BAB II

TINJAUAN PUSTAKA DAN DASAR TEORI

2.1 Tinjauan Pustaka

Penelitian ini dilakukan tidak terlepas dari hasil penelitian-penelitian terdahulu yang pernah dilakukan sebagai bahan perbandingan dan kajian. Diantaranya adalah Penelitian tentang penerapan data mining untuk klasifikasi penyakit *hepatocellular carcinoma* menggunakan *Naïve Bayes* (Doni et al., 2021). Penelitian ini bertujuan untuk mengklasifikasikan tingkat kemungkinan hidup pasien yang telah di diagnosis menderita penyakit *Hepatocellular Carcinoma* dengan menggunakan penerapan metode data mining serta melakukan pengukuran terhadap performa algoritma *Naïve Bayes* dengan mengacu kepada *Confusion Matrix* dan *Kurva ROC.*

Penelitian kedua tentang klasifikasi penyakit diabetes melitus menggunakan algoritma *Naïve Bayes Classifier* (Khasanah et al., 2022). Tujuan dari penelitian ini adalah untuk mengetahui hasil klasifikasi pasin ke dalam dua kategori diagnosis diabetes melitus yaitu ‘Ya’ dan ‘Tidak’ menggunakan algoritma *Naïve Bayes Classifier* dan mengetahui tingkat akurasi dari empat proporsi data yaitu 60:40, 70:30, 80:20 dan 90:10.

Penelitian ketiga tentang model klasifikasi potensi penyakit diabetes melitus menggunakan metode *K-Nearest Neighbor* (Murtono, 2022). Tujuan dari

penelitian ini adalah untuk mengetahui model dan hasil klasifikasi potensi penyakit diabetes melitus berdasarkan hasil yang didapatkan. Hasil yang diperoleh dari klasifikasi pada data ini adalah 98,08%, *precission* sebesar 96,30%, dan *recall* sebesar 100.00%.

Penelitian keempat tentang sistem prediksi penyakit jantung koroner menggunakan metode *Naïve Bayes* (Larassati et al., 2022). Penelitian ini memiliki tujuan untuk membuat prediksi yang akan membantu para dokter untuk melakukan diagnose dengan tepat dan akurat sehingga penyakit jantung koroner dapat ditangani lebih awal. Salah satu algoritma klasifikasi data mining yang digunakan pada penelitian ini adalah algoritma *Naïve Bayes Classifier*. Algoritma ini diterapkan dengan tujuan untuk menghitung probabilitas kemungkinan seseorang pasien berdasarkan data rekam medis pasien.

Penelitian kelima tentang penerapan metode klasifikasi *K-Nearst Neigbor* pada dataset penderita penyakit diabetes (Argina, 2020). Metode yang digunakan yaitu algoritma *K-Nearst Neigbor* (KNN) yang dimana merupakan sebuah metode untuk melakukan klasifikasi terhadap objek berdasarkan data pembelajaran yang jaraknya paling dekat dengan objek tersebut. Pada hasil akhir penelitian ini, telah dihitung akurasi tertinggi 39% pada K=3, *presisi* tertinggi 65% pada K=3 dan K=5, *recall* tertinggi 36% pada K=3, dan *F-Measure* tertinggi 46% pada K=3.

Penelitian keenam tentang Sistem Prediksi Kelulusan Mahasiswa Berbasis *Web* Menggunakan Metode Algoritma *Naïve Bayes* (Atmojo, 2021). Sistem ini dibangun dengan bahasa pemrograman *php* dan database *mySQL*. Data yang

digunakan adalah data kelulusan mahasiswa program studi Informatika sebagai data *training* dan data *testing.* Hasil penelitian menunjukkan akurasi sebesar 93.75%. Dengan kata lain penggunaan metode *Naïve Bayes* untuk prediksi kelulusan mahasiswa telah berhasil di implementasikan menggunakan *Web Programming.*

Perbandingan penelitian yang digunakan dalam penelitian ini dapat dilihat pada tabel perbandingan 2.1.

Tabel 2. 1 Perbandingan Penelitian

No	Penulis	Judul Penelitian	Metode	Hasil
1	Doni et all., 2021	Penerapan Data Mining Untuk Klasifikasi Penyakit Hepatocellular Carcinoma	*Naïve Bayes*	Menggunakan confusion matriks dengan akurasi 70,30%, presisi 73,53% dan recall 77,32%
2	Khasanah et all., 2022	Klasifikasi Penyakit Diabetes Melitus Menggunakan Algoritma Naïve Bayes Bayes Classifier	Naïve Bayes	Nilai akurasi terbaik adalah pada proporsi data testing 40% dan 20% dengan nilai akurasi sebesar 92,31%

Tabel Perbandingan Penelitian Lanjutan

No	Penulis	Judul Penelitian	Metode	Hasil
3	Murtono, 2022	Model Klasifikasi Potensi Penyakit Diabetes Melitus Menggunakan Metode K-Nearst Neighbor	K-Nearst Neighbor (KNN)	Didapatkan hasil klasifikasi dengan akurasi 98,08%, presisi 96,30%, dan recall 100.00%.
4	Larassati et all., 2022	Sistem Prediksi Penyakit Jantung Koroner	Naïve Bayes	Hasil yang diperoleh dari percobaan pertama memiliki akurasi paling tinggi yaitu 83.1%.
5	Argina, 2020	Penerapan Metode Klasifikasi *K-Nearst Neghbor* pada dataset Penderita Penyakit Diabetes	*K-Nearest Neighbor*	Didapatkan hasil yang tertinggi dengan akurasi 39% pada K=3, presisi 65% pada K=3 dan K=5, dan F-Measure 46% pada K=3

Tabel Perbandingan Penelitian Lanjutan

No	Penulis	Judul Penelitian	Metode	Hasil
6	Atmojo, 2021	Sistem Prediksi Kelulusan Mahasiswa Berbasis Web Menggunakan Metode Algoritma *Naïve Bayes*	*Naïve Bayes*	Penggunaan metode *Naïve Bayes* untuk prediksi kelulusan mahasiswa telah Berhasil Diimplementasikan menggunakan *Web Programming.*
7	Ainun Annisa, 2024	Penerapan Metode *Naïve Bayes* Untuk Klasifikasi Penyakit Diabetes	Naïve Bayes	Didapatkan hasil tertinggi pada Perbandingan 80:20 dengan akurasi 90.38%, presisi 45.84%, recall 65.73% dan F1-score 54.01%.

2.2 Dasar Teori

2.2.1 Diabetes Melitus

Diabetes Melitus merupakan penyakit gangguan metabolisme kronis yang ditandai peningkatan glukosa darah (*Hiperlikemi*), disebabkan karena ketidakseimbangan antara suplai dan kebutuhan untuk memfasilitasi masuknya glukosa dalam sel agar dapat digunakan untuk metabolisme dan pertumbuhan sel. Berkurang atau tidak adanya insulin menjadikan glukosa tertahan didalam darah dan menimbulkan peningkatan gula darah, sementara sel menjadi kekurangan glukosa yang sangat dibutuhkan dalam kelangsungan dan fungsi sel.

2.2.2 Naive Bayes Classifier

Naïve Bayes pertama kali diusulkan oleh Revered Thomas Bayes, antara tahun 1702 dan 1761 penggunaan *Naïve Bayes* mulai dikenalkan. *Naïve Bayes Classifier* adalah salah satu model probabilistik yang paling sederhana namun sangat efektif. Menurut Ng, dasar dari Naïve Bayes adalah Teorema Bayes yang digunakan untuk menghitung pobabilitas posterior dari suatu kelas berdasarkan fitur-fitur yang diamati. Penerapan algoritma *Naïve Bayes* dituliskan sebagai berikut:

1. Menghitung Probablitas Prior

Probabilitas Prior adalah probabilitas dari setiap kelas dalam data tanpa mempertimbangkan fitur-fitur yang ada. Rumus untuk menghitung probabilitas prior dari suatu kelas C adalah:

$$P(C) = \frac{Jumlah\ kejadian\ kelas\ C}{total\ jumlah\ kejadian}$$

2. Menghitung Likelihood

Likelihood digunakan untuk menghitung seberapa mungkin data (fitur-fitur) muncul jika diketahui kelasnya.

$$P(Fi|C) = \frac{Jumlah\ kejadian\ fitur\ fi\ dalam\ kelas\ C}{Jumlah\ total\ kejadian\ dalam\ kelas\ C}$$

3. Menghitung Probabilitas Posterior

Probabilitas Posterior adalah probabilitas dari hipotesis setelah mempertimbangkan bukti yang ada.

$$P(C|F) = \frac{P(F|C).P(C)}{P(F)}$$

Dimana :

a. P(C|F) adalah probabilitas posterior dari kelas C diberikan fitur F.
b. P(F|C) adalah probabilitas dari fitur F diberikan kelas C (likelihood).
c. P(C) adalah probabilitas awal dari kelas C (prior).
d. P(F) adalah probabilitas total dari fitur F (evidence).

Dalam Naïve Bayes, diasumsikan bahwa fitur-fitur independent satu sama lain, yang berarti:

$P(F1,F2,...,Fn|C)=P(F1|C)\times P(F2|C)\times ...\times P(Fn|C)$

Oleh karena itu, rumus untuk menghitung probabilitas posterior dengan Naïve Bayes adalah:

$$P(C|F1,F2,...,Fn) \propto P(C) \times P(F1|C) \times P(F2|C) \times ... \times P(Fn|C)$$

Langkah-langkah Menghitung Probabilitas Posterior

1. Mengumpulkan Data: Kumpulkan data dan hitung frekuensi setiap kelas dan fitur.
2. Menghitung Prior: Hitung probabilitas awal dari setiap kelas P(C)P(C)P(C).
3. Menghitung Likelihood: Hitung probabilitas setiap fitur diberikan kelas P(Fi|C)P(Fi|C)P(Fi|C).
4. Mengalikan Likelihood dengan Prior: Kalikan semua likelihood dari fitur-fitur yang diberikan kelas dengan prior untuk mendapatkan probabilitas posterior.

2.2.3 Data Mining

Data Mining telah menerima banyak perhatian di dunia sistem informasi dan masyarakat secara keseluruhan dalam beberapa tahun terakhir karena ketersediaan data dalam jumlah besar yang tersebar luas dan kebutuhan untuk mengubah data tersebut menjadi informasi dan pengetahuan yang berguna. Data Mining digunakan untuk mengekstrak atau menambang pengetahuan dari kumpulan data yang besar.

Data Mining menurut Nandang Iriadi (2021: 3), adalah proses iteratif yang digunakan untuk analisis basis data yang bertujuan menyaring informasi dan pengetahuan yang dapat membuktikan keakuratan data, berpotensi bagi para ahli ilmiah yang terlibat dalam pengambilan keputusan dan pemecahan masalah. Data Mining secara otomatis mengenali pola terkait dalam database.

2.2.4 Confusion Matrix

Tahap pengujian model dilakukan dengan menggunakan metode *confusion matrix* yang menampilkan hasil dari penilaian model melalui tabel matrik. Jika dataset memiliki dua kelas maka kelas yang pertama disebut positif dan kelas lainnya disebut negatif (D. Putra & Wibowo, 2020). *Confusion matrix* ini menggunakan tabel matriks seperti pada gambar dibawah dimana terdapat *record* perbandingan hasil klasifikasi data *testing* atau data uji berdasarkan data *training* atau data latih dengan data sebenarnya (Utomo & Mesran, 2020).

Tabel 2. 2 Confusion Matrix

	Prediksi Negatif	Prediksi Positif
Aktual Negatif	TN	FP
Aktual Positif	FN	TP

a. Akurasi

Metode pertama ialah akurasi, akurasi adalah metode yang didasari tingkat kedekatan antara nilai prediksi dengan nilai sebenarnya. Akurasi adalah hasil dari penjumlahan nilai diagonal dibagi dengan jumlah total keseluruhan data

dan selajutnya dikalikan 100%. Rumus akurasi dijabarkan pada persamaan berikut:

$$Akurasi = \frac{TP + TN}{TP + TN + FP + FN}$$

1. TP adalah *True Positive* (jumlah data positif yang diprediksi benar).

2. TN adalah *True Negative* (jumlah data negatif yang diprediksi benar).

3. FP adalah *False Positive* (jumlah data negatif yang salah diprediksi sebagai positif).

4. FN adalah *False Negative* (jumlah data positif yang salah diprediksi sebagai negatif).

b. Presisi

Presisi adalah metode yang dipakai untuk menghitung nilai proporsi kelas positif yang berhasil diprediksi dengan benar dari keseluruhan hasil prediksi kelas positif. Presisi menunjukkan jumlah data kategori positif yang diklasifikasi secara benar dibagi dengan total data yang diklasifikasi positif. Rumus presisi dijabarkan pada persamaan berikut:

$$Presisi = \frac{TP}{TP + FP}$$

1. TP adalah True Positive, yaitu jumlah data positif yang diprediksi benar oleh model

2. FP adalah False Postive, yaitu jumlah data negatif yang salah diprediksi sebagai positif oleh model.

c. Recall

Recall adalah metode yang digunakan untuk menghitung presentase kelas data positif yang berhasil diprediksi benar dari keseluruhan data kelas positif. Rumus *recall* dijabarkan pada persamaan berikut:

$$Recall = \frac{TP}{TP + FN}$$

1. TP adalah *True Positive,* yaitu jumlah data positif yang diprediksi benar oleh model
2. FN adalah *False Negative*, yaitu jumlah data positif yang salah diprediksi sebagai negatif oleh model.

d. F1-score

Metode *F1-score* ialah rata-rata harmonik dari presisi dan *recall. F1-score* juga biasa disebut *F1-measure*. Rumus *F1-score* dijabarkan pada persamaan berikut:

$$F1\ score = \frac{Presisi.Recall}{Presisi + Recall}$$

1. Presisi adalah presisi dari model ($\frac{TP}{TP+FP}$)
2. Recall adalah recall dari model ($\frac{TP}{TP+FN}$)

2.2.5 Python

Python adalah bahasa pemrograman *high-level*, interpretatif, multiguna, berorientasi objek dengan semantik dinamis. *Sintaks Python* yang sederhana dan mudah dipelajari menekankan keterbacaan dan karenanya mengurangi biaya pemeliharaan program. *Python* mendukung modul dan paket, yang mendorong modularitas program dan *code reuse*. Interpreter *Python* dan pustaka standar yang luas tersedia dalam bentuk *source* atau *biner* tanpa biaya untuk semua platform dan

dapat didistribusikan secara bebas. (*Python Software Foundation*, 2022).

Sebagai bahan pemrograman yang umum, *Python* dapat digunakan untuk memecahkan masalah-masalah numerik. Namun, jika dikombinasikan dengan pustaka-pustaka seperti *Numpy*, *Seaborn*, *Matplotlib*, dan *Pandas*. *Python* dapat secara efisien memproses masalah-masalah numerik dan visualisasi data. Maka dari itu, *Python* merupakan salah satu bahasa pemrograman yang paling sesuai untuk memvisualisasikan data.