BAB II
TINJAUAN PUSTAKA

2.1 Penelitian Terkait Terdahulu

Tabel 2.1 berikut ini merupakan penelitian terkait dengan penelitian yang sedang dilakukan saat ini :

Tabel 2.1 Penelitian Terkait

No	Nama Penulis	Judul/Tahun Terbit	Hasil
1	Irmawati, Kudiantoro Widianto, Faruq Aziz, Achmad Rifai, Ami Rahmawati	Implementasi Artificial Neural Network Dalam Mendeteksi Penyakit Hati (Liver) Journal of Information System, Applied, Management, Accounting and Research Vol. 6 No.1, Februari 2022	Penyakit hati akut dapat mempengaruhi fungsi hati, tetapi dapat mengidentifikasi gejala klinis dan fisik pasien. Salah satu masalah yang dihadapi masyarakat saat ini adalah keterlambatan pengobatan pasien penyakit liver, kebanyakan pasien tidak melakukan pemeriksaan sendiri sampai ditemukan stadium lanjut. Untuk mengatasi masalah tersebut, diperlukan suatu sistem yang dapat menentukan apakah seseorang merupakan penderita penyakit hati, sehingga dapat melakukan pemeriksaan rutin sesegera mungkin dan memungkinkan pasien penyakit hati untuk mendapatkan pengobatan tepat waktu. Sistem dapat

			menghasilkan klasifikasi dengan bantuan algoritma data mining. Dalam makalah ini, Pasien Hati telah diselidiki menggunakan model Artificial Neural Network untuk memprediksi seseorang Pasien Hati atau tidak dan analisis menggunakan ANN dengan Python digunakan untuk mengetahui pengaruh variabel input berdasarkan data dalam literature dan mendapat hasil akurasi sebesar 74%.
2	I Gusti Agung Putu Mahendra1, I Made Agus Wirawan2, I Gede Aris Gunadi3	Enhancement performance of the Naïve Bayes method using AdaBoost for classification of diabetes mellitus dataset type II. International Journal of Advances in Applied Sciences (IJAAS) Vol. 13, No. 3, September 2024, pp. 733~742 ISSN: 2252-8814, DOI: 10.11591/ijaas.v13.i3.pp733-742	Dari hasil evaluasi menggunakan validasi silang dari percobaan dengan total 890 data, Metode Naïve Bayes dengan AdaBoost memiliki akurasi tertinggi dengan 5 kali lipat dan 10 kali lipat, yaitu 85,36% dan masing-masing 87,95%. Sebaliknya, metode Naïve Bayes memiliki akurasi 79,44% dengan 5 kali lipat dan 82,02% dengan 10 kali lipat. Ada peningkatan akurasi 5,92% untuk 5 kali lipat dan 5,93% untuk 10 kali lipat. Si penambahan metode AdaBoost untuk klasifikasi diabetes memberikan nilai akurasi yang lebih tinggi dibandingkan

			dengan Algoritma Naïve Bayes tanpa AdaBoost. Jadi jelas bahwa menerapkan metode AdaBoost pada yang Naif Algoritma Bayes dapat meningkatkan akurasi.
3	Martika Kesuma, Sriyanto , Sutedi	Prediksi Penyakit Liver Menggunakan Algoritma Random Forest Jurnal informasi dan Komputer Vol: 11 No:2 2023	Peneliti ingin mengetahui apakah algoritma Random Forest memiliki nilai akurasi yang tinggi sehingga bisa menjadi landasan untuk menggunakan algoritma Random Forest dalam memprediksi penyakit liver. Peneliti menggunakan dataset Liver Disease Patient Dataset, dalam tahapan penelitian ini dilakukan beberapa langkah yang dimulai dari melakukan Analisis Data, Exploratory Data Analysis, Preprocessing, Pemodelan Algoritma, dan Visualisasi. Dari tahapan penelitian tersebut telah dapat diketahui hasil prediksi penyakit liver akurasi menggunakan Algortima Random Forest. Dari hasil penelitian yang dilakukan dengan algoritma Random Forest didapatkan prediksi dengan nilai akurasi 0.713326

			dengan f1 score 81%.
4	Indra Irawan, M Riski Qisthiano , Muhammad Syahril , Pamuji M. Jakak	Optimasi Prediksi Kelulusan Tepat Waktu: Studi Perbandingan Algoritma Random Forest dan Algoritma K-NN Berbasis PSO Jurnal Pengembangan Sistem Informasi dan Informatika e-ISSN 2746-1335 Vol.4, No. 4, Oktober 2023	Berdasarkan hasil penelitian dan pengujian yang telah dilakukan dengan menggunakan algoritma K-Nearest Neighbor (K-NN) dan Random Forest berbasis Particle Swarm Optimization (PSO) dalam menentukan tingkat kelulusan tepat waktu mahasiswa, ditemukan bahwa algoritma Random Forest memiliki akurasi sebesar 95,79% dan AUC sebesar 0,991. Di sisi lain, ketika algoritma Random Forest dioptimasi dengan PSO, akurasinya meningkat menjadi 97,89% dan AUC menjadi 0,993. Sedangkan untuk algoritma k-NN, akurasinya adalah 93,49% dengan AUC sebesar 0,975. Kemudian setelah dioptimasi dengan PSO, akurasinya menjadi 96,74% dengan AUC sebesar 0,986. Dengan demikian, hasil menunjukkan bahwa algoritma Random Forest yang dioptimasi dengan PSO lebih akurat dibandingkan dengan algoritma k-NN yang juga dioptimasi dengan PSO.

5	Siti Mutmainnah, Ginanjar Abdurrahman, Habibatul Azizah Al Faruq	Optimasi Algoritma C4.5 Menggunakan Teknik Bagging Pada Data Kadar Karat Emas	Untuk menentukan kualitas emas membutuhkan waktu yang lama. Dengan ini data mining dapat dimanfaatkan untuk mengklasifikasikan suatu kadar karat emas. Salah satu metode yang dapat digunakan adalah Algoritma C4.5. Namun metode ini memiliki tingkat akurasi yang kurang tinggi untuk itu digunakan teknik bagging guna meningkatkan akurasi dari Algoritma C4.5. Dari hasil penelitian, dengan menerapkan teknik bagging untuk klasifikasi berbasis ensemble pada algoritma C4.5 dapat meningkatkan akurasi sebesar 12.86%. Dengan akurasi awal 80%, setelah diterapkan teknik bagging menjadi 92.86%. sedangkan untuk presisinya terbagi menjadi 2 dimana nilai positif ≥ 22 memiliki hasil 80% meningkat sebanyak 3,33% menjadi 83,33%, lalu jika nilai positif pada < 22 sebesar 88,89% meningkat sebanyak 11,11% menjadi 100%.

Berdasarkan hasil review beberapa jurnal dengan ini saya menyimpulkan bahwasanya metode yang digunakan untuk beberapa jurnal yang saya review tingkat akurasi cenderung lebih tinggi dengan menggunakan optimasi PSO , Bagging, Adabosst.

2.2 Penyakit Hati

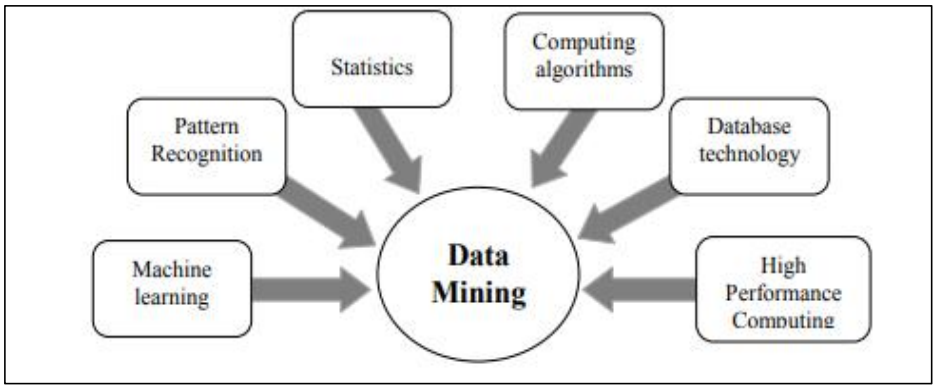
Hati adalah salah satu organ tambahan manusia, sehingga dapat melakukan pekerjaan yang sangat baik di dalam tubuh. Tapi ada stadium penyakit liver didalam tubuh manusia Penyakit hati dapat mempengaruhi fungsi hati, tetapi dapat mengidentifikasi gejala klinis dan fisik pasien Ada berbagai faktor yang menyebabkan penyakit liver ini terus meningkat. Diantaranya gaya hidup masyarakat yang rawan dan tidak terkontrol seperti : diabetes melitus, dislipidemia, hipertensi dan lain-lain [5].

2.3 Data Mining

Data mining dikenal sejak tahun 1990-an, ketika adanya suatu pekerjaan yang memanfaatkan data menjadi suatu hal yang lebih penting dalam berbagai bidang, seperti marketing dan bisnis, sains, dan teknik, serta seni dan hiburan. Sebagian ahli menyatakan bahwa *data mining* merupakan suatu langkah untuk menganalisis pengetahuan dalam basis data atau biasa disebut *Knowledge Discovery in Database* (KDD) . *Data mining* merupakan proses untuk menemukan pola data dan pengetahuan yang menarik dari kumpulan data yang sangat besar [6].

Data mining, secara sederhana merupakan suatu langkah ekstraksi untuk mendapatkan informasi penting yang sifatnya implisit dan belum diketahui. Data

mining mempunyai hubungan dengan berbagai bidang seperti statistic, machine learning, *computing algorithms, database technology.*



Gambar 2.1. Diagram Hubungan *Data Mining*

Secara sistematis, langkah utama untuk melakukan *data mining* terdiri dari tahap, yaitu sebagai berikut :

2.3.1 Eksplorasi Atau Pemprosesan Awal Data

Eksplorasi atau pemrosesan awal data terdiri dari pembersihan data, normalisasi data, transformasi data, penanganan missing value, reduksi dimensi, pemilihan subset fitur, dan sebagainya.

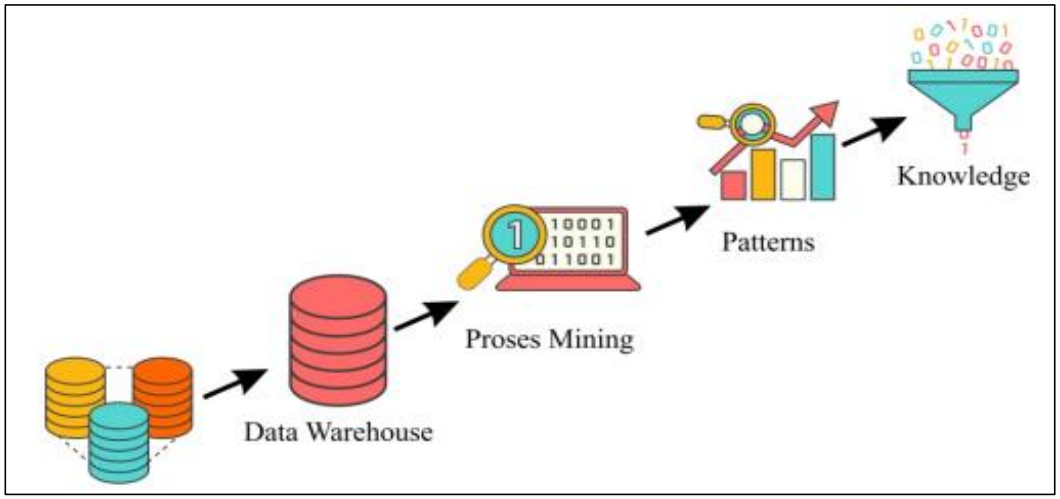
2.3.2 Membangun Model Dan Validasi

Membangun model dan validasi, merupakan melakukan analisis dari berbagai model dan memilih model sehingga menghasilkan kinerja yang terbaik. Pembangunan model dilakukan menggunakan metode-metode seperti klasifikasi, regresi, analisis cluster, dan asosiasi.

2.3.3 Penerapan

Penerapan dilakukan dengan menerapkan model yang dipilih pada data baru untuk menghasilkan kinerja yang baik pada masalah yang diinvestigasi.

Tahapan proses data mining ada beberapa yang sesuai dengan proses KDD (*Knowledge Discovery in Database*) .



Gambar 2.2. Proses KDD (*Knowledge Discovery in Database*)

1. *Cleaning And Integration.*

   a. *Data Cleaning* (Pembersih data)

   Data cleaning (Pembersihan data) adalah proses yang dilakukan untuk menghilangkan noise pada data yang tidak konsisten atau bisa disebut tidak relevan. Data yang diperoleh dari database suatu perusahaan maupun hasil eksperimen yang sudah ada, tidak semuanya memiliki isian yang sempurna misalnya data yang hilang, data yang tidak valid, atau bisa juga hanya sekedar salah ketik. Data yang tidak relevan itu dapat ditangani dengan cara dibuang atau sering disebut dengan proses cleaning. Proses cleaning dapat berpengaruh terhadap performa dari teknik *data mining*.

b. Data Integration (Integrasi Data)

Integrasi data merupakan proses penggabungan data dari berbagai database sehingga menjadi satu database baru. Data yang perlukan pada proses *data mining* tidak hanya berasal dari beberapa database.

2. *Selection and Transformation*

a. *Data Selection* (Seleksi Data)

Tidak semua data yang ada didatabase akan dipakai, karena hanya data yang sesuai saja yang akan dianalisis dan diambil dari database.Misalnya pada sebuah kasus market basket analysis yang akan meneliti faktor kecenderungan pelanggan, maka tidak perlu mengambil nama pelanggan, cukup dengan id pelanggan.

b. *Data Transformation* (Transformasi Data)

Transformasi data merupakan proses pengubahan data dan penggabungan data ke dalam format tertentu, *data mining* membutuhkan format data khusus sebelum diaplikasikan. Misalnya metode standar seperti analysis asosiasi dan clustering hanya bisa menerima inputan data yang bersifat katagorial. Karenanya data yang berupa angka numeric apabila mempunyai sifat kontinyu perlu dibagi menjadi beberapa interval. Proses ini sering disebut dengan transformasi data.

3. *Proses Mining*

Proses mining dapat disebut juga sebagai proses penambangan data. Proses mining merupakan proses utama yang menggunakan metode untuk menemukan pengetahuaan berharga yang tersembunyi dari data.

4. *Evaluation and Precentation*

a. Evaluasi Pola *(Pattren Evaluation)*

Evaluasi pola bertugas untuk mengidentifikasi pola-pola yang menarik ke dalam knowledge based yang ditemukan. Pada tahap ini dihasilkan pola-pola yang khas dari model klasifikasi yang dievaluasi untuk menilai apakah hipotesa yang ada memang tercapai. Bila ternyata hasil yang diperoleh tidak sesuai dengan hipotesa, terdapat beberapa alternative yang bisa diambil seperti menjadikanya umpan baik untuk memperbaiki proses *data mining*, atau mencoba metode *data mining* lain yang lebih sesuai.

b. Presentasi Pengetahuan *(Knowledge Presentation)*

Knowledge presentation merupakan visualisasi dan penyajian pengetahuan mengenai metode yang digunakan untuk memperoleh pengetahuan atau informasi yang telah digali oleh pengguna. Tahap terakhir dari proses data mining adalah memformulasikan keputusan dari hasil analisis yang didapat.

2.4 *Particle* ADABOOST (Adaptive Boosting)

Adaboost merupakan akronim dari Adaptive Boosting termasuk kedalam Ensemble Methods/Boosting Methods yang sering dipakai. Secara garis besar

proses yang dilakukan dalam Adaboost ialah membangun sejumlah weak learners yang tidak memiliki korelasi satu sama lain, lalu kemudian menggabungkan prediksinya. Dalam penerapannya Adaboost dikombinasikan dengan algoritma lain dengan tujuan untuk mengoptimalisasi performa yang dihasilkan. Adaboost $Hk\ (x)$ [7] didefinisikan sebagai:

$$H_k(x) = \sum_{t=1}^{T} \left(\frac{log\ 1}{\beta_t}\right) h_t^k(x) \qquad \text{.....(1)}$$

Dimana $ht\ \ k\ \ (x)$ merupakan weak learners yang memiliki nilai error terendah , sedangkan βt merupakan bobot dari weak learners tersebut. Premis akhir dalam Adaboost dihasilkan dari kombinasi weak learners yang memiliki nilai suara tertinggi

2.5 Particle Swarm Optimization *(*PSO)

Particle Swarm Optimization (PSO) adalah teknik optimasi yang sangat sederhana untuk menerapkan dan memodifikasi beberapa parameter. Dalam Particle Swarm Optimization (PSO), terdapat beberapa teknik untuk optimasi antara lain meningkatkan bobot atribut dari semua atribut atau variabel yang digunakan, memilih atribut (attribute selection), dan seleksi fitur [7]. Particle swarm optimization adalah suatu algoritma yang banyak terinspirasi dari perilaku sosial hewan seperti burung, lebah dan ikan. Seekor hewan dalam algoritma PSO akan dianggap sebagai partikel. Partikel ini akan dipengaruhi oleh kecerdasan dari individu hewan itu sendiri dan dan kecerdasan dari

partikel lain dalam satu kelompok. Apabila satu partikel menemukan jalan yang tepat dan terpendek menuju ke suatu sumber makanan, maka yang terjadi adalah partikelpartikel lain tersebut akan mengikuti partikel yang telah menemukan jalan yang tepat dan terpendek tadi [8]

Prosedur Particle Swarm Optimization (PSO) dapat dilakukan dalam beberapa langkah.

1. Kecepatan awal inisialisasi adalah 0 untuk semua partikel, seperti pada Persamaan 1.

$$(V_{i,j(t)}=0) \qquad \text{…….(2)}$$

Vi,j adalah kecepatan, j adalah letak dari partikel dan i adalah letak dari individu dan t merupakan iterasi.

2. Inisialisasi posisi awal partikel dengan batasan sesuai range [xmin,max]. proses inisialisasi posisi terdapat pada Persamaan 2.

$$(t)=x\text{min}+(x\text{max}-x\text{min}) \qquad \text{……. (3)}$$

X merupakan posisi partikel dan r adalah nilai random

3. Inisialisasi Pbest dan Gbest awal dimana pada iterasi ke 0 nilai Pbest sama dengan posisi awal sesuai dengan Persamaan 3 dan Gbest merupakan Pbest dengan nilai fitness terbaik.

$$(Pbesti,j(t)=xi,j(t)) \qquad \text{……. (4)}$$

Pbest merupakan personal best pada individu ke-i dan partikel ke-j. Xij merupakan posisi partikel

4. Update kecepatan dilakukan untuk menentukan arah perpindahan posisi partikel yang ada di populasi. Kecepatan dihitung sesuai Persamaan 4. Terdapat batasan untuk kecepatan yang digunakan yaitu berdasarkan nilai maksimum dan minimum posisi partikel untuk menentukan batas kecepatan maksimum dan minimum yang dipengaruhi oleh interval (k) yang sebaiknya dilakukan pada proses inisialisasi.

$$v_{i,j}^{t+1} = w.\ v_{i,j}^{t} + c_1 r_1 (Pbes_{i,j}^{t} - x_{i,j}^{t}) + c_2 r_2 (Gbes_{g,j}^{t} - x_{i,j}^{t}) \quad \ldots\ldots\ldots\ldots\ldots (5)$$

$$vj^{max} = k\frac{(xj,max - xj,\min)}{2}\ k \in (0,1] \quad \ldots\ldots\ldots\ldots\ldots (6)$$

$$if\ v_{ij}(t+1) > v_j^{max}\ then\ v_{ij}(t+1) = vj^{max} \quad \ldots\ldots\ldots\ldots\ldots (7)$$

$$if\ v_{ij}(t+1) < -v_j^{max}\ then\ v_{ij}(t+1) = -v_j^{max} \quad \ldots\ldots\ldots\ldots\ldots (8)$$

Nilai c1 dan c2 adalah koefisien akselerasi, nilai r1 dan r2 adalah partikel random, nilai w adalah bobot inertia.

5. Perbaharuan posisi dilakukan untuk menentukan posisi terbaru dari setiap partikel berdasarkan hasil update kecepatan sebelumnya. Setelah didapatkan nilai kecepatan maka dilanjutan dengan perhitungan sigmoid dari kecepatan tersebut sesuai dengan Persamaan sebelumnya kemudian hasil signoid yang telah didapat akan diproses lebih lanjut pada Persamaan sebelumnya sehingga

didapatkan posisi terbaru Setelah itu menentukan hasil fitness terbaru yang tentunya juga akan mendapat nilai Pbest terbaru.

$$sig(v_{i,j}^{t}) = \frac{1}{2+e-v_{i,j}^{t}},\ j = 1,2,..d \qquad \text{..... (9)}$$

$$if\ rand[0,1] > sig(v_{i,j}^{t})\ then\ x_{i,j}^{t+1} = 0$$
$$if\ rand[0,1] < sig(v_{i,j}^{t})\ then\ x_{i,j}^{t+1} = 1 \qquad \text{..... (10)}$$
$$j = 1,2,\ldots.d\ (20)$$

6. Perbaharuan Pbest, yaitu dengan membandingkan nilai fitness dari Pbest pada iterasi sebelumnya dengan fitness dari perbaharuan posisi. Nilai yang terbaik akan menjadi Pbest yang baru pada iterasi selanjutnya.

$$k = 1 + decimal(s1)\ x\frac{a-1}{2n1-1} \qquad \text{..... (11)}$$

2.6 K-Nearest Neighbor

Algoritma KNN merupakan algoritma yang banayak digunakan dalam melakukan klasifikasi. KNN merupakan algoritma yang sederhana untuk diimplementasikan tetapi menghasilkan akurasi yang baik .Salah satu kelemahan dari algoritma ini adalah dalam penentuan nilai k, jika nikai k terlalu besar maka akan membuat hasil klasifikasi menjadi tidak jelas atau kabur, sedangkan jika nilai k terlalu kecil atau di misalkan k=1 maka akan menyebabkan hasil klasifikasi terasa kaku karena tidak ada pilihan .Maka dari itu diperlukan penelitian penentuan nilai k yang baik [9]. Algortima K-Nearest Neighbor (KNN) adalah merupakan sebuah metode untuk melakukan klasifikasi terhadap obyek baru berdasarkan (K) tetangga terdekatnya. KNN termasuk algoritma supervised learning, dimana hasil

dari query instance yang baru, diklasifikasikan berdasarkan mayoritas dari kategori pada KNN. Kelas yang paling banyak muncul yang akan menjadi kelas hasil klasifikasi[10]. Jarak terdekat antara data training akan diukur dengan euclidean distance dan manhattan distance. Menurut penelitian terdahulu, euclidean distance mempunyai hasil lebih akurat dari pada manhattan distance [11].

2.7 Bagging

Bagging adalah singkatan dari bootstrap aggregating, menggunakan subdataset (bootstrap) untuk menghasilkan set pelatihan L (learning), L melatih dasar belajar menggunakan prosedur pembelajaran yang tidak stabil, dan kemudian, selama pengujian, mengambil rata-rata [13]. Bagging baik digunakan untuk klasifikasi dan regresi. Model ensemble berlaku dengan penggabungan berbagai teknik pengambilan sampel seperti bagging, boosting, dan lain-lain untuk menjamin keragaman di kolom classifier. Penanganan noise data merupakan masalah penting untuk proses pembelajaran klasifikasi, sejak terjadinya noise yang tinggi dalam proses pelatihan atau pengujian (klasifikasi) pada dataset, mempengaruhi keakuratan prediksi pengklasifikasi yang dipelajari.

2.8 Confusion Matrix

Matriks konfigurasi adalah tabel yang terdiri dari jumlah baris data uji yang diprediksi benar dan salah dengan model klasifikasi yang digunakan. Tabel

Confusion Matrix diperlukan untuk memilih kinerja terbaik dari sebuah model klasifikasi [14]. Confusion matrix adalah matrix 2x2 yang merepresentasikan hasil klasifikasi biner pada suatu dataset. Terdapat beberapa rumus umum yang dapat digunakan untuk menghitung performa klasifikasi. Hasil dari nilai accuracy, precision dan recall bisa ditampilkan dalam persentase [15].

2.8.1 Accuracy (Akurasi)

Akurasi adalah salah satu metrik untuk mengevaluasi model klasifikasi. Secara informal, akurasi adalah sebagian kecil dari prediksi model kami yang benar. Secara formal, akurasi memiliki definisi sebagai berikut: Untuk klasifikasi biner, akurasi juga dapat dihitung dalam hal positif dan negatif sebagai berikut:

$$\text{Akurasi} = \frac{\text{Number of Correect Prediction}}{\text{Total Number of Prediction}} \quad \ldots\ldots\ldots\ldots\ldots (12)$$

Untuk klasifikasi biner, akurasi juga dapat dihitung dalam hal positif dan negatif sebagai berikut:

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

Dimana TP = True Positif

TN = True Negatif

FP = False Positif

FN = False Negatif

2.8.2 Precision

Precision dalam Confusion Matrix didefinisikan sebagai rasio item terkait yang dipilih dengan semua item yang dipilih. Akurasi adalah kemungkinan bahwa item yang dipilih terkait. Dapat diartikan sebagai kecocokan antara permintaan informasi dan respons terhadap permintaan itu [16]

2.8.3 Recall

Recall adalah proporsi jumlah dokumen teks yang relevan terkendali diantara semua dokumen teks relevan yang ada pada koleksi [15].Recall merupakan probabilitas bahwa suatu item yang relevan akan dipilih. Recall dapat dihitung dengan jumlah rekomendasi yang relevan yang dipilih oleh user dibagi dengan jumlah semua rekomendasi yang relevan baik dipilih maupun rekomendasi yang tidak terpilih [16]