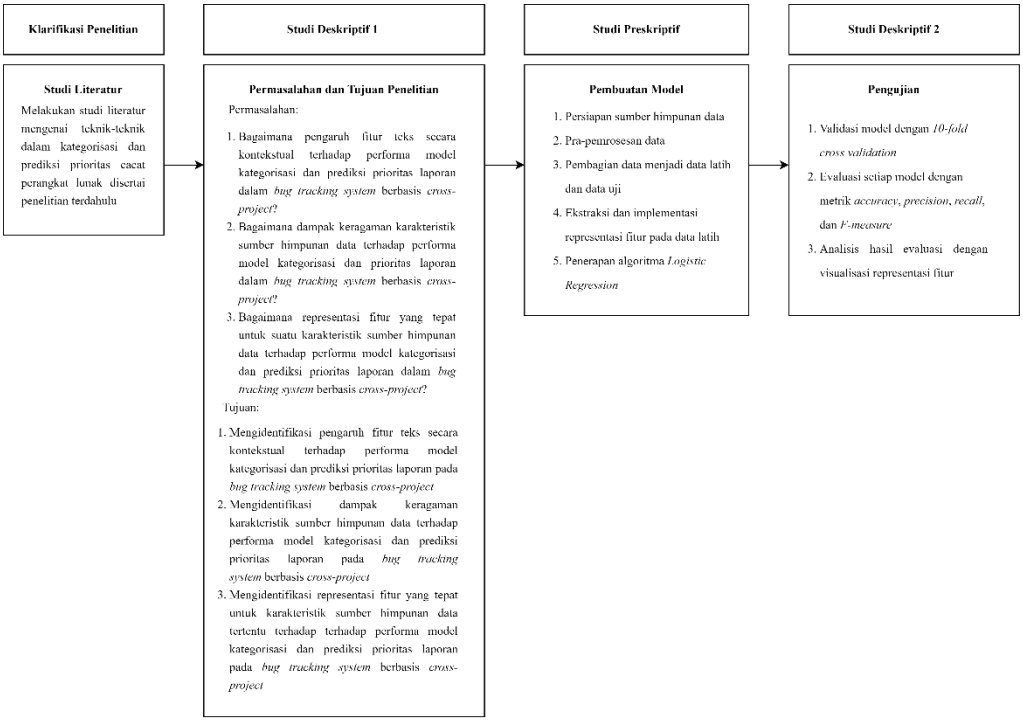
BAB III
METODE PENELITIAN

3.1 Desain Penelitian

Penelitian ini dirancang menggunakan *Design Research Methodology* (DRM) yang terdiri atas klarifikasi penelitian, studi deskriptif 1, studi preskriptif, dan studi deskriptif 2 (Blessing & Chakrabarti, 2009). Berdasarkan DRM tersebut, skema penelitian dirancang sebagaimana terdapat pada Gambar 3.1.



Gambar 3.1 Skema Penelitian

Berdasarkan skema penelitian di atas, berikut adalah deskripsi dari masing-masing tahapan:

1. Klarifikasi Penelitian

   Studi literatur dilakukan untuk mengidentifikasi permasalahan penelitian terkait kategorisasi dan prediksi prioritas laporan pada BTS berbasis *cross-project*. Berbagai artikel jurnal dan prosiding yang dapat diandalkan digunakan sebagai rujukan dalam mengidentifikasi permasalahan tersebut. Penelitian-penelitian dengan implementasi yang relevan dan berpotensi untuk diimplementasikan dalam penelitian ini juga dijadikan

sebagai rujukan. Buku-buku yang relevan untuk memperkuat landasan teori digunakan sehingga penelitian tetap terarah.

2. Studi Deskriptif 1

   Tahapan berikutnya adalah mendefinisikan permasalahan dan tujuan penelitian yang berlandaskan hasil studi literatur yang dilakukan pada tahap sebelumnya. Tahapan ini akan dijadikan landasan dalam merancang dan membangun solusi atas hal tersebut.

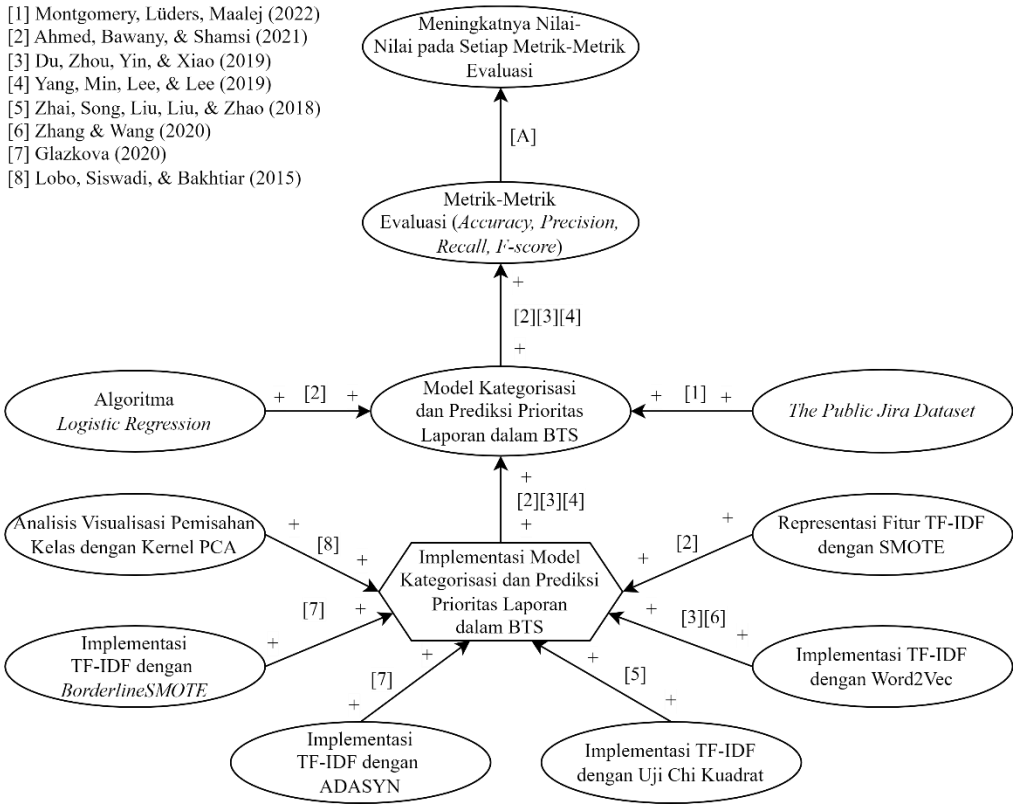
3. Studi Preskriptif

   Pada tahapan ini, langkah-langkah dalam pembuatan model untuk kategorisasi dan prediksi prioritas laporan pada BTS diterapkan. Langkah-langkah tersebut mengacu pada penelitian terdahulu yang kemudian dimodifikasi untuk meningkatkan performa model yang dihasilkan.

4. Studi Deskriptif 2

   Pada tahapan terakhir, performa dari setiap model yang dihasilkan akan dibandingkan satu dengan yang lainnya. Kemampuan generalisasi dari model yang dihasilkan juga akan diukur menggunakan *10-fold cross validation*.

Studi preskriptif dan studi deskriptif 2 dilakukan untuk menganalisis perbandingan representasi fitur yang merujuk pada kerangka kerja *CaPBug* dengan mengimplementasikan metode-metode yang telah diterapkan pada penelitian-penelitian terkait. Hal tersebut diusulkan untuk mengidentifikasi hasil yang dapat diperoleh jika menggunakan representasi fitur tertentu, mengingat bahwa performa suatu model dapat dipengaruhi oleh karakteristik suatu sumber himpunan data. Gambar 3.2 menunjukkan model dampak yang digunakan dalam penelitian ini.



Gambar 3.2 Model Dampak yang Disusun

3.2 Alat dan Bahan Penelitian

Alat yang digunakan dalam penelitian ini adalah dengan spesifikasi sebagai berikut:

1. Processor Intel(R) Core (TM) i7-1065G7
2. RAM 16GB
3. Sistem operasi Windows 10 (Build 19044.2251 Version 21H2)

Perangkat lunak yang digunakan dalam penelitian adalah *Visual Studio Code* sebagai *Integrated Development Environment* (IDE). Bahan yang digunakan dalam penelitian ini sumber himpunan data publik yang diperoleh dari penelitian terkait. Sumber himpunan data yang bersifat publik lebih sering digunakan karena dapat digunakan kembali untuk penelitian-penelitian di masa yang akan datang (Jorayeva, Akbulut, Catal, & Mishra, 2022). Sumber himpunan data yang digunakan adalah *the Public Jira Dataset* yang terdiri dari 16 repositori publik Jira dengan 1822 proyek yang terdiri atas 2,7 juta laporan

dengan total 32 juta perubahan, 9 juta komentar, dan 1 juta keterhubungan antarlaporan (Montgomery, Lüders, & Maalej, 2022).

Tabel 3.1 Deskripsi Repositori pada the Public Jira Dataset

No.	Nama Repositori	Deskripsi
1.	Apache	Perangkat lunak server situs web
2.	Hyperledger	*Ledgers* terdistribusi berbasis *blockchain*
3.	IntelDAOS	Solusi penyimpanan terdistribusi
4.	JFrog	Sekumpulan *tools DevOps* yang bersifat universal
5.	JIRA	Perangkat lunak untuk pelacakan laporan dan manajemen proyek
6.	JiraEcosystem	Keseluruhan ekosistem untuk JIRA
7.	MariaDB	*Relational Database Management System* (RDBMS)
8.	Mindville	Perangkat lunak untuk pelacakan dan manajemen operasional bisnis
9.	Mojang	Pengembangan gim
10.	MongoDB	*NoSQL Database Management System*
11.	Qt	*Framework* untuk pengembangan aplikasi berbasis *cross-platform* seperti *desktop*, *embedded*, dan *mobile*
12.	RedHat	Produk-produk perangkat lunak bersifat *open source* berbasis *Linux*
13.	Sakai	Perangkat lunak untuk manajemen, komunikasi, dan kolaborasi dalam ranah pendidikan
14.	SecondLife	Gim berbasis dunia virtual
15.	Sonatype	Perangkat lunak untuk mengidentifikasi kerentanan dalam penggunaan *library* bersifat *open source* dan meluncurkan baris kode yang lebih aman
16.	Spring	*Framework* untuk pengembangan proyek

Mengingat bahwa spesifikasi perangkat yang digunakan belum memadai untuk memproses seluruh data yang terdapat pada sumber himpunan data tersebut, repositori yang digunakan adalah Mindville, JFrog, dan Hyperledger.

Justifikasi dari dipilihnya ketiga repositori terdapat pada hasil dan pembahasan. Distribusi dari sampel sumber himpunan data tersebut terdapat pada Tabel 3.2. Distribusi selengkapnya terdapat pada Lampiran 1. Singkatan-singkatan yang terdapat pada tabel tersebut di antaranya adalah DIT (*Documented Issue Types*), UIT (*Used Issue Types*), DLT (*Documented Link Types*), ULT (*Used Link Types*), Ch/I (*Changes per Issue*), Co/I (*Comments per Issue*), dan UP (*Unique Projects*). Deskripsi dari atribut-atribut pada sumber himpunan data tersebut selengkapnya terdapat pada Lampiran 2.

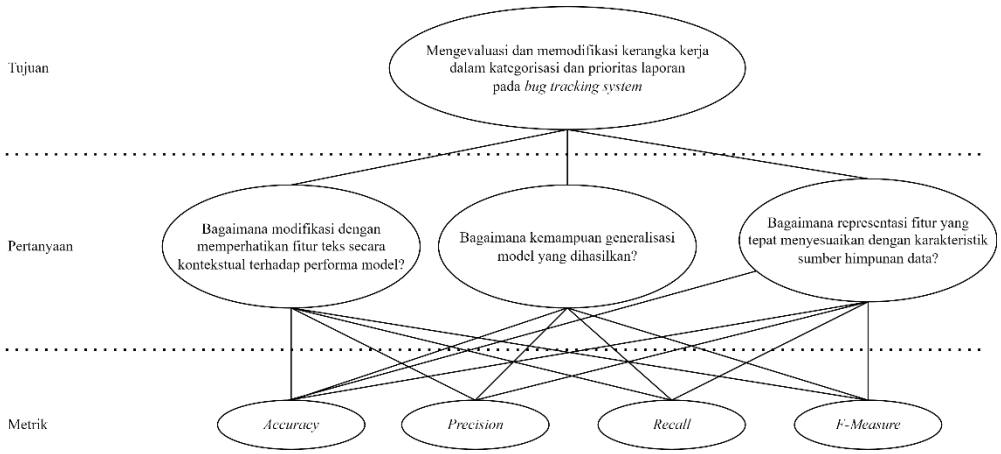
Tabel 3.2

Distribusi Sampel pada the Public Jira Dataset

Nama Repositori	Tahun Lahir	Jumlah Laporan	DIT	UIT	*Links*	DLT	ULT	Ch/I	Co/I	UP
Mindville	2015	2.134	2	2	46	N/A	4	3	N/A	10
JFrog	2006	15.535	30	22	3.303	17	10	9	1	35
Hyperle-dger	2016	28.146	9	9	16.846	6	6	12	2	36

3.3 Instrumen Penelitian

Instrumen yang digunakan dalam penelitian ini dapat dibagi menjadi dua, yaitu *library* yang digunakan untuk membuat dan mengevaluasi model serta rumus-rumus terkait untuk mengukur tingkat keberhasilan model yang dihasilkan. Bahasa pemrograman yang akan digunakan adalah bahasa pemrograman *Python* versi 3.9. Instrumen tersebut direpresentasikan dalam *goal question metric* pada Gambar 3.2.



Gambar 3.3 *Goal Question Metric* yang Digunakan

Library yang digunakan terdapat pada Tabel 3.3. Kemudian, rumus-rumus yang digunakan sebagi metrik-metrik evaluasi terdapat pada Tabel 3.4. Rumus-rumus tersebut mengacu pada *confusion matrix* di Tabel 3.5.

Tabel 3.3

Library yang Digunakan

No.	*Library*	Deskripsi Kegunaan
1.	*pandas*	Mempersiapkan dan memanipulasi data
2.	*numpy*	Memanipulasi data
3.	*scipy*	Perhitungan saintifik
4.	*requests*	Memanggil HTTP *request* untuk memperoleh sekumpulan *stop words* dari GitHub
5.	*Natural Language Toolkit* (NLTK)	Melakukan pra-pemrosesan data dalam pemrosesan bahasa alami
6.	*re*	Memanfaatkan *regular expression* pada tahap pra-pemrosesan data
7.	*flashtext*	Mengekstraksi kata-kata pada seluruh dokumen dan dimanfaatkan pada perhitungan uji chi kuadrat
8.	*scikit-learn*	Menerapkan pemisahan data latih dan data uji, algoritma *Logistic Regression*, *10-fold cross validation*, PCA, dan evaluasi model dengan metrik-metrik yang telah ditentukan

9.	*imbalanced-learn*	Mengimplementasikan SMOTE, ADASYN, dan *BorderlineSMOTE*
8.	*gensim*	Mengimplementasikan Word2Vec
10.	*pickle*	Mengekspor dan mengimpor kembali hasil model yang telah diperoleh
11.	*matplotlib*	Menganalisis melalui visualisasi distribusi sumber himpunan data yang digunakan

Tabel 3.4

Metrik-Metrik Evaluasi

No.	Nama Metrik	Rumus
1.	*Accuracy*	$\frac{\mathrm{TP}+\mathrm{TN}}{\mathrm{TP}+\mathrm{TN}+\mathrm{FP}+\mathrm{FN}}$
2.	*Precision*	$\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}}$
3.	*Recall*	$\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}}$
4.	*F-score*	$\frac{2 \times Precision \times Recall}{Precision + Recall}$

Tabel 3.5

Confusion Matrix

Jenis Kelas	Aktual	
Prediksi	TP (*True Positive*)	FP (*False Positive*)
	TN (*True Negative*)	FN (*False Negative*)

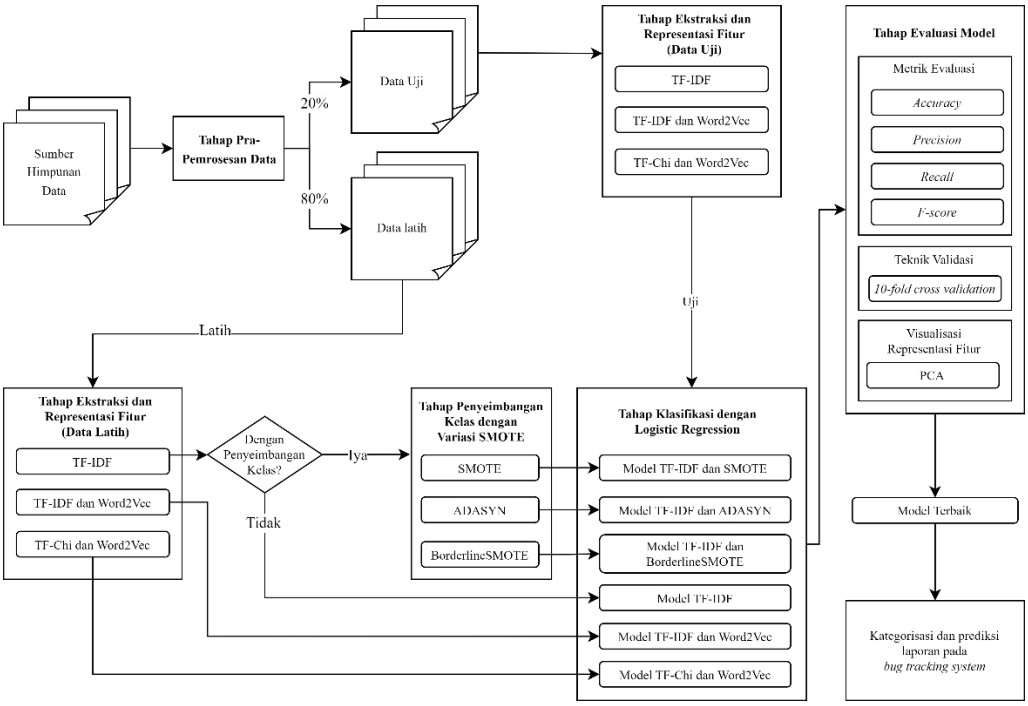
Confusion matrix di atas merupakan acuan dalam mengukur performa model yang akan dihasilkan. Permasalahan yang dihadapi adalah klasifikasi multikelas sehingga ukuran dari *confusion matrix* menyesuaikan dengan jumlah kelas yang terdapat pada masing-masing permasalahan yang akan diselesaikan. Kelas beserta deskripsinya yang terdapat pada permasalahan prioritas cacat perangkat lunak terdapat pada Tabel 3.6. Contoh penggunaan *confusion matrix* terdapat pada Lampiran 3.

Tabel 3.6 Tingkat Prioritas Cacat Perangkat Lunak
pada the Public Jira Dataset

No.	Tingkat Prioritas	Deskripsi
1.	P1	*Blocker*, laporan yang disampaikan akan memblokir kemajuan pengembangan perangkat lunak
2.	P2	*Critical*, laporan yang disampaikan mengandung permasalahan serius sehingga berpotensi besar memblokir kemajuan pengembangan perangkat lunak
3.	P3	*Major*, laporan yang disampaikan mengandung permasalahan yang berpotensi memblokir kemajuan pengembangan perangkat lunak
4.	P4	*Normal*, laporan yang disampaikan mengandung permasalahan umum yang berpotensi mengganggu kemajuan pengembangan perangkat lunak
5.	P5	*Minor*, laporan yang disampaikan mengandung permasalahan kecil yang dapat diselesaikan secara mudah
6.	P6	*Trivial*, laporan yang disampaikan mengandung permasalahan kecil atau bahkan tidak berdampak terhadap kemajuan pengembangan perangkat lunak

3.4 Prosedur Penelitian

Prosedur penelitian yang diimplementasikan adalah analisis perbandingan representasi fitur yang merujuk pada kerangka kerja *CaPBug* yang sebagaimana terdapat pada Gambar 3.3.



Gambar 3.4 Prosedur Penelitian

Berikut adalah langkah-langkah yang diimplementasikan pada prosedur tersebut:

1. Persiapan Data

   Pada tahap ini, data dari setiap repositori pada basis data MongoDB akan disimpan ke dalam format *comma-separated values* (.CSV). Kemudian, masing-masing file akan diimpor ke dalam *Visual Studio Code*. Tahap ini dilakukan untuk menentukan repositori-repositori yang akan dianalisis dalam tahap berikutnya.

2. Pra-Pemrosesan Teks

   Pra-pemrosesan teks dilakukan terhadap data dari setiap repositori agar memiliki format yang bersih, konsisten, dan siap untuk dianalisis lebih lanjut dengan harapan dapat memberikan performa yang lebih baik secara signifikan (Lourdusamy & Abraham, 2018).

3. Pembagian Data Latih dan Data Uji

   Pembagian data latih dan data uji dilakukan untuk memisahkan data yang digunakan dalam proses pelatihan dengan data yang digunakan dalam mengevaluasi hasil pelatihan. Data latih dan data uji yang dihasilkan dari setiap repositori dapat dibagi menjadi dua jenis, yaitu data dengan label kelas berupa kategori dan prioritas. Implementasi dari pembagian data tersebut menggunakan *stratified random sampling*, yaitu teknik untuk membagi suatu populasi menjadi sekumpulan kelompok yang disebut strata (dalam konteks ini adalah label kelas) dan mengambil sebagian data secara acak dari masing-masing stratum (R. Singh & Mangat, 1996). Hal tersebut dilakukan agar setiap label kelas diwakili dalam data latih dan data uji.

4. Ekstraksi dan representasi fitur

   Setiap data latih ditransformasikan ke dalam representasi numerik agar metode-metode komputasional dapat diimplementasikan. Representasi fitur yang digunakan mencakup TF-IDF, Word2Vec dengan pembobotan TF-IDF, dan Word2Vec dengan pembobotan TF-CHI. Khusus untuk representasi TF-IDF, teknik SMOTE dan variasinya (ADASYN dan *BorderlineSMOTE*) juga diimplementasikan. Kemudian, representasi Word2Vec akan menggunakan nilai *window* 2 hingga 6.

5. Klasifikasi dan evaluasi model

   Setiap representasi fitur akan dilatih menggunakan algoritma *Logistic Regression*. Teknik validasi menggunakan *10-fold cross validation* diimplementasikan sehingga masing-masing representasi fitur akan menghasilkan 10 model. Model yang terbaik dari setiap representasi fitur

akan dibandingkan dengan model yang lainnya berdasarkan metrik-metrik evaluasi yang telah dipaparkan.

6. Analisis hasil

   Hasil dari tahap sebelumnya akan dianalisis lebih lanjut pada tahap ini. Model dengan representasi fitur tertentu dapat menjadi model yang terbaik pada suatu karakteristik sumber himpunan data, namun tidak berlaku untuk sumber himpunan data dengan karakteristik lainnya.

3.5 Analisis Data

Analisis data dilakukan menggunakan *Microsoft Excel* untuk keperluan visualisasi terkait jumlah data berdasarkan label kelas, baik kategori maupun prioritas. Hasil dari setiap metrik evaluasi diperoleh secara langsung menggunakan *library scikit-learn*. Visualisasi terhadap karakteristik sumber himpunan data berdasarkan representasi fitur dilakukan menggunakan *library matplotlib* dan didukung juga dengan *library scikit-learn*, spesifiknya menggunakan modul *principal component analysis* (PCA).