



### **Hak cipta dan penggunaan kembali:**

Lisensi ini mengizinkan setiap orang untuk menggubah, memperbaiki, dan membuat ciptaan turunan bukan untuk kepentingan komersial, selama anda mencantumkan nama penulis dan melisensikan ciptaan turunan dengan syarat yang serupa dengan ciptaan asli.

### **Copyright and reuse:**

This license lets you remix, tweak, and build upon work non-commercially, as long as you credit the origin creator and license it on your new creations under the identical terms.

## BAB II

### LANDASAN TEORI

#### 2.1 Profil Karyawan

Profil karyawan menurut Cahyadi (2017) adalah informasi-informasi personal milik seorang karyawan. Selain informasi personal, profil karyawan di Kompas Gramedia memuat informasi akademis karyawan, riwayat mutasi, nilai *performance* karyawan, nilai dari masing-masing *assessment*, data kerabat karyawan, data keluarga dan juga data anak karyawan tersebut. Kompas Gramedia menyimpan informasi-informasi personal seluruh karyawannya di *database*.

#### 2.2 Nilai Penilaian Kinerja

Nilai Penilaian Kinerja atau lebih sering disebut Nilai PK adalah penilaian atas kinerja karyawan yang diberikan setiap rentang waktu tertentu. Saat ini kategori Penilaian Kinerja yang ada di Kompas Gramedia adalah sebagai berikut (Cahyadi, 2017).

1. Outstanding
2. Exceed Expectation
3. Meet Expectation
4. Need Improvement
5. Unacceptable

#### 2.3 Data Mining

Data *mining* adalah proses yang menggunakan teknik statistik, matematika, kecerdasan buatan, dan *machine learning* untuk mengekstraksi dan mengidentifikasi informasi yang bermanfaat dan pengetahuan yang terkait dari

berbagai *database* (Kusrini & Luthfi, 2009). Menurut Witten, Frank dan Hall (2011), *data mining* memecahkan masalah dengan menganalisis data yang telah ada di *database*.

Berikut adalah contoh dari metode-metode yang dapat digunakan dalam proses *data mining*.

1. *Classification*

Metode *classification* mengklasifikasikan *item* (data) ke dalam salah satu kelas dari kelas-kelas yang telah ditentukan sebelumnya (Weiss & Kulikowski, 1991).

2. *Regression*

Metode *regression* memetakan *item* (data) ke dalam *variable* prediksi yang bersifat nyata, contoh penggunaannya adalah untuk mengestimasi kemungkinan seorang pasien akan bertahan setelah diberikan penanganan medis (Fayyad, Piatetsky-Shapiro, & Smyth, 1996).

3. *Clustering*

Metode *clustering* digunakan untuk mengidentifikasi kumpulan kategori atau kelompok yang terbatas untuk menggambarkan data (Jain & Dubes).

4. *Summarization*

Metode *summarization* digunakan untuk mencari deskripsi yang padat dari sebuah *subset* data (Fayyad, Piatetsky-Shapiro, & Smyth, 1996).

5. *Association*

Metode asosiasi adalah sebuah fungsi untuk menemukan probabilitas terjadinya atau adanya suatu item dalam suatu *subset* data (Oracle, n.d.).

## 6. *Dependency Modelling*

*Dependency modelling* mencari sebuah model yang menggambarkan ketergantungan variabel secara signifikan (Fayyad, Piatetsky-Shapiro, & Smyth, 1996).

### 2.4 **Klasifikasi**

Klasifikasi adalah salah satu bentuk dari teknik atau metode data *mining* yang termasuk dalam kategori *predictive mining* yaitu suatu teknik yang dapat digunakan untuk meramalkan atau memprediksi kecenderungan data di masa depan (Fayyad, 1996).

Klasifikasi masuk ke dalam *supervised induction*, dimana pengujian yang memanfaatkan kumpulan pengujian dari *record* dan atribut yang terklasifikasi untuk menentukan *output* dan kelas tambahan. Salah satu contoh algoritmanya adalah *decision tree* yang terkenal dan mudah dalam implementasinya ke dalam bentuk grafik (Kumara & Supriyanto, 2015).

Komponen-komponen utama dari proses klasifikasi adalah sebagai berikut (Kumara & Supriyanto, 2015).

1. Kelas, merupakan variabel tidak bebas yang merupakan label dari hasil klasifikasi.
2. Prediktor, merupakan variabel bebas suatu model berdasarkan dari karakteristik atribut data yang diklasifikasi.
3. Set data pelatihan, merupakan sekumpulan data lengkap yang berisi kelas dan predictor untuk dilatih agar model dapat mengelompokkan ke dalam kelas yang tepat. Contohnya adalah grup pasien yang telah di-*test* terhadap serangan jantung, grup pelanggan di suatu supermarket, dan sebagainya.

4. Set data uji, berisi data-data baru yang akan dikelompokkan oleh model guna mengetahui akurasi dari model yang telah dibuat.

## 2.5 Algoritma C4.5

Pada akhir tahun 1970 sampai awal tahun 1980 J. Ross Quinlan, seorang peneliti di bidang *machine learning*, membuat sebuah algoritma *decision tree* yang dikenal dengan ID3 (Iterative Dichotomiser). Quinlan kemudian mengembangkan algoritma ID3 mejadi algoritma C4.5 yang merupakan penyempurnaan algoritma sebelumnya (Witten, Frank, & Hall, 2011).

Algoritma C4.5 memiliki kelebihan yaitu mudah dimengerti, fleksibel, dan menarik karena dapat divisualisasikan dalam bentuk gambar yaitu pohon keputusan (Sugiat, 2013). Pohon keputusan merupakan salah satu bentuk penggambaran model klasifikasi. Pada pohon keputusan, simpul dalam menyatakan pengujian terhadap suatu atribut, cabang menyatakan hasil dari suatu pengujian, sementara daun menyatakan kelas yang diprediksi (Mabrur, 2011).

Secara umum langkah algoritma C4.5 untuk membangun pohon keputusan adalah sebagai berikut (Mabrur, 2011).

1. Memilih atribut paling atas sebagai akar (*root*)
2. Memilih cabang untuk setiap nilai
3. Bagi kasus dalam cabang
4. Ulangi proses untuk masing-masing cabang sampai semua kasus pada cabang memiliki kelas yang sama.

Untuk memilih atribut sebagai akar, didasarkan pada nilai *gain* tertinggi dari atribut-atribut yang ada. Untuk menghitung *gain* digunakan Rumus 2.1.

$$Gain(S, A) = Entropy(S) - \sum_{i=1}^n \left( \frac{|S_i|}{|S|} \right) \times Entropy(S_i) \quad \dots(2.1)$$

Dengan

S : Himpunan kasus

A : Atribut

n : Jumlah partisi A

|S<sub>i</sub>| : Jumlah kasus pada partisi ke-i

|S| : Jumlah kasus dalam S

Sedangkan untuk menghitung *entropy* dapat digunakan Rumus 2.2.

$$Entropy(S) = \sum_{i=1}^n - p_i \times \log_2 p_i \quad \dots(2.2)$$

Dengan

S : Himpunan (dataset) kasus

n : Banyaknya partisi S

$p_i$  : Probabilitas yang didapat dari Sum(Ya) atau Sum(Tidak) dibagi n kasus

Untuk memudahkan penjelasan mengenai algoritma C4.5 berikut ini disertakan contoh kasus yang dituangkan dalam Tabel 2.1 (Mabrur, 2011).

Tabel 2.1 Keputusan Bermain Tennis

NO	OUTLOOK	TEMPERATURE	HUMIDITY	WINDY	PLAY
1	Sunny	Hot	High	FALSE	No
2	Sunny	Hot	High	TRUE	No
3	Cloudy	Hot	High	FALSE	Yes
4	Rainy	Mild	High	FALSE	Yes
5	Rainy	Cool	Normal	FALSE	Yes
6	Rainy	Cool	Normal	TRUE	Yes
7	Cloudy	Cool	Normal	TRUE	Yes
8	Sunny	Mild	High	FALSE	No
9	Sunny	Cool	Normal	FALSE	Yes
10	Rainy	Mild	Normal	FALSE	Yes
11	Sunny	Mild	Normal	TRUE	Yes
12	Cloudy	Mild	High	TRUE	Yes
13	Cloudy	Hot	Normal	FALSE	Yes
14	Rainy	Mild	High	TRUE	No

Dalam kasus yang tertera pada Tabel 2.1, akan dibuat pohon keputusan untuk menentukan main tenis atau tidak dengan melihat keadaan cuaca, temperatur, kelembaban dan keadaan angin. Adapun langkah-langkahnya sebagai berikut.

1. Menghitung jumlah kasus, jumlah kasus untuk keputusan Yes, jumlah kasus untuk keputusan No, dan *entropy* dari semua kasus dan kasus yang dibagi berdasarkan atribut OUTLOOK, TEMPERATURE, HUMIDITY dan WINDY. Setelah itu lakukan penghitungan *gain* untuk masing-masing atribut. Hasil perhitungan ditunjukkan oleh Tabel 2.2.

Tabel 2.2 Tabel Perhitungan Node 1

Node		Jumlah Kasus (S)	Tidak (S <sub>1</sub> )	Ya (S <sub>2</sub> )	Entropy	Gain
1	TOTAL	14	4	10	0.863120569	
	OUTLOOK					0.258521037
		CLOUDY	4	0	4	
		RAINY	5	1	4	0.721928095
		SUNNY	5	3	2	0.970950594
	TEMPERATURE					0.183850925
		COOL	4	0	4	0
		HOT	4	2	2	1
		MILD	6	2	4	0.918295834
	HUMIDITY					0.370506501
		HIGH	7	4	3	0.985228136
		NORMAL	7	0	7	0
	WINDY					0.005977711
		FALSE	8	2	6	0.811278124
		TRUE	6	4	2	0.918295834

Baris TOTAL kolom *entropy* pada Tabel 2.2 dihitung dengan Rumus 2.2, sebagai berikut.

$$Entropy(Total) = \left(-\frac{4}{14} * \log_2\left(\frac{4}{14}\right)\right) + \left(-\frac{10}{14} * \log_2\left(\frac{10}{14}\right)\right)$$

$$Entropy(Total) = 0.863120569$$

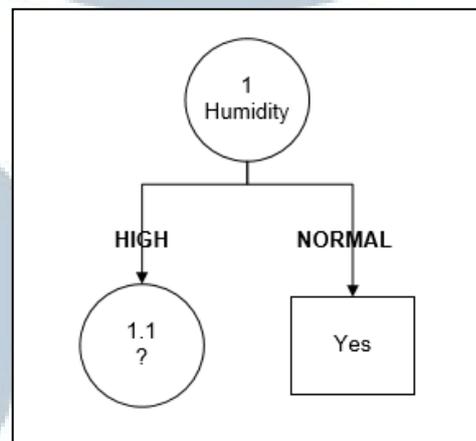
Sementara itu nilai *gain* pada baris OUTLOOK dihitung dengan menggunakan Rumus 2.1 sebagai berikut.

$$Gain(Total, Outlook) = Entropy(Total) - \sum_{i=1}^n \left( \frac{|Outlook_i|}{|Total|} \right) \times Entropy(Outlook_i)$$

$$Gain(Total, Outlook) = 0.863120569 - \left( \left( \frac{4}{14} * 0 \right) + \left( \frac{5}{14} * 0.723 \right) + \left( \frac{5}{14} * 0.97 \right) \right)$$

$$Gain(Total, Outlook) = 0.23$$

Dari hasil pada Tabel 2.2 dapat diketahui bahwa atribut dengan *gain* tertinggi adalah HUMIDITY yaitu sebesar 0.37. Dengan demikian HUMIDITY dapat menjadi node akar. Ada 2 nilai atribut dari HUMIDITY yaitu HIGH dan NORMAL. Dari kedua nilai atribut tersebut, nilai atribut NORMAL sudah mengklasifikasikan kasus menjadi satu yaitu keputusannya Ya, sehingga tidak perlu dilakukan perhitungan lebih lanjut, tetapi untuk nilai atribut HIGH masih perlu dilakukan perhitungan lagi. Dari hasil tersebut dapat digambarkan pohon keputusan sementara seperti tampak pada Gambar 2.1.



Gambar 2.1 Pohon Keputusan Hasil Perhitungan Node 1

2. Menghitung jumlah kasus, jumlah kasus untuk keputusan Ya, jumlah kasus untuk keputusan Tidak, dan *entropy* dari semua kasus dan kasus yang dibagi berdasarkan atribut OUTLOOK, TEMPERATURE dan WINDY yang dapat menjadi node akar dari nilai atribut HIGH. Setelah itu lakukan penghitungan

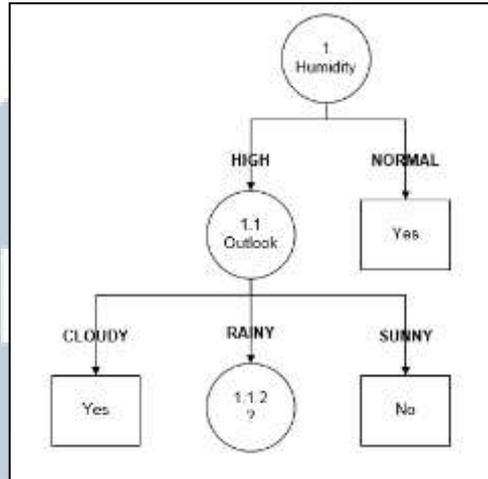
*gain* untuk masing-masing atribut. Hasil perhitungannya ditunjukkan oleh Tabel 2.3.

Tabel 2.3 Tabel Perhitungan Node 1.1

NODE			Jumlah Kasus (S)	Tidak (S1)	Ya (S2)	Entropy	Gain
	HUMIDITY HIGH		7	4	3	0.985228136	
	OUTLOOK						0.69951385
		CLOUDY	2	0	2	0	
		RAINY	2	1	1	1	
		SUNNY	3	3	0	0	
	TEMPERATURE						0.020244207
		COOL	0	0	0	0	
		HOT	3	2	1	0.918295834	
		MILD	4	2	2	1	
	WINDY						0.020244207
		FALSE	4	2	2	1	
		TRUE	3	2	1	0.918295834	

Dari hasil pada Tabel 2.3 dapat diketahui bahwa atribut dengan *gain* tertinggi adalah OUTLOOK yaitu sebesar 0.67. Dengan demikian OUTLOOK dapat menjadi *node* cabang dari nilai atribut HIGH. Ada 3 nilai atribut dari OUTLOOK yaitu CLOUDY, RAINY dan SUNNY. Dari ketiga nilai atribut tersebut, nilai atribut CLOUDY sudah mengklasifikasikan kasus menjadi yaitu keputusan Ya dan nilai atribut SUNNY sudah mengklasifikasikan kasus menjadi satu dengan keputusan Tidak, sehingga tidak perlu dilakukan perhitungan lebih lanjut, tetapi untuk nilai atribut RAINY masih perlu dilakukan perhitungan lagi. Pohon keputusan yang terbentuk sampai tahap ini dapat dilihat pada Gambar 2.2.

UNIVERSITAS  
MULTIMEDIA  
NUSANTARA

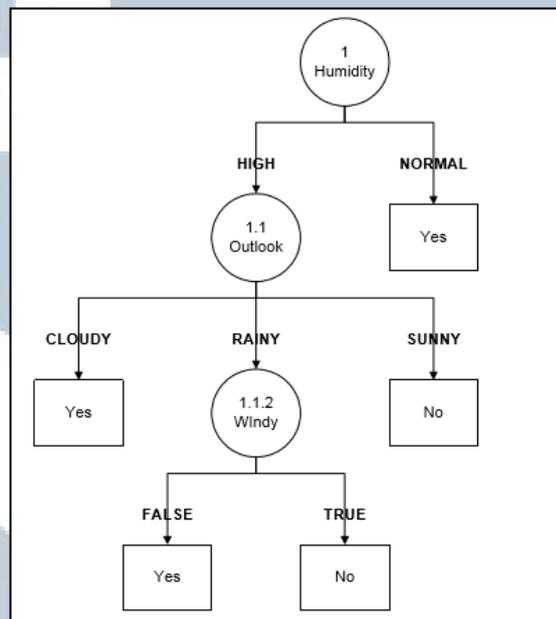


Gambar 2.2 Pohon Keputusan Hasil Perhitungan Node 1.1

3. Menghitung jumlah kasus, jumlah kasus untuk keputusan Ya, jumlah kasus untuk keputusan Tidak, dan *entropy* dari semua kasus dan kasus yang dibagi berdasarkan atribut TEMPERATURE dan WINDY yang dapat menjadi node cabang dari nilai atribut RAINY. Setelah itu melakukan penghitungan *gain* untuk masing-masing atribut. Hasil perhitungan ditunjukkan oleh Tabel 4. Dari hasil pada Tabel 4 dapat diketahui bahwa atribut dengan *gain* tertinggi adalah WINDY yaitu sebesar 1. Dengan demikian WINDY dapat menjadi *node* cabang dari nilai atribut RAINY. Ada 2 nilai atribut dari WINDY yaitu FALSE dan TRUE. Dari kedua nilai atribut tersebut, nilai atribut FALSE sudah mengklasifikasikan kasus menjadi 1 yaitu keputusannya Ya dan nilai atribut TRUE sudah mengklasifikasikan kasus menjadi satu dengan keputusan Tidak, sehingga tidak perlu dilakukan perhitungan lebih lanjut untuk nilai atribut ini. Pohon keputusan yang terbentuk sampai tahap ini dapat dilihat pada Gambar 2.3. Dengan memperhatikan pohon keputusan pada Gambar 2.3, diketahui bahwa semua kasus sudah masuk dalam kelas. Dengan demikian, pohon keputusan pada Gambar 2.3 merupakan pohon keputusan terakhir yang terbentuk.

Tabel 2.4 Perhitungan Node 1.1.2

NODE		Jml Kasus (S)	Tidak (S1)	Ya (S2)	Entropy	Gain
1.1.2	HUMIDITY-HIGH dan OUTLOOK-RAINY	2	1	1	1	
	TEMPERATURE					0
	COOL	0	0	0	0	
	HOT	0	0	0	0	
	MILD	2	1	1	1	
	WINDY					1
	FALSE	1	0	1	0	
	TRUE	1	1	0	0	



Gambar 2.3 Pohon Keputusan Hasil Perhitungan Node 1.1.2

## 2.6 Pengukuran Hasil

Untuk mengevaluasi performa sebuah model yang dibangun oleh algoritma klasifikasi dapat dilakukan dengan menghitung akurasi dengan persamaan sebagai berikut (Muhammad, 2011).

$$Accuracy = \frac{\text{jumlah data benar}}{\text{banyak data}} \times 100\% \quad \dots(2.3)$$

## 2.7 Technology Acceptance Model (TAM)

Technology Acceptance Model (TAM) pertama kali dikembangkan oleh Davis pada tahun 1986 yang merupakan pengembangan dari teori perilaku berencana (Theory of Reasoned Action/TRA) oleh Ajzen dan Fishbein (Ilham, 2017). Menurut Ilham (2017), TAM merupakan model sederhana yang sangat valid dalam menjelaskan model perilaku penerimaan teknologi. Teori TAM berargumentasi bahwa penerimaan individu terhadap sistem teknologi informasi ditentukan oleh dua konstruk yakni persepsi kegunaan (*perceived usefulness*) dan persepsi kemudahan penggunaan (*perceived ease of use*).

Davis (1989) mendefinisikan persepsi mengenai kegunaan (*usefulness*) berdasarkan kata *useful* yang artinya *capable of being used advantageously* atau dapat digunakan untuk tujuan yang menguntungkan. Persepsi mengenai kemudahan (*ease of use*) adalah tingkat dimana seseorang percaya bahwa dengan menggunakan sistem tertentu tidak memerlukan usaha apapun (Davis, 1989). Dari definisinya diketahui bahwa persepsi kemudahan penggunaan merupakan suatu kepercayaan (*belief*) terhadap kemampuan seseorang dalam menggunakan atau mengoperasikan suatu sistem teknologi (Ilham, 2017).

Pada awalnya terdapat 14 poin untuk *perceived usefulness* seperti pada Gambar 2.4 dan 14 poin untuk *perceived ease of use* seperti yang digambarkan pada Gambar 2.5. Setelah itu, Davis melakukan wawancara dengan 15 orang pengguna komputer berpengalaman untuk menentukan apakah poin-poin tersebut sesuai untuk mengukur *perceive ease of use* dan *perceive usefulness*. Hasil wawancara tersebut menghasilkan 10 poin pada masing-masing persepsi seperti yang digambarkan pada Gambar 2.6.

Old Item #	Item	Rank	New Item #	Cluster
1	Job Difficult Without	13		C
2	Control Over Work	9	2	
3	Job Performance	2	6	A
4	Addresses My Needs	12		C
5	Saves Me Time	11		B
6	Work More Quickly	7	3	B
7	Critical to My Job	5	4	C
8	Accomplish More Work	6	7	B
9	Cut Unproductive Time	10		B
10	Effectiveness	1	8	A
11	Quality of Work	3	1	A
12	Increase Productivity	4	5	B
13	Makes Job Easier	8	9	C
14	Useful	NA	10	NA

Gambar 2.4 Pretest Results: Perceived Usefulness (Davis, 1989)

Old Item #	Item	Rank	New Item #	Cluster
1	Confusing	7		B
2	Error Prone	13		
3	Frustrating	3	3	B
4	Dependence on Manual	9	(replace)	C
5	Mental Effort	5	7	B
6	Error Recovery	10		
7	Rigid & Inflexible	6	5	A
8	Controllable	1	4	A
9	Unexpected Behavior	11		
10	Cumbersome	2	1	A
11	Understandable	4	8	B
12	Ease of Remembering	8	6	C
13	Provides Guidance	12	(replace)	C
14	Easy to Use	NA	10	NA
NA	Ease of Learning	NA	2	NA
NA	Effort to Become Skillful	NA	9	NA

Gambar 2.5 Pretest Results: Perceived Ease of Use (Davis, 1989)

Setelah mendapatkan 10 poin pada masing-masing persepsi, Davis melakukan tes *reliability* dan *validity* terhadap 10 poin tersebut dengan melakukan studi lapangan kepada 112 karyawan IBM Toronto, Kanada. Dari hasil studi lapangan tersebut didapatkan 6 poin untuk setiap persepsi seperti pada Gambar 2.7.

U N I V E R S I T A S  
M U L T I M E D I A  
N U S A N T A R A

Scale Items	Factor 1 (Usefulness)	Factor 1 (Ease of Use)
<b>Usefulness</b>		
1 Quality of Work	.80	.10
2 Control over Work	.86	-.03
3 Work More Quickly	.79	.17
4 Critical to My Job	.87	-.11
5 Increase Productivity	.87	-.10
6 Job Performance	.93	-.07
7 Accomplish More Work	.91	-.02
8 Effectiveness	.96	-.03
9 Makes Job Easier	.80	-.16
10 Useful	.74	.23
<b>Ease of Use</b>		
1 Cbersome	.00	.73
2 Ease of Learning	.08	.60
3 Frustrating	.02	.65
4 Controllable	.13	.74
5 Rigid & Inflexible	.09	.54
6 Ease of Remembering	-.17	.62
7 Mental Effort	-.07	.76
8 Understandable	.29	.64
9 Effort to Be Skillful	-.25	.88
10 Easy to Use	.23	.72

Gambar 2.6 Factor Analysis of Perceived Usefulness and Ease of Use Questions: Study 1 (Davis, 1989)

Scale Items	Factor 1 (Usefulness)	Factor 2 (Ease of Use)
<b>Usefulness</b>		
1 Work More Quickly	.91	.01
2 Job Performance	.98	-.03
3 Increase Productivity	.98	-.03
4 Effectiveness	.94	.04
5 Makes Job Easier	.95	-.01
6 Useful	.88	.11
<b>Ease of Use</b>		
1 Easy to Learn	-.20	.97
2 Controllable	.19	.83
3 Clear & Understandable	-.04	.89
4 Flexible	.13	.63
5 Easy to Become Skillful	.07	.91
6 Easy to Use	.09	.91

Gambar 2.7 Factor Analysis of Perceived Usefulness and Ease of Use Items: Study 2 (Davis, 1989)

## 2.8 Skala Likert

Skala Likert adalah skala pengukuran yang dikembangkan oleh Likert (1932) yang mempunyai empat atau lebih butir-butir pertanyaan yang dikombinasikan sehingga membentuk sebuah skor/nilai yang merepresentasikan sifat individu misalnya pengetahuan, sikap, dan perilaku. Menurut Darmadi (2011), rumus yang digunakan untuk menghitung total nilai Skala Likert adalah sebagai berikut.

$$t = T \times Pn \dots(2.4)$$

Dengan

t : Total Nilai Skala Likert

T : Total Responden

Pn : Nilai Skala Likert

Skala Likert dikembangkan menggunakan 5 titik respon yaitu sangat setuju, setuju, tidak memutuskan atau netral, tidak setuju, dan sangat tidak setuju (Likert, 1932). Pada Skala Likert Lima Tingkat, nilai Skala Likert tertinggi adalah 5, sedangkan nilai Skala Likert terendah adalah 1 dengan interval persentase sesuai dengan Tabel 2.5.

Tabel 2.5 Interval Persentase Skala Likert

No	Keterangan	Interval
1	Sangat Tidak Setuju	0% - 19,99%
2	Tidak Setuju	20% - 39,99%
3	Netral	40% - 59,99%
4	Setuju	60% - 79,99%
5	Sangat Setuju	80% - 100%

Untuk menghitung persentase nilai Skala Likert, rumus intepretasinya adalah sebagai berikut.

$$P = \frac{t}{Y} \times 100\% \quad \dots(2.5)$$

Dengan

P : Persentase nilai

t : Total Nilai Skala Likert

Y : Total Nilai Tertinggi Skala Likert