BAB III

METODOLOGI PENELITIAN

3.1 Metode Penelitian

Algoritma C5.0 merupakan algoritma berbasis decision tree yang dapat menangani atribut kontinyu dan diskrit. Algoritma C5.0 juga merupakan algoritma penyempurnaan dari algoritma decision tree lainnya seperti algoritma C4.5 dan juga algoritma ID3. Algoritma c5.0 menentukan parent bagi node selanjutnya dengan cara memilih atribut dengan nilai information gain yang tertinggi. Proses ini akan berlanjut sampai bagian dari sampel tidak dapat dibagi. Berikut merupakan persamaan untuk menghitung entropy atribut:

$$I(S_1, S_2, \dots, \dots, S_m) = -\sum_{i=1}^{m} pi * log_2(pi) \quad (1)$$

Keterangan:

S : Himpunan Kasus

m : Jumlah Sampel

pi : Proporsi Kelas

Sedangkan untuk mendapatkan informasi nilai subset tersebut dapat dilihat pada persamaan berikut:

$$E(A) = \sum_{j=i}^{y} \frac{s_{ij}+\dots+s_{mj}}{s} I(S_{1j}, \dots, S_{mj}) \quad (2)$$

Keterangan:

$\frac{s_{ij}+\dots+s_{mj}}{s}$ = Jumlah subset J yang dibagi dengan jumlah sampel S.

Maka untuk mendapatkan nilai gain dapat dihitung dengan menggunakan persamaan sebagai berikut:

$$Gain(A) = I(S_1, S_2, \dots, S_m) - E(A) \quad (3)$$

Keterangan:

A : Atribut

S : Himpunan Kasus

S1 : Jumlah Sampel

Proses pemotongan atau proses menghilangkan cabang-cabang yang tidak diperlukan dalam *tree* atau disebut juga *pruning* (pemangkasan) menggunakan sebuah rumus untuk menghitung nilai *error pruning*, rumushnya adalah sebagai berikut:

$$e = \frac{r + \frac{z^2}{2n} + z\sqrt{\frac{r}{n} - \frac{r^2}{n} + \frac{z^2}{4n^2}}}{1 + \frac{z^2}{n}} \quad (4)$$

Keterangan:

r : nilai perbandingan *error rate*

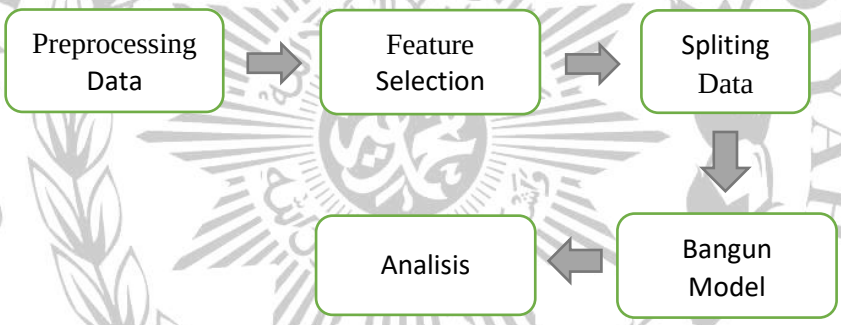
n : nilai total sample

z : Φ-1(c)

c : confidence level

3.2 Alur Penelitian

Pada penelitian ini terdapat beberapa tahapan, tahapannya adalah sebagai berikut:



3.2.1 Preprocessing Data

Tahapan pertama pada penelitian ini adalah Preprocessing data, dimana pada data yang sudah ada akan dilakukan *Data Cleansing* atau pembersihan data. Pada tahap ini data akan diperiksa dan dianalisis kualitas dari data, setelah diperiksa dan dianalisis, data tersebut akan dibersihkan jika terdapat anomaly atau kecacatan pada data tersebut. Kecacatan pada data tersebut bisa berupa data yang salah, data yang rusak, tidak akurat, tidak lengkap, dan salah format. Hal ini dilakukan agar meningkatkan efisiensi kerja dan menurunkan tingkat error yang dapat menyebabkan kesalahan dalam menganalisis data.

3.2.2 *Feature selection*

Pada tahap ini akan dilakukan pemilihan variabel dari dataset yang ada untuk digunakan ketika melakukan pembangunan model. Pada penelitian ini ada dua skenario yang menggunakan Feature Selecton yang berbeda, pada skenario pertama *Feature selection* yang digunakan adalah *Chi-Square Test* dan pada skenario kedua *Feature selection* yang digunakan adalah Pearson Correlation. Metode *Chi-Square Test* merupakan sebuah metode uji statistik yang digunakan untuk membandingkan hasil yang diamati dan yang diharapkan. Tujuannya adalah untuk mengidentifikasi adanya kesenjangan diantara data yang sebenarnya dan data prediksi yang disebabkan oleh keterkaitan antar variabel yang dipertimbangkan. Metode *Chi-Square Test* lebih efektif untuk digunakan pada data bersifat kategorikal karena mudah diimplementasikan dan cepat dihitung. Namun kelemahaannya adalah jika ada data yang bersifat kontinu, maka data tersebut harus diubah terlebih dahulu menjadi kategorial, dan metode *Chi-Square Test* hanya melakukan evaluasi hubungan antara satu fitur dan target dengan mengabaikan korelasi antar setiap fitur. Sedangkan Pearson Correlation adalah metode yang memanfaatkan koefisien korelasi pearson untuk memilih fitur yang paling relevan terhadap variabel target dari sebuah dataset yang telah ditentukan. Nilai koefisien korelasi pearson berkisar antara -1 hingga 1 dimana -1 menunjukkan korelasi negatif sempurna, 0 menunjukan tidak ada korelasi linier, dan 1 menunjukkan korelasi positif sempurna. Pearson Correlation lebih sederhana dan cepat untuk dihitung serta membantu untuk mengurangi dimensi data yang dapat mempengaruhi kinerja model dan mengurangi overfitting.

3.2.3 Spliting data

Pada tahapan ini, dataset yang sudah dibersihkan dan dipilih variabel yang akan digunakan akan dibagi menjadi 2 bagian yaitu data training dan data testing. Data training digunakan untuk melatih model yang akan dibangun, dan data testing nantinya akan digunakan setelah latihan selesai untuk menguji model yang sudah dibangun.

3.2.4 Bangun Model

Pada tahap ini, model klasifikasi akan dibangun menggunakan bahasa pemrograman python pada website google collab untuk membentuk pohon keputusan.

3.2.5 Analisis

Pada tahapan ini akan dilakukan proses analisis menggunakan K-Fold Cross Validation. K-Fold Cross Validation adalah sebuah teknik analisis yang digunakan dengan cara membagi data menjadi beberapa subset atau "folds" untuk melakukan evaluasi terhadap suatu model *Machine Learning*.

Dalam proses evaluasi model, digunakan beberapa metrik untuk mengukur kinerja model klasifikasi, yaitu akurasi, presisi, recall, dan F1-Score. Berikut adalah penjelasan dari masing-masing metrik tersebut:

a. Akurasi (*accuracy*)

Akurasi adalah perbandingan antara jumlah prediksi yang benar dengan total keseluruhan data yang diuji. Metrik ini mengukur seberapa baik model dalam mengklasifikasikan data dengan benar. Namun, akurasi dapat menjadi kurang representatif jika terdapat ketidakseimbangan jumlah data antar kelas.

$$Akurasi = \frac{Jumlah\ Prediksi\ Benar}{Total\ Data} \quad (5)$$

b. Presisi (*Precision*)

Presisi mengukur seberapa banyak prediksi positif yang benar dibandingkan dengan seluruh prediksi positif yang dibuat oleh model. Metrik ini sangat penting ketika kesalahan dalam mengklasifikasikan data sebagai positif dapat memberikan dampak signifikan. Nilai presisi yang tinggi menunjukan bahwa model jarang salah dalam memberikan prediksi positif

$$Presisi = \frac{True\ Positive\ (TP)}{True\ Positive\ (TP) + False\ Positive\ (FP)} \quad (6)$$

Keterangan :

True positive (TP) : jika model memprediksi jawabannya iya, dan hasilnya iya

True Negative (TN) : jika model memprediksi jawabannya tidak, dan hasilnya tidak

False Positive (FP) : jika model memprediksi iya, namun hasilnya tidak

False Negative (FN) : jika model memprediksi tidak, namun hasilnya iya

c. Recall

Recall mengukur seberapa banyak data positif yang benar-benar terdeteksi oleh model dibandingkan dengan seluruh data positif yang sebenarnya ada. Metrik ini sangat penting dalam kasus dimana kesalahan dalam melewatkan data positif harus diminimalkan. Nilai recall yang tinggi menunjukkan bahwa model dapat menangkap hampir semua data positif yang ada

$$Recall = \frac{True\ Positive\ (TP)}{True\ Positive\ (TP) + False\ Negative\ (FN)} \quad (7)$$

Keterangan :

True positive (TP) : jika model memprediksi jawabannya iya, dan hasilnya iya

True Negative (TN) : jika model memprediksi jawabannya tidak, dan hasilnya tidak

False Positive (FP) : jika model memprediksi iya, namun hasilnya tidak

False Negative (FN) : jika model memprediksi tidak, namun hasilnya iya

d. F1-Score

F1-Score adalah harmonik rata-rata antara presisi dan recall. Metrik ini digunakan untuk mencari keseimbangan antara presisi dan recall, terutama ketika ada ketidakseimbangan jumlah data antar kelas. Nilai *F1-Score* yang tinggi menunjukkan bahwa model memiliki keseimbangan yang baik antara presisi dan recall.

$$F1 - Score = 2\ x\ \frac{Presisi\ x\ Recall}{Presisi + Recall} \quad (8)$$