dihitung.

3.2.6 Evaluasi Model

Proses evaluasi model dilakukan guna mengukur performa algoritma *Random Forest* dan SVM dalam memprediksi status keluhan masyarakat berdasarkan data pengujian. Evaluasi ini bertujuan memastikan bahwa model tidak sekadar efektif pada data latih, serta dapat mengidentifikasi pola secara akurat pada data yang belum pernah dilihat sebelumnya.

Langkah-langkah tahapan pada evaluasi model sebagai berikut:

1. Prediksi pada data uji

   Model *Random Forest* dan SVM yang telah dilatih digunakan untuk memprediksi status keluhan masyarakat pada data uji. Prediksi ini dibandingkan dengan nilai target asli untuk mengukur performa model. Proses ini bertujuan untuk mengevaluasi kemampuan model untuk mengaplikasikan pola yang telah dipelajari terhadap data baru yang belum pernah dikenali sebelumnya.

2. Menghitung metrik evaluasi

   a. Akurasi : Menilai persentase prediksi akurat berdasarkan keseluruhan data uj. Akurasi menyajikan ringkasan mengenai performa model dalam membuat prediksi, tetapi kurang efektif jika data tidak seimbang.

$$Akurasi = \frac{Jumlah\ Prediksi\ Benar}{Total\ Data\ Uji} \quad (1)$$

   b. *Precision : Precision* menunjukkan proporsi prediksi benar untuk suatu kelas dibandingkan dengan semua prediksi untuk kelas tersebut. *Precision* penting jika kesalahan jenis *false positive* (prediksi salah sebagai benar) perlu diminimalkan.

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive} \quad (2)$$

c. *Recall* : *Recall* mengukur kemampuan model untuk menemukan semua data dari kelas tertentu. *Recall* penting jika kesalahan jenis *false negative* (prediksi benar dianggap salah) perlu diminimalkan.

$$Recall = \frac{True\ Positive}{True\ Positive + False\ Negaitve} \quad (3)$$

d. *F1-Score* : *F1-Score* merupakan nilai rata-rata harmonis antara *precision* dan *recall*, menghasilkan keseimbangan antara keduanya. *F1-Score* berguna ketika data tidak seimbang, karena menggabungkan kekuatan *precision* dan *recall*.

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (4)$$

Dengan menghitung metrik-metrik ini, performa model dapat dievaluasi secara menyeluruh, baik dalam menghindari kesalahan prediksi maupun dalam mengenali pola pada data. Evaluasi akurasi, presisi, *recall* dan *f1-score* dapat dilihat pada Gambar 3.3

	precision	recall	f1-score	support
0	1.00	0.90	0.95	20
1	0.85	1.00	0.92	11
Accuracy			0.94	101
Macro avg	0.92	0.95	0.93	101
Weightad avg	0.95	0.84	0.94	101

Gambar 3. 3 Evaluasi akurasi, presisi, *recall*, *f1-score*

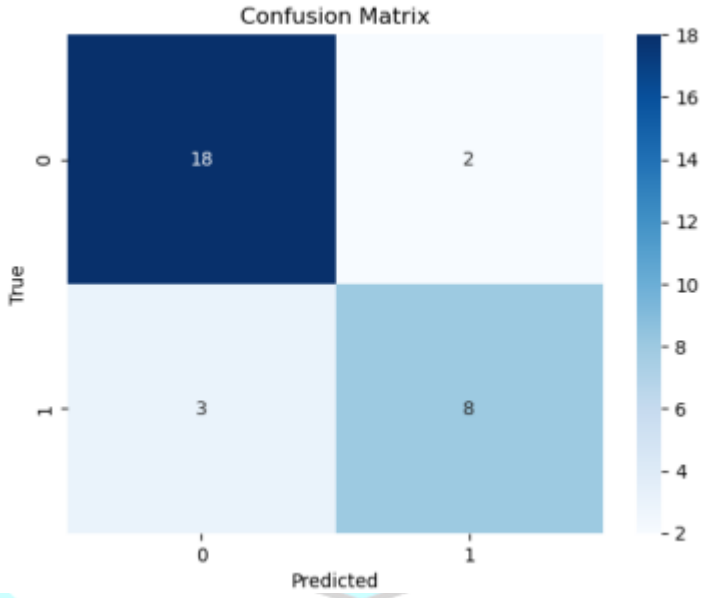
Sumber : (Satrio Junaidi, Valicia Anggela and Kariman, 2024)

3. *Classification Report*

Classification report menyajikan nilai *precision, recall, F1-score,* dan *support* (jumlah data per kelas) untuk setiap kategori target (diterima dan belum diakomodir). Laporan ini membantu mengidentifikasi kelemahan model dalam menangani kelas tertentu, misalnya jika salah satu kelas memiliki *precision* atau *recall* yang rendah.

4. *Confusion matrix*

   Confusion matrix digunakan untuk memvisualisasikan jumlah prediksi benar dan salah untuk setiap kelas target. Matriks ini menampilkan nilai *True Positive, False Positive, True Negative,* dan *False Negative,* sehingga memberikan pemahaman lebih mendalam tentang bagaimana model mengklasifikasikan data. *Confusion Matrix* dapat dilihat pada Gambar 3.4.



Gambar 3. 4 *Confusion Matrix*

Sumber : (Satrio Junaidi, Valicia Anggela *and* Kariman, 2024)

3.2.7 Kesimpulan

Tahap ini bertujuan untuk menyajikan hasil penelitian dalam bentuk yang mudah dipahami. Kesimpulan disusun berdasarkan hasil evaluasi kedua model, dengan visualisasi perbandingan performa antara *Random Forest* dan SVM. Berikut contoh langkah -langkah penyelesaian berbentuk rumus dari *Random Forest* dan SVM :

1. Contoh data
   a. Kolom permasalahan : sering terjadi banjir dikarenakan jalanannya rendah
   b. Kolom isu : infrastruktur jalan
   c. Kolom status : diterima atau belum diakomodir.

2. Rumus dari *Random Forest*

   Random Forest memprediksi target dengan membangun banyak *decision tree* dan menggunakan voting mayoritas untuk menentukan hasil akhir :

$$\hat{y}RF = Mode\,(T_1(x), T_2(x), \ldots, T_k(x)) \quad (1)$$

   Dimana :

   a. $T_i(x)$: Prediksi dari pohon ke-i untuk *input* x

   b. k : jumlah pohon dalam hutan.

3. Rumus dari SVM

$$\hat{y}SVM = sign\,(w.x + b) \quad (2)$$

   Dimana :

   a. w : Vektor bobot *hyperplane*.

   b. x : Vektor fitur *input*.

   c. b : Bias *hyperplane*.

Contoh proses *Random Forest* :

1. *Input* data

   a. Fitur: TF-IDF dari teks "sering terjadi banjir dikarenakan jalanannya rendah" dan *encoding* untuk kategori "infrastruktur jalan".

   b. Target: status.

2. *Output* dari Setiap *Decision Tree*

   a. Pohon 1: Prediksi = diterima.

   b. Pohon 2: Prediksi = belum diakomodir.

   c. Pohon 3: Prediksi = diterima.

3. Hasil : "diterima" (karena sebagian besar pohon memprediksi status ini).

Kesimpulan dari proses model *Random Forest* memberikan hasil prediksi (diterima) berdasarkan pola umum dalam data latih yang mencakup teks dan kategori isu terkait.

Contoh proses SVM :

1. *Input* data
   a. Fitur: TF-IDF dari teks "sering terjadi banjir dikarenakan jalanannya rendah" dan *encoding* untuk kategori "infrastruktur jalan".
   b. Target: status.

2. *Hyperplane* : SVM mencari *hyperplane* yang memperbesar jarak pemisah antara dua kelas (diterima dan belum diakomodir).

3. Prediksi
   a. Jika $w \cdot x + b > 0$, maka prediksi = diterima.
   b. Jika $w \cdot x + b \leq 0$, maka prediksi = belum diakomodir
   c. Dalam kasus ini, $w \cdot x + b = 2.5$, sehingga prediksi = diterima

Kesimpulan dari proses model SVM memutuskan status (diterima) berdasarkan margin maksimal antara kelas dan posisi fitur pada *hyperplane*. Dari proses algoritma *Random Forest* dan SVM dapat di simpulkan dengan Tabel 3.2

Tabel 3. 2 Perbandingan kesimpulan

Aspek	***Random Forest***	**SVM**
Pendekatan	Voting dari banyak pohon keputusan	Mencari hyperplane terbaik untuk memisahkan kelas.
Prediksi Akhir	Diterima	Diterima
Kelebihan	Toleransi tinggi terhadap *outlier* dan fitur yang banyak.	Baik untuk data dengan margin yang jelas.
Kekurangan	Bisa kurang akurat pada data non-linear.	Sensitif terhadap outlier.

Dalam contoh data “sering terjadi banjir dikarenakan jalanannya rendah” dengan kategori “infrastruktur jalan”, baik *Random Forest* maupun SVM memprediksi

status keluhan sebagai “diterima”. Perbedaan pendekatan keduanya mencerminkan keunggulan masing-masing algoritma, *Random Forest* bekerja dengan voting dari banyak *decision tree*, sedangkan SVM memisahkan data dengan *hyperplane* yang memaksimalkan *margin*.