BAB III
METODOLOGI PENELITIAN

3.1 Subjek dan Objek Penelitian

subjek penelitian adalah entitas yang diamati, baik itu orang, tempat, atau benda. Dalam penelitian ini, subjeknya adalah beberapa judul *Game* yang tersedia di platform penyedia *Game*, yang telah dikumpulkan dalam satu dataset. Objek penelitian adalah atribut yang berasal dari orang atau kegiatan yang ditetapkan oleh peneliti. Pada penelitian ini, objeknya adalah data rating *Game* yang terdiri dari lima label: 3+, 7+, 12+, 16+, dan 18+.

3.2 Alat dan Bahan Penelitian

3.2.1 Alat Penelitian

Penelitian ini dilakukan menggunakan alat berupa *hardware* atau perangkat keras dan *software* atau perangkat lunak sebagai alat yang mendukung proses pengerjaan penelitian.

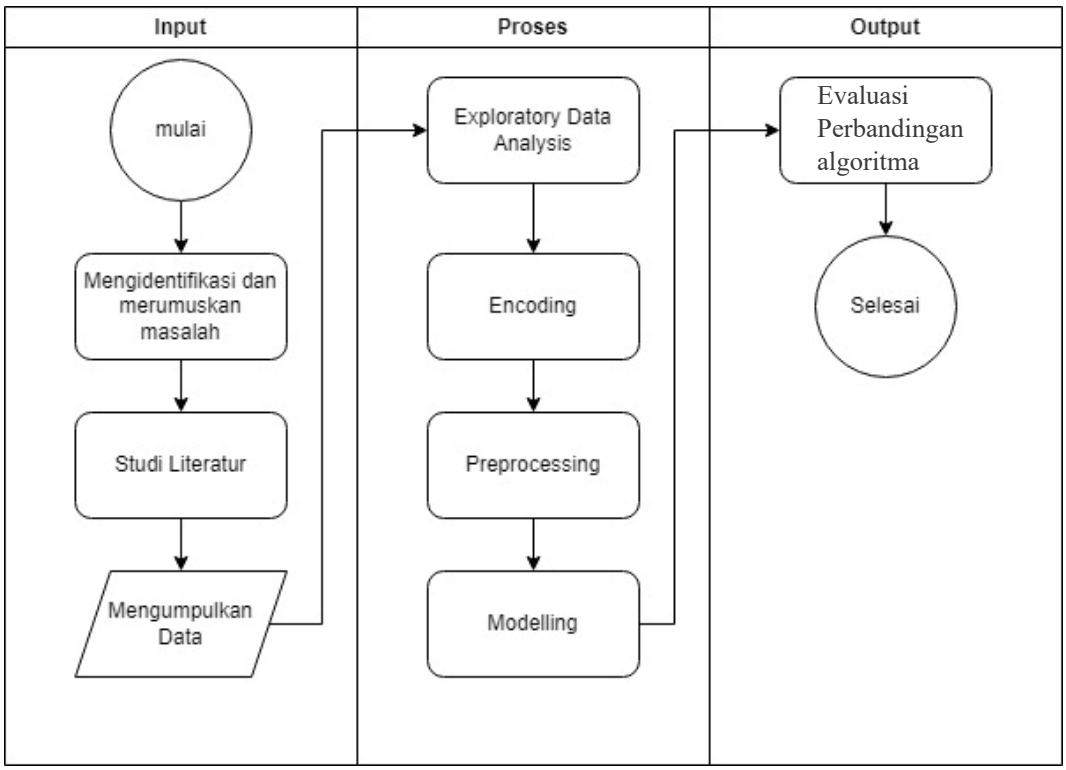
1. Perangkat Keras (*Hardware)*
    a. Laptop dengan spesifikasi Processor AMD Ryzen 3 3250U (2C / 4T, 2.6 / 3.5GHz, 1MB L2 / 4MB L3), RAM 8GB, dan storage 256GB SSD M.2 2242 PCIe 3.0×2
2. Perangkat Lunak *(Software)*
    a. OS *Windows* 11
    b. *Microsoft Office* 2019
    c. Bahasa pemrograman *Python*
    d. *Google Colab* untuk pemrosesan data
    e. *Microsoft Excel*

3.2.2 Bahan Penelitian

Bahan yang diambil sebagai dasar penelitian adalah data primer berupa judul *Game*, konten, dan label yang akan dihimpun dari hasil observasi peneliti pada

setiap *Game*. Penggunaan dataset ini menjadi landasan utama dalam menjalankan penelitian, memungkinkan untuk melakukan analisis yang lebih mendalam pada kumpulan data tersebut.

3.3 Diagram Alir Penelitian



Gambar 3. 1 Diagram Alir Penelitian

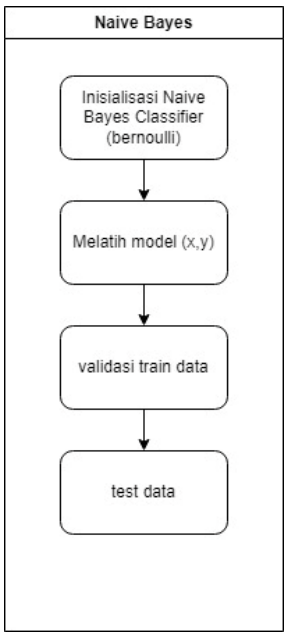
Gambar 3.1 menggambarkan diagram alir penelitian atau flowchart secara umum, yang mencakup semua langkah yang dilakukan dalam penelitian ini, mulai dari pengumpulan data hingga interpretasi hasil. Flowchart ini memandu setiap tahapan penelitian secara sistematis dan logis, memastikan bahwa semua langkah penting diikuti secara berurutan untuk mencapai hasil yang valid dan dapat diandalkan. Pada tahap pengumpulan data, berbagai sumber data diidentifikasi dan dikumpulkan dengan cermat untuk memastikan kelengkapan dan akurasi data yang akan digunakan. Setelah data dikumpulkan, tahap preprocessing dilakukan untuk membersihkan dan mempersiapkan data.

Selanjutnya, pada tahap *modelling*, dua algoritma yang berbeda *yaitu Naive Bayes* dan *K-Nearest Neighbors (KNN)* digunakan untuk membangun model

prediktif. *Naive Bay*es digunakan karena kemampuannya dalam menangani data berukuran besar dengan cepat dan efisien, serta sifat probabilistiknya yang sederhana namun kuat. Sementara itu, *KNN* dipilih karena kemampuannya dalam menghasilkan prediksi yang akurat dengan memanfaatkan kedekatan data berdasarkan fitur-fiturnya. Setelah model dibangun menggunakan kedua algoritma tersebut, tahap evaluasi dilakukan untuk mengukur kinerja masing-masing model menggunakan metrik evaluasi yang relevan, seperti akurasi, presisi, *recall*, dan *F1-score*.

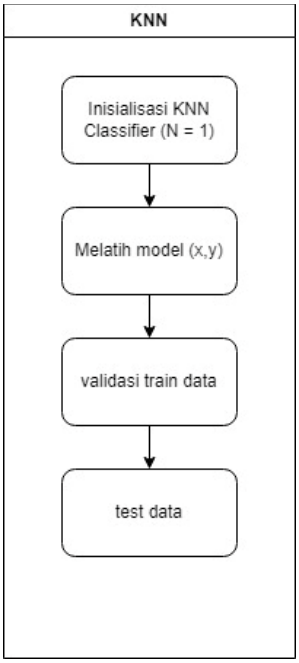
Tahap interpretasi hasil kemudian dilakukan untuk menganalisis dan memahami hasil yang diperoleh dari model-model tersebut. Analisis ini mencakup perbandingan kinerja kedua algoritma, interpretasi hasil prediksi, serta identifikasi faktor-faktor yang mempengaruhi kinerja model. Dengan mengikuti alur yang jelas dan terstruktur dalam flowchart ini, penelitian diharapkan dapat mencapai hasil yang valid, andal, dan bermanfaat dalam konteks klasifikasi *rating Game* berdasarkan algoritma *Naive Bayes* dan *KNN* sesuai dengan standar *International Age Rating Coalition (IARC)*.

3.3.1 Flowchart Modelling



Gambar 3. 2 *Flowchart Model Naïve Bayes*

Gambar 3.2 menampilkan *flowchart* dari proses model *Naive Bayes* dengan rincian langkah-langkah proses permodelan. Pada tahap awal, model *Naive Bayes* diinisialisasi menggunakan distribusi *Bernoulli*. Distribusi *Bernoulli* adalah salah satu varian dari algoritma *Naive Bayes* yang cocok untuk data biner. Setelah inisialisasi, model dilatih menggunakan dataset yang telah disiapkan. Pada langkah ini, dataset dibagi menjadi fitur (x) dan label (y). Model *Naive Bayes* mempelajari hubungan antara fitur dan label untuk membangun model probabilistik yang akan digunakan untuk prediksi. Setelah model dilatih, langkah berikutnya adalah melakukan validasi terhadap data pelatihan. Validasi ini bertujuan untuk mengukur kinerja model pada data yang sudah diketahui untuk memastikan bahwa model tidak *overfitting*. Langkah terakhir adalah menguji model pada data pengujian yang belum pernah dilihat oleh model sebelumnya. Tujuan dari pengujian ini adalah untuk mengevaluasi kemampuan model dalam memprediksi label dari data baru. Kinerja model pada data pengujian memberikan gambaran tentang seberapa baik model dapat menggeneralisasi dan memberikan prediksi yang akurat di luar dataset pelatihan.



Gambar 3. 3 *Flowchart Model K-Nearest Neighbors*

Gambar 3.3 menampilkan *flowchart* dari proses model *KNN* dengan rincian langkah-langkah proses permodelan. Pada tahap awal, model *KNN* diinisialisasi dengan parameter N = 1, yang berarti bahwa hanya satu tetangga terdekat yang akan digunakan untuk menentukan kelas dari data yang tidak diketahui. Setelah inisialisasi, model dilatih menggunakan dataset yang telah disiapkan. Dataset dibagi menjadi fitur (x) dan label (y). Setelah model siap, langkah berikutnya adalah melakukan validasi terhadap data pelatihan. Validasi ini bertujuan untuk mengukur kinerja model pada data yang sudah diketahui. Langkah terakhir adalah menguji model pada data pengujian yang belum pernah dilihat oleh model sebelumnya. Tujuan dari pengujian ini adalah untuk mengevaluasi kemampuan model dalam memprediksi label dari data baru. Kinerja model pada data pengujian memberikan gambaran tentang seberapa baik model dapat menggeneralisasi dan memberikan prediksi yang akurat di luar dataset pelatihan. Model *KNN* melakukan prediksi berdasarkan mayoritas kelas dari N tetangga terdekat dari titik data yang tidak diketahui.

3.3.2 Mengidentifikasi dan Merumuskan Masalah

Pada tahap ini, peneliti menentukan area, topik, dan masalah penelitian yang akan diteliti. Selain itu, peneliti juga mengusulkan metode yang akan diterapkan dalam pelaksanaan penelitian. Pada fase ini, peneliti mengkaji permasalahan-permasalahan yang masih terjadi dalam kehidupan sehari-hari. Tujuan penelitian dijelaskan dengan rinci, termasuk lingkup yang akan dikaji.

3.3.3 Studi Literatur

Langkah berikutnya adalah menjalankan tinjauan pustaka yang melibatkan pengumpulan informasi terkait dengan isu penelitian yang sedang dianalisis. Data yang akan dikumpulkan mencakup informasi tentang kategori *rating*, konten, judul *Game*, dan algoritma yang diusulkan, yaitu *Naïve Bayes* dan *KNN*. Informasi ini diperoleh dari sumber-sumber seperti jurnal ilmiah, buku, situs web, serta berbagai media elektronik lainnya. Melalui tinjauan pustaka, tujuan yang ingin dicapai adalah untuk memperdalam pemahaman tentang isu yang dibahas dalam penelitian

ini, sekaligus menyediakan dasar yang kokoh untuk pengembangan penelitian lebih lanjut.

3.3.4 Pengumpulan Data

Pengumpulan data pada bidang klasifikasi yang menggunakan dataset rating usia *Game*s dari Google Play Store melibatkan beberapa langkah penting untuk mengumpulkan, membersihkan, dan menyiapkan data yang akan digunakan untuk membangun dan mengevaluasi model klasifikasi. Pertama, harus menentukan tujuan dan ruang lingkup proyek, misalnya untuk mengembangkan model klasifikasi yang dapat memprediksi rating usia untuk *Game* berdasarkan berbagai fitur yang tersedia. Selanjutnya, mengidentifikasi sumber data, yaitu Google Play Store, dan memutuskan cara pengumpulan data, baik melalui teknik web scraping atau API (jika tersedia). Informasi yang relevan dari Google Play Store mencakup nama *Game*, konten, rating, dan lainnya yang dirujuk pada International Age Rating Coalition (IARC).

3.3.5 *Exploratory Data Analysis*

Exploratory Data Analysis (EDA) adalah pendekatan analisis data yang berfokus pada pemeriksaan dan peringkasan karakteristik utama dataset, sering kali menggunakan metode visualisasi. *EDA* adalah langkah awal yang sangat penting dalam analisis data karena membantu para analis dan ilmuwan data memahami struktur, pola, anomali, dan hubungan dalam data sebelum menerapkan model statistik atau machine learning. Tujuan utama *EDA* adalah untuk memahami struktur data, mendeteksi anomali, menemukan pola dan hubungan antara variabel, serta mengidentifikasi kebutuhan pemrosesan data lebih lanjut.

3.3.6 *Encoding*

Encoding pada klasifikasi adalah proses mengubah data kategori atau teks menjadi format numerik yang dapat dipahami oleh algoritma *machine learning*. Banyak model *machine learning* hanya dapat menangani data numerik, sehingga data kategori rating *Game*s harus diubah menjadi angka. Terdapat beberapa teknik *encoding* yang umum digunakan, termasuk *one-hot encoding* dan *label encoding*. *One-hot encoding* mengubah setiap kategori menjadi vektor biner yang panjangnya

sama dengan jumlah kategori, di mana hanya satu elemen yang bernilai 1 dan sisanya 0, untuk mewakili setiap kategori unik. *Label encoding*, memberikan angka unik untuk setiap kategori. Teknik *encoding* ini membantu algoritma machine learning untuk memproses dan menganalisis data kategori dengan lebih efektif, memungkinkan model untuk mengenali pola dan hubungan dalam data yang lebih kompleks. Dengan demikian, *encoding* adalah langkah penting dalam pra-pemrosesan data untuk klasifikasi, memastikan bahwa data yang dimasukkan ke dalam model berada dalam format yang dapat diolah dengan benar.

Pada proses klasifikasi ini, terdapat lima label klasifikasi yang berbeda, yaitu 3+, 7+, 12+, 16+, dan 18+. Oleh karena itu, untuk mengubah label tersebut ke dalam format numerik yang dapat digunakan dalam model *machine learning*, digunakan metode *label encoding*. Dengan metode ini, masing-masing label tersebut akan diberi nilai numerik yang unik, yaitu 0 untuk 12+, 1 untuk 16+, 2 untuk 18+, 3 untuk 3+, dan 4 untuk 7+. Hal ini memungkinkan model *machine learning* untuk memahami dan memproses label-label tersebut dengan lebih efektif.

3.3.7 *Preprocessing*

Preprocessing dalam konteks klasifikasi rating usia *Game*s adalah serangkaian langkah penting yang dilakukan untuk membersihkan dan mengubah data agar siap digunakan dalam proses klasifikasi. Langkah-langkah ini mencakup pembersihan data dengan menghapus kolom data yang tidak relevan misalnya pada judul *game* dan *rating*, Judul *game* dihapus karena nama *game* biasanya berupa teks dan tidak mengandung informasi numerik atau kategorikal yang dapat digunakan secara langsung dalam *model machine learning*. Nama tidak memiliki hubungan langsung dengan prediksi *rating game* dan hanya akan memperkenalkan *noise* ke dalam model. *Rating* dihapus karena *rating* adalah variabel target yang ingin kita prediksi. Dalam *supervised learning*, variabel target harus dipisahkan dari data fitur.

pengkodean label untuk mengubah label rating usia menjadi format yang sesuai dengan algoritma klasifikasi, pemisahan data menjadi data pelatihan dan data pengujian misalnya menggunakan pemisahan data rasio seperti 80% data *training* 20% data *testing* dan sebagainya, normalisasi nilai fitur-fitur ke dalam skala yang

seragam menggunakan *StandardScaler* dari *library scikit-learn*, balancing data untuk memastikan distribusi kelas yang seimbang, dan transformasi data jika diperlukan. Proses preprocessing ini penting untuk memastikan data yang digunakan dalam klasifikasi rating usia *Game*s bersih, relevan, dan siap digunakan oleh model klasifikasi.

3.3.8 *Modelling*

Langkah selanjutnya melibatkan pembangunan model klasifikasi menggunakan metode *Naïve Bayes* dan *KNN*. Tahap ini menjadi inti dari ekstraksi pengetahuan dari data yang telah dikumpulkan, dan akan dijalankan melalui implementasi menggunakan bahasa pemrograman *Python*. Metode ini memastikan bahwa model memiliki sejumlah besar data untuk dipelajari selama tahap *training* sehingga dapat mengenali pola dan karakteristik yang mendasari klasifikasi kategori *rating* dengan lebih baik.

Naïve Bayes digunakan utamanya dalam memprediksi probabilitas kelas atau label tertentu berdasarkan fitur-fitur yang diberikan. Implementasinya melibatkan perhitungan probabilitas posterior dari kelas atau label berdasarkan data latih, dan memilih kelas atau label dengan probabilitas tertinggi untuk prediksi pada data uji. Proses ini memungkinkan evaluasi kemampuan model klasifikasi untuk mengategorikan data dengan akurasi tinggi.

Sementara itu, algoritma *K-Nearest Neighbors (KNN*) adalah algoritma klasifikasi yang sederhana dan intuitif. Pendekatan *KNN* mencari kelas terdekat dari data uji berdasarkan jarak dalam ruang fitur. Dengan kata lain, jika sebuah *instance* memiliki tetangga-tetangga dari kelas tertentu, maka *instance* tersebut kemungkinan besar akan termasuk dalam kelas yang sama. *KNN* tidak memiliki proses pelatihan yang sebenarnya; modelnya hanya "mengingat" data latihnya untuk kemudian memprediksi kelas data uji berdasarkan tetangga terdekatnya.

3.3.9 Evaluasi dan Analisis Performa

Pada langkah ini, menggunakan metode *confusion matrix* untuk mengevaluasi kinerja model klasifikasi *Naïve Bayes* dan *KNN* dalam

mengkategorikan *rating Game*. Evaluasi ini bertujuan untuk mengukur akurasi, presisi, *recall*, dan skor F1 dalam menilai kinerja model. Analisis hasil evaluasi ini akan membantu dalam membuat kesimpulan yang lebih solid, memberikan pemahaman mendalam mengenai efektivitas dan keandalan model klasifikasi yang diterapkan. Kesimpulan ini tidak hanya memberikan wawasan tentang performa model saat ini, tetapi juga dapat menjadi dasar untuk rekomendasi atau pengembangan lebih lanjut di masa depan, terutama dalam pemilihan algoritma klasifikasi yang tepat untuk kasus serupa. Oleh karena itu, evaluasi ini diharapkan memberikan wawasan yang lebih baik terkait kinerja model klasifikasi *Naïve Bayes* dan *KNN* dalam mengklasifikasikan data, berpotensi menjadi pijakan untuk saran atau proyek pengembangan di bidang ini.