

BAB II LANDASAN TEORI

2.1 Tinjauan Pustaka

Dalam melakukan penelitian ini peneliti mengambil tinjauan pustaka sebagai salah satu alat penerapan metode penelitian. Tinjauan Pustaka ini juga bertujuan untuk membandingkan hasil penelitian yang satu dengan yang lain. Berikut Daftar Literatur penelitian dapat dilihat pada tabel 2.1 dibawah ini

Tabel 2. 1 Daftar Literatur

No	Nama Peneliti	Judul	Hasil	Perbedaan Penelitian
1	Aldi et al 2016	Klasifikasi Pelanggan Pada Customer Churn Prediction Menggunakan Decision Tree	Dari hasil klasifikasi, menunjukkan 636 pelanggan loyal dan 357 pelanggan <i>churn</i>	Dalam penelitian ini metode yang digunakan yaitu Decision Tree C4.5
2	Istiqomah, Susanti & Zukhromah 2019	Klasifikasi Status Kredit Nasabah BMT (Baitul Maal Wa Tamwil) Menggunakan Algoritma C5.0	Hasil yang didapatkan dari penelitian ini pohon keputusan menghasilkan 7 klasifikasi menghasilkan keakuratan sebesar 83.8% serta eror sebesar 16.2%.	Dalam penelitian ini menggunakan metode C5.0 untuk memprediksi kemacetan kredit ataupun tunggakan pada BMT
3	Wardani et al 2018	Prediksi Customer Churn Dengan Algoritma Decision Tree C4.5 Berdasarkan Segmentasi Pelanggan Pada Perusahaan Retail	Hasil yang didapatkan yaitu mengetahui bahwa pelanggan UD. Mawar Sari berpotensi churn adalah pelanggan pada kelas Dorman	Dalam penelitian ini digunakan metode algoritma C4.5 untuk mengetahui kelas pelanggan yang melakukan churn

Tabel 2. 2 Daftar Literatur (Lanjutan)

No	Nama Peneliti	Judul	Hasil	Perbedaan Penelitian
4	Sungkar & Qurohman 2021	Penerapan Algoritma C5.0 Untuk Prediksi Kelulusan Pembelajaran Mahasiswa Pada Matakuliah Arsitektur sistem Komputer.	Dari hasil penelitian mendapatkan nilai akurasi yang tinggi yaitu sebesar 93.3%	Penelitian tersebut menggunakan algoritma C5.0 untuk mengetahui apakah mahasiswa lulus pada mata kuliah Arsitektur Sistem computer
5	Murni Marcania 2020	Prediksi Pengangkatan Karyawan Dengan Metode Klasifikasi Algoritma C5.0 (Studi Kasus PT. Kiyokunii Indonesia Factory-2)	Dari penelitian tersebut didapat hasil <i>accuracy</i> sebesar 96.75%. disimpulkan bahwa memprediksi menggunakan algoritma C5.0 dinilai berhasil	Algoritma C5.0 digunakan sebagai metode memprediksi pengangkatan karyawan
6	Nurhaningsih, Susanti & Handajani 2019	Implementasi Algoritma C5.0 Untuk Klasifikasi Penyakit Gagal Ginjal Kronik	Hasil dari nilai akurasi yang diperoleh adalah 99.3%	Dalam penelitian ini menggunakan algoritma C5.0 untuk memprediksikan penyakit ginjal kronik

2.1.1 Literatur 1

Klasifikasi Pelanggan Pada Customer Churn Prediction Menggunakan Decision Tree. Pada penelitian ini bertujuan untuk mengklasifikasi pelanggan. Penentuan proses klasifikasi data dalam menemukan informasi dan pengetahuan dilakukan menggunakan data mining dengan menggunakan salah satu metodenya untuk prediksi yaitu metode *decision tree*, representasi sederhana dari teknik klasifikasi.

Dalam penelitian ini algoritma yang digunakan untuk membangun *decision tree* adalah algoritma C4.5 dan data yang digunakan yaitu berjumlah 996 *data set sample* pelanggan dengan atribut gender, umur, metode pembayaran, dan jumlah transaksi. Klasifikasi dilakukan berdasarkan nilai *confidence* pelanggan termasuk loyal atau *churn*. Dari hasil klasifikasi, menunjukkan 61 pelanggan loyal dan 35 pelanggan *churn*, kemudian diakumulasi total pelanggan loyal menjadi 636 pelanggan dan pelanggan *churn* terdapat 357 pelanggan.

2.1.2 Literatur 2

Klasifikasi Status Kredit Nasabah BMT (*Baitul Maal wa Tamwil*) Menggunakan Algoritma C5.0. Pada penelitian ini dilakukan bertujuan untuk mengetahui penyebab dari nasabah yang berstatus kredit macet. Klasifikasi berdasarkan Pendidikan terakhir, jenis pekerjaan, penghasilan, serta jangka peminjaman dapat dijadikan oleh pihak Lembaga sebagai acuan untuk mengantisipasi terjadinya tunggakan atau kredit macet. Pohon keputusan yang digunakan pada penelitian ini adalah Algoritma *Commercial Version 5.0 (C5.0)*. Atribut yang masuk pada pohon keputusan adalah atribut Pendidikan, pendapatan, dan jangka waktu peminjaman. Pohon keputusan menghasilkan 7 klasifikasi. Klasifikasi tersebut menghasilkan keakuratan sebesar 83.8% serta eror sebesar 16.2%. dari hal tersebut dapat disimpulkan bahwa klasifikasi menggunakan Algoritma C5.0 menghasilkan akurasi yang cukup tinggi sehingga dapat digunakan oleh pihak Lembaga BMT dalam melihat klasifikasi lancar atau tidaknya kredit nasabah.

2.1.3 Literatur 3

Prediksi Customer Churn Dengan Algoritma Decision Tree C4.5 Berdasarkan Segmentasi Pelanggan Pada Perusahaan Retail. Pada penelitian ini dari proses segmentasi pelanggan menggunakan pemodelan RFM (*Recency, Frequency, Monetary*) dapat diketahui karakteristik pelanggan UD. Mawar Sari. Dari proses segmentasi tersebut dapat diketahui kelas pelanggan yang dimiliki UD. Mawar Sari yaitu kelas *Dormant, Everyday, Golden* dan *Superstar*. Hasil kinerja dari algoritma decision tree C4.5 pada kelas pelanggan *Dormant* yaitu *recall* 97.51%, *precision* 75.18% dan *accuracy* 76.18%. Pada kelas pelanggan *everyday* yaitu *recall* 100%, *precision* 99.04% dan *accuracy* 99.04%. Hasil kinerja pada kelas pelanggan *golden* yaitu *recall* 100%, *precision* 98.84% dan *accuracy* 98.84%. Dan hasil kinerja pada kelas pelanggan *superstar* yaitu *recall* 96.15%, *precision* 99.43% dan *accuracy* 95.63%. dan dari hasil tersebut dapat disimpulkan bahwa pelanggan UD. Mawar Sari yang berpotensi *churn* adalah kelas pelanggan *Dormant*.

2.1.4 Literatur 4

Penerapan Algoritma C5.0 Untuk Prediksi Kelulusan Pembelajaran Mahasiswa Pada Matakuliah Arsitektur Sistem Komputer. Pada penelitian ini menggunakan Algoritma C5.0 untuk memprediksi kelulusan nilai mahasiswa/I pada matakuliah arsitektur sistem komputer. Proses prediksi kelulusan dilakukan berdasarkan dengan klasifikasi Algoritma C5.0 dengan atribut yang digunakan yaitu Nilai Kehadiran, Nilai Tugas, Nilai Ujian Tengah Semester, dan Nilai Ujian Akhir. Hasil akhir dari proses klasifikasi Algoritma C5.0 adalah pohon keputusan dengan rule didalamnya. Dari kinerja klasifikasi Algoritma C5.0 mendapatkan nilai akurasi yang tinggi yaitu sebesar 93.33%.

2.1.5 Literatur 5

Prediksi Pengangkatan Karyawan Dengan Metode Klasifikasi Algoritma C5.0 (Studi Kasus PT. Kiyokunii Indonesia Factory-2). Pada penelitian ini bertujuan untuk menentukan pengangkatan karyawan tetap di PT. Kiyokuni Indonesia Factory-2 yang dilihat melalui beberapa kriteria yang telah ditetapkan oleh perusahaan menggunakan Algoritma C5.0 dengan metode *decision tree*. Dengan menggunakan sebanyak 500 data karyawan PT. Kiyokuni Indonesia Factory-2, proses pengujian menggunakan metode *RapidMiner9.5*. Dari hasil pengujian penelitian dalam memprediksi pengangkatan karyawan tetap PT. Kiyokunii Indonesia Factory-2, maka memperoleh hasil dari Algoritma C5.0 atau *Decision Tree* yaitu tingkat *accuracy* sebesar 96.75%. Dengan hasil *accuracy* tersebut maka dapat dikatakan bahwa menggunakan metode Algoritma C5.0 untuk memprediksi pengangkatan karyawan tetap pada PT. Kiyokunii Indonesia Factory-2 dinilai berhasil.

2.1.6 Literatur 6

Implementasi Algoritma C5.0 Untuk Klasifikasi Penyakit Gagal Ginjal Kronik. Pada penelitian ini menggunakan pengklasifikasian metode Algoritma C5.0 yang bertujuan untuk memprediksi penyakit gagal ginjal kronik dan menghitung nilai akurasi. Berdasarkan hasil penelitian menggunakan metode Algoritma C5.0 pada klasifikasi penyakit gagal ginjal kronik dengan faktor eritrosit, urea, kreatinin, dan trombosit mempunyai struktur pohon klasifikasi dengan jumlah 6 tingkatan klasifikasi. Hasil dari pengujian nilai akurasi yang diperoleh sebesar 99.3%.

2.2 Pre-Processing

Preprocessing merupakan proses awal yang akan mengubah dataset awal menjadi data dengan format yang sesuai dan siap untuk diproses. Beberapa contoh hal yang dilakukan dalam *preprocessing* meliputi berbagai proses yang antara lain:

a. Pembersihan data (*Data Cleaning*)

Pembersihan data merupakan proses menghilangkan data yang tidak lengkap dan tidak konsisten

b. Integrasi data (*Data Integration*)

Data dari berbagai database digabungkan menjadi satu dalam *database* baru

c. Seleksi data (*Data Selection*)

Data yang ada pada database sering kali tidak semuanya dipakai, oleh karena itu hanya data yang sesuai untuk dianalisis yang akan diambil dari database.

d. Transformasi data (*Data Transformation*)

Data diubah ke dalam format yang sesuai untuk diproses dalam Data Mining. (Agarwal, 2014)

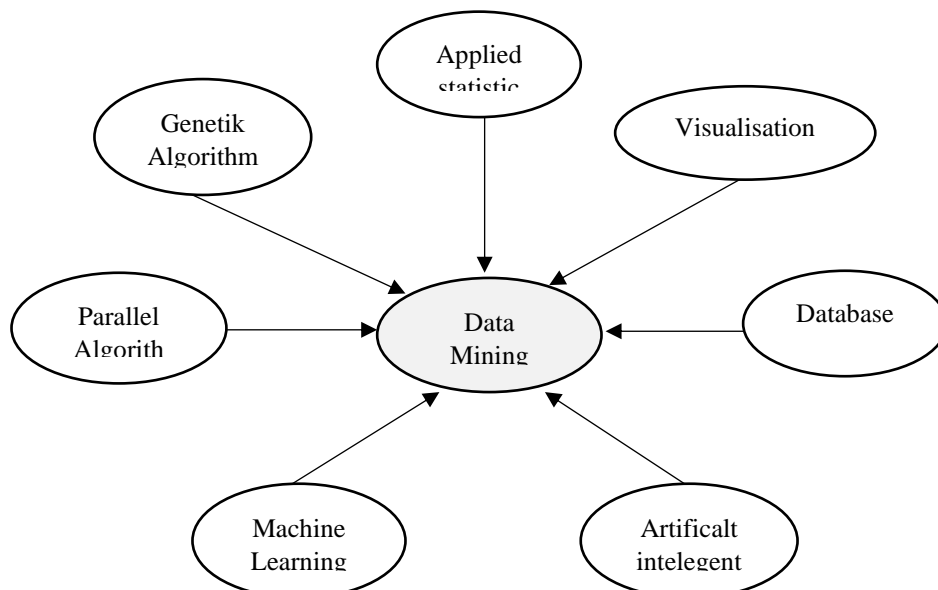
Proses yang ada dalam tahap *preprocessing* dapat terdiri dari salah satu kegiatan proses ataupun gabungan dari beberapa proses diatas. Proses yang ada tergantung dari tujuan yang akan dicapai dalam *preprocessing* tersebut (Karthick & Malathi, 2015).

2.3 Data Mining

Data mining didefinisikan sebagai proses mencari data dan suatu informasi dalam jumlah besar dari suatu tempat penyimpanan (*database*) informasi lainnya. Biasanya proses data mining juga menemukan pola yang unik dari suatu informasi. Berdasarkan tugas yang dapat dilakukan data mining dibagi menjadi beberapa kelompok, yaitu deskripsi, estimasi, prediksi, klasifikasi, pengklusteran dan

asosiasi. Pola yang ditemukan pada proses data mining harus memiliki arti dan juga memberikan keuntungan bagi penggunanya.

Data mining ialah suatu metode atau cara untuk menggali dan mendapatkan informasi berharga dan tersembunyi dari suatu *database* yang besar (Marcania, 2020). Output dalam data mining dapat dijadikan sebagai bahan acuan dalam pengambilan keputusan ataupun untuk memperbaiki keputusan dimasa yang akan datang. Perlu diketahui bahwa data mining merupakan salah satu bidang yang cukup banyak didukung oleh cabang ilmu lain di dalam teknologi informasi yaitu statistik, teknologi basis data, *machine learning*, sistem pakar, algoritma paralel, algoritma genetika, pengenalan pola, visualisasi data, dan lain sebagainya.



Gambar 2. 1 Data Mining Multidisipliner

2.2.1 Teknik-Teknik Data Mining

Ada beberapa Teknik dalam data mining yaitu sebagai berikut:

1. *Predictive Modelling* merupakan pengolahan data mining dengan dengan melakukan peramalan yang bertujuan untuk membangun model peramalan sebuah nilai yang memiliki ciri-ciri tertentu.

2. *Association* (Asosiasi) merupakan teknik yang terdapat dalam data mining yang mempelajari hubungan antar data.
3. *Clustering* (Klastering) teknik yang mengelompokan data ke dalam sebuah kelompok tertentu.
4. *Classification* (Klasifikasi) adalah pengklasifikasian data. Perbedaan klasifikasi dengan klastering adalah terdapat pada data, pada klastering variabel dependen tidak ada, sedangkan pada klasifikasi harus ada.

2.2.2 Proses Data Mining

Ada tiga langkah utama dalam data mining secara sistematis:

1. Pemrosesan awal data/Eksplorasi

Pemrosesan awal data/Eksplorasi terdiri dari pembersihan data, normalisasi data, transformasi data, penanganan data yang salah, reduksi dimensi, pemilihan subset fitur, dan lain sebagainya.

2. Membangun model dan melakukan validasi terhadapnya

Membangun model dan melakukan validasi terhadapnya berarti melakukan analisis berbagai model dan memilih model dengan kinerja prediksi yang terbaik. Dalam langkah ini digunakan metode-metode seperti klasifikasi, regresi, analisis cluster, deteksi anomali, analisis asosiasi, analisis pola sekuensial, dan sebagainya. Dalam beberapa referensi, deteksi anomali juga masuk dalam Langkah eksplorasi. Akan tetapi, deteksi anomali juga dapat digunakan sebagai algoritma utama, terutama untuk mencari data-data yang spesial.

3. Penerapan

Langkah ini menerapkan model pada data yang baru untuk menghasilkan perkiraan/prediksi masalah yang diinvestigasi.

2.3 Klasifikasi

Klasifikasi ialah proses membagi sekumpulan data sehingga setiap data menjadi anggota suatu kategori atau kelas. Pengelompokan berdasarkan hubungan antara variabel kriteria dengan variabel target. Klasifikasi adalah proses dari mencari suatu himpunan model (fungsi) yang dapat mendeskripsikan dan membedakan kelas-kelas data atau konsep-konsep, yang bertujuan menggunakan model tersebut agar dapat memprediksi kelas dari suatu objek yang mana kelasnya belum diketahui (Rani, 2015). Berikut tujuan dari klasifikasi adalah:

1. Mendapatkan sebuah model dari data training dimana dapat membedakan *record* ke kelas yang sesuai, setelah ini model digunakan untuk mengklasifikasi *record* yang mana kelas belum diketahui pada data testing sebelumnya.
2. Mengambil sebuah keputusan dengan memprediksi suatu masalah atau kasus, berdasarkan hasil dari model klasifikasi yang diperoleh.

2.4 *Decision Tree* (pohon keputusan)

Decision tree merupakan salah satu teknik yang dapat digunakan untuk melakukan klasifikasi terhadap sekumpulan objek atau record. teknik ini terdiri dari kumpulan *decision tree*, dihubungkan oleh cabang, bergerak ke bawah dari node sampai berakhir di *leaf node*. *Decision tree* merupakan pohon keputusan pada teknik klasifikasi untuk mempresentasikan aturan (Istiqomah et al, 2019). *Decision tree* dibentuk melalui pengklasifikasian variabel prediktor dan variabel respon.

Kedua variabel tersebut dikenal dengan nama atribut. Atribut digunakan sebagai acuan untuk mendapatkan keputusan berupa *if-then*.

Pada proses ini, *record* akan diubah menjadi bentuk pohon keputusan. Berikut adalah tiga *node* (simpul) yang terdapat pada pohon keputusan.

1. *Root node* atau *node* akar, merupakan simpul paling atas yang memiliki cabang lebih dari satu.
2. *Internal node* atau *node* cabang, merupakan simpul yang terdapat satu cabang, merupakan simpul yang terdapat satu cabang masuk dan cabang keluar lebih dari satu.
3. *Leaf node* atau *node* terminal, merupakan simpul terakhir yang tidak memiliki cabang keluar.

Beberapa keunggulan yang dimiliki *decision tree* sebagai alat pendukung keputusan, yaitu:

- a. Mudah untuk dimengerti dan dipahami. Orang-orang bisa memahami model *decision tree* dengan penjelasan singkat.
- b. Memiliki nilai walaupun dengan sedikit data yang rumit. Wawasan penting dapat dihasilkan berdasarkan para ahli yang menggambarkan situasi dan preferensi mereka untuk hasil.
- c. Membantu menentukan nilai terburuk, terbaik, dan nilai yang diharapkan untuk berbagai skenario.
- d. Menggunakan model kotak putih jika hasil diberikan oleh model.
- e. Dapat dikombinasikan dengan teknik pengambilan keputusan lainnya (Marcania, 2020).

2.5 Algoritma C5.0

Algoritma C5.0 merupakan Algoritma klasifikasi yang menghasilkan pohon keputusan ditemukan oleh Ross Quinlan pada tahun 1987. Algoritma ini merupakan penyempurnaan dari Algoritma ID3 dan C4.5. Algoritma C5.0 lebih baik dari pada Algoritma C4.5 dalam hal memori dan akurasi. Kecepatan dalam membuat pohon keputusan melalui software dinilai sangat cepat dibandingkan Algoritma lainnya. Algoritma C5.0 menerapkan model berbasis aturan (*rule based model*) sehingga memudahkan untuk melihat aturan pada pohon keputusan. Selain itu, Algoritma C5.0 dapat mengatasi nilai yang hilang (*missing value*). Hal inilah yang menjadi kelebihan dari Algoritma C5.0.

Algoritma C5.0 adalah salah satu algoritma klasifikasi data mining yang mana pemilihan atribut yang akan diproses menggunakan *information gain* sebagai kriterianya. Algoritma C5.0 memiliki fitur yang membuatnya lebih unggul dari pada algoritma terdahulunya., fitur tersebut antara lain:

1. Algoritma C5.0 dirancang untuk menganalisis basis data substansial yang berisi puluhan sampai ratusan *record* dan satuan hingga ratusan *field numeric* dan nominal. Memaksimalkan tingkat penafsiran pengguna terhadap hasil yang disajikan dalam dua bentuk yaitu menggunakan pohon keputusan (*decision tree*) dan sekumpulan aturan *IF-Then* yang mudah dimengerti.
2. Algoritma C5.0 mudah digunakan dan tidak membutuhkan pengetahuan yang tinggi tentang statistik atau *machine learning*.
3. Algoritma C5.0 merupakan penyempurnaan dari algoritma ID3 dan C4.5. Dalam proses pembentukan pohon keputusan nilai informasi gain tertinggi akan terpilih sebagai *root* bagi *node* selanjutnya. Algoritma ini dimulai dengan

semua data yang dijadikan akar dari pohon keputusan sedangkan atribut yang dipilih akan menjadi pembagi bagi sampel tersebut.

Formula ukuran atribut adalah:

$$Info(D) = - \sum_{i=1}^m p_i \log_2(p_i) \dots \dots \dots (2.1)$$

Dengan info (D) merupakan informasi yang dibutuhkan untuk mengklasifikasi label kelas sepuluh *tuple* di D. p_i adalah peluang bukan nol dengan sebuah *tuple* acak di D. Fungsi log menggunakan basis 2, karena informasi yang dikode dalam bit. Info (D) juga kenal sebagai *entropy*. Nilai *entropy* yang dihasilkan untuk mengklasifikasi tupel dari D berdasarkan partisi oleh A adalah:

$$Info A(D) = \sum_{j=1}^Y \frac{|D_j|}{D} \times Info(D_j) \dots \dots \dots (2.2)$$

Untuk mendapatkan nilai *gain ratio* pada atribut A selanjutnya digunakan rumus:

$$GAIN\ Ratio(A) = \frac{Gain(S,A)}{\sum_{i=1}^n Entropy(S_i)} \dots \dots \dots (2.3)$$

Gain (A) menyatakan berapa banyak cabang yang akan diperoleh pada A. atribut A dengan *information gain* tertinggi. Informasi, *Gain* (A), dipilih sebagai atribut pada *node* N. *Gain* (S,A) merupakan nilai gain dari variabel $\sum_{i=1}^n Entropy(s_i)$ merupakan banyaknya nilai entropy dalam suatu variabel.

2.6 Confusion Matrix

Confusion matrix adalah tool yang banyak digunakan untuk evaluasi model klasifikasi untuk memperkirakan objek yang benar atau salah.

Rumus presisi, akurasi, dan *recall*:

1. Akurasi : tingkat kedekatan antara nilai prediksi dengan nilai actual
2. Presisi : tingkat ketepatan antara informasi yang diminta oleh

pengguna dengan jawaban yang diberikan oleh sistem.

3. Recall : tingkat keberhasilan sistem dalam menentukan Kembali sebuah informasi (Marcania, 2020)

Keakuratan hasil klasifikasi dapat diukur dengan menggunakan *confusion matrix*. *Confusion matrix* adalah media yang berguna untuk menganalisis seberapa baik *Classifier* dapat mengenali tupel dari kelas yang berbeda. Tabel 2.2 berikut menunjukkan Confusion Matrix.

Tabel 2.1 Confusion Matrix
(Sumber: Suyanto, 2019)

		Kelas actual	
		<i>True</i>	<i>False</i>
Kelola Hasil Prediksi	<i>True</i>	TP (<i>True Positive</i>)	FP (<i>False Positive</i>)
	<i>False</i>	FN (<i>False Negative</i>)	TN (<i>True Negative</i>)

Berikut keterangan Tabel 2.2:

1. TP (*True Positive*) yaitu jumlah positif yang dilabeli dengan benar oleh *classifier*.
2. TN (*True Negative*) yaitu jumlah negatif yang dilabeli benar oleh *classifier*.
3. FP (*False Positive*) yaitu jumlah positif yang dilabeli salah oleh *classifier*.
4. FN (*False Negative*) yaitu jumlah negatif yang dilabeli benar oleh *classifier*.

2.6.1 Akurasi

Akurasi merupakan hasil dari semua perhitungan nilai prediksi yang benar dan dibagi dengan keseluruhan data. Nilai akurasi terbaik jika nilai akurasi tersebut sama dengan 1 dan yang terburuk 0.

$$Akurasi = \frac{TP+TN}{TP+TN+FN+FP} \dots\dots\dots (2.4)$$

Keterangan:

TP (*True Positive*) = hasil prediksi sesuai dengan nilai yang sebenarnya yaitu sama-sama benar

TN (*True Negative*) = hasil prediksi salah dan nilai yang sebenarnya juga salah

FN (*false Negative*) = hasil prediksi salah, namun hasil sebenarnya benar

FP (*False Positive*) = hasil prediksi benar, namun hasil sebenarnya salah

2.6.2 Sensitivity (Recall)

Sensitivity (SN) disebut juga *Recall* atau *True positive rate* merupakan jumlah prediksi yang benar dibagi dengan keseluruhan jumlah kelas yang salah. Nilai SN terbaik yaitu 1 dan nilai terburuk adalah 0.

$$Recall = \frac{TP}{TP+FN} \dots\dots\dots (2.5)$$

2.6.3 Precision (Positive Predictive Value)

Precision didapat dari menghitung jumlah seluruh nilai produktif positif yang benar dibagi dengan jumlah keseluruhan prediksi kelas yang benar. Nilai 1 merupakan nilai terbaik precision sedangkan 0 yang terburuk.

$$Precision = \frac{TP}{TP+FP} \dots\dots\dots (2.6)$$

2.6.4 F-1 Score

F-1 *Score* atau biasa disebut juga *F1-Measure* merupakan metrik yang menutupi kekurangan pada *recall* dan *precision* di dalam penilaian performa terhadap kelas positif dengan cara menghitung rata-rata dari keduanya (Nuraliza dkk, 2019).

$$F1 = 2 X \frac{Precision \times Recall}{Precision + Recall} \dots\dots\dots (2.7)$$

2.7 Python

Python merupakan bahasa pemrograman yang interpretative dan multiguna dengan filosofi penyusunan yang berpusat pada tingkat keterbacaan kode. Python sebagai Bahasa pemrograman memiliki fungsi sebagai Bahasa yang berfungsi untuk menggabungkan kapabilitas, kemampuan, dengan tata kalimat kode yang sangat jelas yang mempunyai fungsionalitas pustaka umum yang besar serta lengkap. Beberapa fitur yang terdapat pada python ini adalah sebagai bahasa pemrograman dinamis yang dilengkapi dengan pengelolaan memori secara otomatis. Bahasa pemrograman python pada umumnya dimanfaatkan sebagai suatu bahasa skrip walaupun pada saat penerapannya penggunaan bahasa pemrograman python ini lebih luas dan memuat konteks pemanfaatan yang umumnya tidak dikerjakan dengan memanfaatkan bahasa skrip.

Bahasa *python* juga banyak digunakan oleh programmer karena dikenal dengan bahasa pemrograman yang mudah dipelajari serta karena struktur sintaknya rapi dan mudah untuk dipahami (Nisa & Miftahul, 2020).