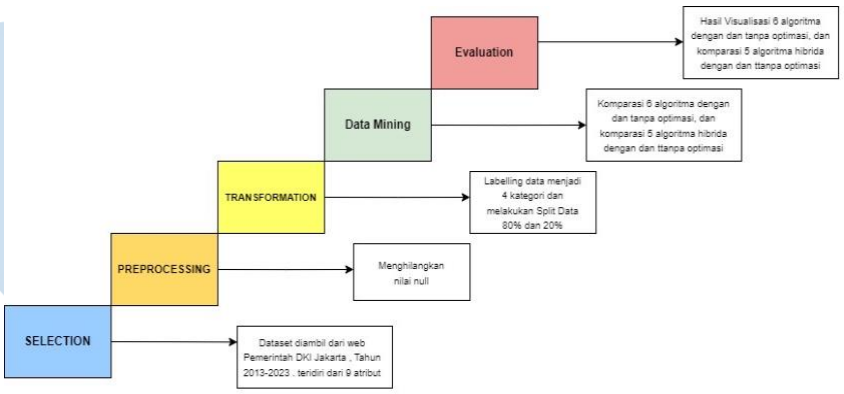
BAB III

METODOLOGI PENELITIAN

3.1 Gambaran Umum Objek Penelitian

Objek penelitian pada penelitian ini adalah polusi udara di wilayah Jakarta. Penelitian ini memanfaatkan data Indeks Standar Pencemaran Udara di wilayah Jakarta yang diukur dari lima Stasiun Pemantau Kualitas Udara (SPKU) di Provinsi DKI Jakarta yang diperoleh dari laman https://satudata.jakarta.go.id/ , dengan rincian data dari tahun 2013-2023. Tujuan dari penelitian ini adalah untuk mengembangkan metode klasifikasi tingkat kualitas udara di Jakarta dengan menggunakan algoritma *machine learning* dan untuk menentukan tingkat kualitas udara di Jakarta . Hasil yang diharapkan dari penelitian ini untuk memberikan wawasan baru bagi para peneliti dan memberikan kontribusi positif terhadap upaya pengelolaan lingkungan di Jakarta. Informasi yang diberikan oleh model dapat digunakan untuk mendukung kebijakan dan tindakan yang lebih tepat dalam mengurangi dampak pencemaran udara.

3.2 Alur Penelitian



Gambar 3. 1 Alur Penelitian KDD

Pada Gambar 3.1 merupakan Alur penelitian pada penelitian ini disesuaikan dengan *framework Knowledge Discovery in Databases* (KDD).

3.2.1 Pre-KDD

Pada tahap Pre-KDD dimulai dengan memperkenalkan dataset sebelum memasuki proses *Knowledge Discovery in Database*. Tahap ini mencakup penjelasan detail tentang dataset yang akan digunakan, termasuk *library* yang digunakan, deskripsi singkat tentang atribut di dalam dataset, dan visualisasi grafik untuk memberikan gambaran awal tentang dataset tersebut. Setelah pemahaman dasar tentang dataset terbentuk, langkah selanjutnya adalah melakukan penelitian yang sesuai dengan *framework* KDD, yang melibatkan penerapan model data *mining* untuk menggali pengetahuan dari dalam *database*, Berdasarkan Gambar 3.1 yang merupakan alur penelitian menggunakan *framework* KDD, tahapan KDD terbagi menjadi lima tahapan, yang akan diuraikan sebagai berikut.

3.2.2 Data *Selection*

Tahap awal dalam proses KDD adalah *Data Selection*, yaitu proses pemilihan data dari sumber data operasional . Sebelum mengumpulkan data untuk penelitian ini, langkah awal dilakukan dengan melakukan penelusuran literatur jurnal terkait klasifikasi kualitas udara. Tujuannya adalah untuk memperoleh informasi yang dapat digunakan sebagai referensi dalam penelitian ini. Data yang digunakan pada penelitian ini adalah data Indeks Standar Pencemar Udara tahun 2013-2023. Dataset yang diperoleh berupa Excel yang terdiri dari 10.476 data, dan memilik 9 variabel secara keseluruhan. Pada Tabel 3.1 merupakan variabel dataset ISPU DKI Jakarta.

Tabel 3. 1 Variabel Dataset ISPU DKI Jakarta

Variabel	Keterangan
pm10	Mencatat nilai partikulat PM10, salah satu parameter yang diukur.
pm25	Mencatat nilai partikulat PM10, salah satu parameter yang diukur.
so2	Mencatat nilai partikulat so2, salah satu parameter yang diukur.
co	Mencatat nilai partikulat co, salah satu parameter yang diukur.
o3	Mencatat nilai partikulat o3, salah satu parameter yang diukur.

no2	Mencatat nilai partikulat no2, salah satu parameter yang diukur.
max	Mencatat nilai ukur paling tinggi dari seluruh parameter yang diukur dalam waktu yang sama.
critical	Mencatat parameter yang hasil pengukurannya paling tinggi.
categori	Menyimpan informasi kategori hasil perhitungan Indeks Standar Pencemaran Udara.

3.2.3 Data *Pre-processing*

Tahap kedua dalam proses penelitian ini adalah *Data Pre-processing*, yang mencakup *data cleansing* . Dalam tahap *data cleansing* , fokusnya adalah membersihkan data mentah untuk meningkatkan kualitas data secara keseluruhan. Proses *data cleansing* pada penelitian ini dengan membersihkan nilai *null*. *Data cleansing* bertujuan untuk meningkatkan kehandalan data yang akan digunakan dalam analisis.

3.2.4 Data *Tranformation*

Tahap ketiga dalam penelitian ini adalah *Data Transformation*, Pada tahap ini, dibagi menjadi dua bagian yaitu *labelling* dan split data. Penelitian ini akan melakukan *labelling* data menjadi empat kategori . Selanjutnya pada bagian split data, dataset akan dibagi menjadi dua bagian, yaitu data *training* dan data *testing*. Pembagian ini dilakukan dengan persentase 80% untuk data *training* dan 20% untuk data *testing*. Pembagian persentase data 80 % dan 20 % ini merujuk pada penelitian [21] yang menyatakan bahwa dari hasil pengujian yang telah dilakukan pada semua skenario, akurasi tertingi didapatkan dalam *splitting data* data 80:20 . Setelah split data dilakukan, langkah selanjutnya adalah Data *Mining*.

3.2.5 Data Mining

Tahap keempat dalam penelitian ini adalah *Data Mining*, yang merupakan suatu proses di mana pola atau informasi menarik dalam data terpilih diidentifikasi menggunakan teknik atau metode khusus [37]. Pemilihan metode atau algoritma yang sesuai sangat mempengaruhi pada tujuan dan keseluruhan proses *Knowledge Discovery in Databases* (KDD). Model yang dikembangkan akan disesuaikan dengan dataset yang

digunakan, dengan harapan menghasilkan hasil sesuai dengan tujuan penelitian ini.

Dalam penelitian ini, akan dilakukan perbandingan antara Algoritma KNN, *Naïve Bayes , SVM, Decision Tree, Random Forest* dan XGBoost dan algoritma hibrida KNN-XGBoost, *Naïve Bayes*-XGBoost, SVM-XGBoost, *Decision Tree*-XGBoost, dan *Random Forest*-XGBoost sebelum dan sesudah menggunakan optimasi. Pemilihan algoritma ini didasarkan pada penelitian [10] , [11] , dan [58] dan akan dilakukan dengan menggunakan bantuan *python*.

3.2.6 Interpretation/ Evaluation

Tahap kelima merupakan *Interpretation / Evaluation*, yang bertujuan untuk memastikan apakah model atau informasi yang ditemukan sesuai dengan tujuan penelitian atau tidak. Pada tahap ini, evaluasi dilakukan terhadap keefektifan dan kualitas model sebelum penggunaannya, dengan tujuan menentukan apakah model dapat mencapai tujuan yang telah ditetapkan. Dalam penelitian ini, akan dilakukan evaluasi terhadap model dengan mengukur nilai akurasi, presisi, *recall*, dan f1-*score* dari 22 model yang digunakan. Proses evaluasi ini bertujuan untuk mengukur sejauh mana model dapat memberikan hasil yang dapat akurat dengan tujuan penelitian.

3.3 Metode Penelitian

Metode Penelitian merupakan salah satu aspek dalam studi ilmiah. Ada dua teknik penelitian yang umum digunakan yaitu penelitian kualitatif dan penelitian kuantitatif. Pada penelitian ini akan menggunakan penelitian kuantitatif dan penyelesaian dilakukan menggunakan *framework* KDD), dengan memanfaatkaan *Jupyter Notebook* sebagai perangkat lunak dan menggunakan bahasa pemrograman *Python.*

Penelitian ini akan membandingkan kinerja dari 22 model yaitu 6 algoritma KNN, *Naïve Bayes*, SVM, *Decision Tree*, *Random Forest*, dan XGBoost dengan sebelum menggunakan optimasi dan setelah menggunakan optimasi. Serta membandingkan 5 algoritma hibrida KNN-XGBoost, *Naïve*

Bayes-XGBoost, SVM-XGBoost, *Decision Tree*-XGBoost, dan *Random Forest*-XGBoost dengan sebelum menggunakan optimasi dan setelah menggunakan optimasi, dalam menghasilkan metrik evaluasi seperti akurasi, presisi, *recall*, dan F1-*score.* Perbandingan antara keenam algoritma tersebut dapat dilihat dalam Tabel 3.2 .

Tabel 3. 2 Perbandingan 6 Algoritma

Algoritma	Kelebihan	Kekurangan
KNN	-. Tangguh dalam menghadapi data *training* yang dipenuhi dengan *noise* dan jumlahnya besar.	-. Perlu menentukan jumlah tetangga terdekat (K) dari data target dalam algoritma KNN.
Naïve Bayes	-. Memerlukan jumlah data *training* yang minim untuk menetapkan parameter yang diperlukan dalam proses klasifikasi.	-. Rentan terhadap data yang tidak seimbang .
SVM	-. Kemampuannya untuk menemukan *hyperplane* yang berbeda untuk memaksimalkan margin antara kelas-kelas yang berbeda.	-. Masalah dengan data yang memiliki atribut yang serupa, yang bisa secara signifikan memengaruhi tingkat akurasi.
Decision Tree	-. Proses pembagian dilakukan berdasarkan atribut yang menunjukkan *information gain* tertinggi, dan memilih atribut yang memberikan kontribusi terbaik.	-. Kecenderungan untuk *overfitting* pada data *training* yang kompleks, sensitif terhadap perubahan kecil dalam data input, dan cenderung membuat model yang kompleks dan sulit untuk diinterpretasi.
Random Forest	-. Mampu mengombinasikan beberapa pohon dan untuk model yang terdiri dari satu pohon tunggal dalam melakukan klasifikasi dan prediksi kelas.	-. Kompleksitas model yang dihasilkan oleh banyak pohon keputusan sulit untuk diinterpretasi, serta membutuhkan sumber daya komputasi yang lebih besar karena proses pelatihan dan

		prediksi yang melibatkan sejumlah besar pohon.
XGBoost	-. Menerapkan model formal yang lebih terstruktur untuk mengatur *overfitting* data.	-. Rentan terhadap *overfitting* terutama jika tidak diatur dengan benar.

Sumber : [DQLab]

Pada Tabel 3.2 merupakan tabel perbandingan antara keenam algoritma yang digunakan pada penelitian ini, Keenam algoritma tersebut masing-masing memiliki kelebihan dan kekurangan. Algoritma KNN memiliki kelebihan dalam menghadapi data *training* yang dipenuhi dengan *noise* dan jumlahnya besar. Namun, kekurangan pada KNN yaitu harus menentukan jumlah tetangga terdekat dari nilai K, Sehingga dapat menjadi tantangan karena hasil perhitungan jarak kurang akurat karena harus memilih nilai pada K[59]. Pada algoritmaa *Naïve Bayes,* kelebihannya adalah memerlukan jumlah data *training* yang minim untuk menetapkan parameter yang diperlukan dalam proses klasifikasi, dan kekurangannya adalah rentan terhadap data yang tidak seimbang yang memungkinkan kesalahan klasifikasi[60].

Algoritma SVM memiliki kelebihan dalam menemukan *hyperplane* yang berbeda untuk memaksimalkan margin antara kelas-kelas yang berbeda, untuk kekurangannya yaitu memiliki masalah dengan data yang memiliki atribut yang serupa, hal ini bisa secara signifikan memengaruhi tingkat akurasi[61]. Kemudian untuk algoritma *Decision Tree* memiliki kelebihan yaitu dapat memilih atribut yang memberikan kontribusi terbaik dalam membuat pembagian yang lebih efektif untuk memprediksi target dari data, adapun kekurangan dari *Decision Tree* yaitu kecenderungan untuk *overfitting* pada data *training* dan cenderung membuat model yang kompleks[50]. Sedangkan pada algoritma *Random Forest* memiliki kelebihan yaitu dalam melakukan klasifikasi dan prediksi kelas, *Random Forest* mampu mengkombinasikan beberapa pohon dan untuk model yang terdiri dari satu pohon tunggal. Sementara, kekurangannya adalah memiliki kompleksitas model yang dihasilkan oleh banyak pohon keputusan sulit untuk

diinterpretasi[62]. Kemudian untuk algoritma XGBoost, kelebihannya adalah memiliki model formal yang lebih terstruktur untuk mengatur *overfitting* data, sehingga memberikan performa yang lebih optimal, dan untuk kekurangannya adalah rentan terhadap *overfitting* terutama jika tidak diatur dengan benar[63].

3.4 Teknik Pengumpulan Data

3.4.1 Data Collection

Penelitian ini menggunakan data sekunder, dengan cara mengakses data melalui *website* (https://satudata.jakarta.go.id/). Dataset ini memuat informasi tentang Indeks Standar Pencemar Udara (ISPU) Rentang waktu dataset penelitian ini diambil dari tahun 2013 bulan Januari sampai dengan tahun 2023 bulan Desember. Pada gambar 3.2 merupakan data mentah dengan format .csv

periode_data	tanggal	pm10	so2	co	o3	no2	max	critical	categori	lokasi_spku
202001	01/01/20	38	36	25	46	9	46	O3	BAIK	DKI5
202001	02/01/20	45	36	39	102	8	102	O3	TIDAK SEHAT	DKI5
202001	03/01/20	51	37	27	63	10	63	O3	SEDANG	DKI5
202001	04/01/20	51	38	19	85	10	85	O3	SEDANG	DKI5
202001	05/01/20	52	39	25	62	9	62	O3	SEDANG	DKI5
202001	06/01/20	62	37	39	64	9	64	O3	SEDANG	DKI5
202001	07/01/20	50	38	39	66	11	66	O3	SEDANG	DKI5
202001	08/01/20	52	40	22	70	11	70	O3	SEDANG	DKI5
202001	09/01/20	82	38	58	71	13	82	PM10	SEDANG	DKI4
202001	10/01/20	44	37	27	47	9	47	O3	BAIK	DKI5
202001	11/01/20	44	38	25	80	9	80	O3	SEDANG	DKI5
202001	12/01/20	43	38	17	46	5	46	O3	BAIK	DKI5
202001	13/01/20	42	38	15	51	8	51	O3	SEDANG	DKI4
202001	14/01/20	40	32	17	48	9	48	O3	BAIK	DKI4
202001	15/01/20	44	29	16	63	9	63	O3	SEDANG	DKI3
202001	16/01/20	70	60	41	78	12	78	O3	SEDANG	DKI3
202001	17/01/20	86	34	40	125	20	125	O3	TIDAK SEHAT	DKI4
202001	18/01/20	59	30	25	75	7	75	O3	SEDANG	DKI4
202001	19/01/20	76	34	71	110	15	110	O3	TIDAK SEHAT	DKI3
202001	20/01/20	56	31	20	65	9	65	O3	SEDANG	DKI3
202001	21/01/20	62	36	41	50	15	62	PM10	SEDANG	DKI3
202001	22/01/20	82	36	43	64	15	82	PM10	SEDANG	DKI4
202001	23/01/20	60	32	29	59	18	60	PM10	SEDANG	DKI4

Gambar 3. 2 Data Mentah ISPU

3.5 Teknik Analisis Data

Penelitian ini menggunakan metode data mining bersama dengan pendekatan klasifikasi. Pada Tabel 3.3 merupakan perbandingan antara *framework* KDD, CRISP-DM,dan SEMMA.

Tabel 3. 3 Perbandingan antara KDD, CRISP-DM,dan SEMMA

Data Mining Framework	**KDD**	**CRISP-DM**	**SEMMA**
No of Steps	5	6	5
Name of Steps	-	*Business Understanding*	-

	Selection	*Data Understanding*	*Sample*
	Pre-processing		*Explore*
	Transformation	*Data Preparation*	*Modify*
	Data Mining	*Modeling*	*Model*
	Interpretation /Evaluation	*Evaluation*	*Assessment*
	-	*Deployment*	-

Berdasarkan tabel 3.3 yang merupakan tabel perbandingan ketiga *framework* yaitu KDD, CRISP-DM, dan SEMMA. Dari ketiga *framework*, penelitian akan menggunakan *framework* KDD karena KDD menyajikan suatu proses terstruktur untuk mengenali pola yang valid, inovatif, bermanfaat, dan dapat dimengerti dari kumpulan data yang besar [64]. Penggunaan KDD didasarkan pada penelitian [15] KDD dipilih untuk menyajikan kerangka kerja yang terstruktur untuk memperoleh pengetahuan baru dari data kualitas udara.

Berdasarkan perbandingan *framework* pada tabel 3.2, penerapan *framework* KDD dalam penelitian ini dilakukan dengan menggunakan bahasa pemrograman *Python*. Pada Tabel 3.4, terdapat perbandingan bahasa pemrograman *Python* dan *R* [65]:

Tabel 3. 4 Perbandingan Python dan R

Performance	**Python**	**R**
Kelebihan	-. Memiliki berbagai modul dan standar *library* yang mencakup berbagai fungsi. -. Memiliki bahasa pemrograman yang dinamis,dan cocok untuk diterapkan pada *machine learning* dan *deep learning.* -. Memeiliki kinerja yang baik dalam mengklasifikasi data dan dapat dijalankan di berbagai *platform* atau bersifat *Open Source* dan sintaks yang mudah dipahami.	-. Dirancang khusus untuk analisis statistik dan visualisasi data. Sehingga memiliki visualisasi yang kuat. -. Bersifat *Open Source* yang artinya dapat diunduh dan digunakan secara gratis. -. Dapat berinteraksi dengan bahasa pemrograman lain seperti *Python* dan SQL.

Kekurangan	-. Ukuran file eksekusi aplikasi yang besar. -. Sintaks *Python* sangat bergantung pada indentasi.	-. Kurangnya performa untuk pengolahan *Big Data* -.. Sintaks R terlalu sulit dibaca.

Berdasarkan Tabel 3.4 yang merupakan perbandingan *Python* dan R, Penelitian ini menggunakan bahasa pemrograman Pyhton karena lebih cocok untuk teknik *machine learning* dan memiliki sintaks yang mudah dipahami sehingga cocok untuk pengolahan *Big Data*, dan memiliki performa yang baik dalam mengklasifikasikan data [66] .

Teknik analisis data dilakukan menggunakan *tools Jupyter Notebook* dan bahasa pemrograman *Python*. Tabel 3.5 merupakan perbandingan antara *Jupyter Notebook* dan RStudio sebagai *tools* analisis data.

Tabel 3. 5 Perbandingan Jupyter Notebook dan R Studio

Performance	**Jupyter Notebook**	**RStudio**
Bahasa Pemrograman	-.Pyhton	-.R
Kelebihan	-. Dapat dimanfaatkan untuk mendukung berbagai *environment.* -. Mempunyai fitur *Save* and *Checkpoint* yang berperan dalam pembuatan *checkpoint*. -. *JupyterHub* mampu beroperasi dengan jumlah pengguna yang mencapai puluhan ribu. -. *Open Source*s dan didesain untuk dijalankan di berbagai infrastruktur.	-. Mudah terhubung dengan berbagai jenis basis data. -. Tidak melakukan operasi parameter *machine learning* secara otomatis.
Kekurangan	-. Sulit untuk melakukan *debug*.	-. Tidak secara otomatis menjalankan operasi parameter *machine learning*. Memerlukan pengetahuan lebih dalam pemrograman atau *coding.*

Berdasarkan Tabel 3.5, Penelitian ini, peneliti menggunakan *Jupyter Notebook* sebagai *tools* karena *platform* ini menyediakan berbagai teknik dalam *data mining*, termasuk teknik *supervised* dan *unsupervised learning*. *Jupyter Notebook* juga menyajikan beragam *library* yang mendukung pembuatan sistem, *library* ini merupakan kumpulan modul dengan kode yang dapat digunakan secara berulang dalam berbagai program. Pemilihan *Jupyter Notebook* didasarkan pada kelebihannya dibandingkan dengan RStudio, terutama dalam kemampuannya beroperasi dalam berbagai *environment* [67].

Dalam ekosistem Python, *Jupyter Notebook* menyediakan berbagai *library* seperti *numPy, Pandas, matplotlib, dan sklearn*. Selain itu, platform ini juga menyajikan berbagai algoritma dan menghasilkan *output* visualisasi grafis [68]. Kehadiran *library-library* tersebut sangat bermanfaat di bidang data *science* dan *machine learning*, khususnya dalam konteks penelitian ini yang menggunakan metode klasterisasi yang akan diimplementasikan menggunakan bahasa pemrograman *Python*. Kelebihan *Jupyter Notebook* juga terletak pada integrasinya dengan *Python*, dan library yang disediakan tidak memiliki batasan waktu tertentu. Sedangakan RStudio idak secara otomatis menjalankan operasi parameter *machine learning* dan memerlukan pengetahuan lebih dalam pemrograman atau *coding*. Oleh karena itu, pilihan untuk menggunakan *Jupyter Notebook* dianggap lebih tepat untuk memastikan kelancaran dan keandalan dalam pelaksanaan penelitian.