

BAB II

LANDASAN TEORI

2.1 State of The Art

Penelitian prakiraan cuaca telah beberapa kali dilakukan dengan metode yang berbeda-beda, seperti logika fuzzy Sugeno dan Mamdani, *naïve bayes* dan C.45. Peneliti yang menggunakan logika fuzzy sugeno dan mamdani mendapatkan hasil akurasi terbesar 63% dan 80% (Harmoko & AZ), sedangkan penelitian yang menggunakan metode *naïve bayes* mendapatkan akurasi sebesar 82,65% (Subhan & Fanani, 2017) dan penelitian yang menggunakan algoritma C4.5 menghasilkan akurasi sebesar 88,89% (Novandya & Oktaria, 2017). Berdasarkan deskripsi hasil penelitian sebelumnya, algoritma C4.5 dan *naïve bayes* menghasilkan nilai akurasi yang tinggi yakni 88,89% dan 82,65%. Literature review terkait prakiraan cuaca dan metode optimasi secara rinci dapat dilihat pada lampiran 6.

Tabel 2.1 menunjukan metode *Naïve Bayes* bayes menggunakan 10 parameter input dan dua parameter output (Subhan & Fanani, 2017), sementara algoritma C4.5 menggunakan 10 parameter input dan 6 parameter output (Novandya & Oktaria, 2017). Berdasarkan *study literature* penelitian yang menggunakan metode *naïve bayes* melakukan menguji 183 data sehingga menyarankan penelitian lainnya untuk menambah *dataset*, sedangkan penelitian yang menggunakan metode C4.5 menyarankan penelitian selanjutnya menggunakan metode optimasi untuk meningkatkan tingkat akurasi.

Penelitian ini akan mengembangkan penelitian yang dilakukan kedua penelitian tersebut dengan menambah dataset menjadi 10 tahun, serta melakukan optimasi menggunakan metode *ensemble*.

Metode *ensemble* yang digunakan merupakan *Bagging* dan *Boosting*. Tabel 2.2 memaparkan penelitian terkait optimasi menggunakan metode *ensemble* pada algoritma *naïve bayes* dan C4.5. Berdasarkan tabel tersebut kenaikan tertinggi terjadi pada penelitian “Penerapan Dizcretization dan Teknik Bagging Untuk Meningkatkan Akurasi Klasifikasi Berbasis Ensemble pada Algoritma C4.5 dalam Mendiagnosa Diabete” yang dilakukan oleh Mirqotussa’adaha dkk. Penelitian tersebut melakukan optimasi algoritma C4.5 dalam mendiagnosa diabete menggunakan *bagging*, hasil penelitian tersebut menunjukkan adanya kenaikan akurasi sebesar 5% (‘Adaha, Muslima, Sugiharti, Prasetyo, & Alimah, 2017).

Tabel 2.1 Matrix Penelitian Terkait Prakiraan Cuaca

NO	Judul, Penulis	Ruang Lingkup Penelitian		
		Metode	Input	Output
1.	APLIKASI PENDUKUNG KEPUTUSAN DENGAN LOGIKA FUZZY (STUDY KASUS: PRAKIRAAN CUACA DI BMKG)	Logika Fuzzy	1. Kelembaban udara 2. Tekanan udara 3. Suhu udara	1. Hujan 2. Cerah 3. Cerah Berawan 4. Hujan Berawan

Tabel 2.1 Matrix Penelitian Terkait Prakiraan Cuaca (Lanjutan 1)

NO	Judul, Penulis	Ruang Lingkup Penelitian		
		Metode	Input	Output
	JAMBI , Lucy Simorangkir, Muchammad Nur (2013)			5. Berawan
2	PEMANFATAN DATA MINING UNTUK PRAKIRAAN CUACA , Subekti Mujiasih (2016)	<i>Association Rule, C4.5, Classification dan Random Forest</i>	Data Sinoptik 0 dan 1	1. Hujan 2. Tidak Hujan
3	Prototipe Model Prediksi Peluang Kejadian Hujan Menggunakan Metode Fuzzy Logic Tipe Mamdani dan Sugeno , Iis Widya Harmoko , Nazori AZ (2016)	Metode Fuzzy Logic Tipe Mamdani dan Sugeno	1. SOI 2. SST 3. MJO 4. Li 5. RH	1. Cerah-Berawan 2. Berawan-Ringan 3. Hujan Ringan-Sedang 4. Hujan Sedang-Lebat 6. Hujan Sedang-Lebat

Tabel 2.1 Matrix Penelitian Terkait Prakiraan Cuaca (Lanjutan 2)

NO	Judul, Penulis	Ruang Lingkup Penelitian		
		Metode	Input	Output
4.	METODE SIKLIS DAN ADAPTIVE NEURO FUZZY INFERENCE SYSTEM UNTUK PERAMALAN CUACA. Fahrur Rozi, Farid Sukmana (2016)	<i>Adaptive Neuro Fuzzy Inference System</i>	1. Suhu 2. Tekanan udara 3. Kelembaban udara Kecepatan angin	1. Cerah 2. Mendung 3. Hujan
5.	PERANCANGAN SISTEM PERAMALAN CUACA BERBASIS LOGIKA FUZZY, Ema Sastri Puspita, Liza Yulianti (2016)	Logika Fuzzy Sugeno	1. Suhu 2. Kelembaban 3. Angin	1. Cerah 2. Mendung 3. Hujan
6.	PROTOTIPE SISTEM PRAKIRAAN CUACA BERDASARKAN SUHU DAN KELEMBAPAN DENGAN METODE LOGIKA FUZZY DAN BACKPROPAGATION BERBASIS MIKROKONTROLER, Ratna Aisuwarya, Dodon Yendri, Werman Kasoep, Kiki Amelia, Adi Arga Arifnur (2016)	Logika Fuzzy dan Backpropagation	1. Suhu 2. Titik embun	1. Hujan lebat 2. Gerimis 3. Berawan 4. Panas 5. Sangat panas

Tabel 2.1 Matrix Penelitian Terkait Prakiraan Cuaca (Lanjutan 3)

NO	Judul, Penulis	Ruang Lingkup Penelitian		
		Metode	Input	Output
7.	IMPLEMENTASI BACKPROPAGATION NEURAL NETWORK DALAM PRAKIRAAN CUACA DI DAERAH BALI SELATAN , I Made Dwi Udayana Putra, G. K. Gandhiadi, Luh Putu Ida Harini (2016)	<i>Backpropagation Neural Network</i>	1. Suhu udara 2. Titik embun 3. Kecepatan angin 4. Tekanan udara 5. Jarak pandang	1. Cerah 2. Cerah berawan 3. Cerah berawan cumulonimbus 4. Berawan 5. Berawan dengan cumulonimbus 6. Badai Guntur 7. Hujan ringan 8. Hujan ringan cumulonimbus 9. Hujan sedang 10. Hujan sedang cumulonimbus 11. Hujan lebat 12. Hujan lebar cumulonimbus

Tabel 2.1 Matrix Penelitian Terkait Prakiraan Cuaca (Lanjutan 4)

NO	Judul, Penulis	Ruang Lingkup Penelitian		
		Metode	Input	Output
				13. Hujan ringan dengan Guntur 14. Hujan sedang dengan badai dan Guntur 15. Hujan lebat dengan badai dan Guntur.
8.	Analysis of Data Mining Techniques for Weather Prediction , Fahad Sheikh, S. Karthick, D. Malathi, J. S. Sudarsan and C. Arun (2016)	<i>Naïve bayes and C4.5 Decision Tree</i>	1. <i>Max temperature</i> 2. <i>Min temperature</i> 3. <i>Mean temperature</i> 4. <i>Minop humidity</i> 5. <i>Mean humidity</i> 6. <i>Wind speed</i> 7. <i>Cloud</i> Rainfall	1. Hujan 2. Tidak Hujan

Tabel 2.1 Matrix Penelitian Terkait Prakiraan Cuaca (Lanjutan 5)

NO	Judul, Penulis	Ruang Lingkup Penelitian		
		Metode	Input	Output
9.	PENERAPAN DATA MINING UNTUK PRAKIRAAN CUACA DI KOTA MALANG MENGGUNAKAN ALGORITMA ITERATIVE DICHOTOMISER TREE (ID3) , Broto Poernomo T.P.1), Rina Dewi Indah Sari (2017)	Algoritma ID3	<ol style="list-style-type: none"> 1. Suhu minimal 2. Suhu maksimal 3. Suhu rata-rata 4. Kelembanan 	<ol style="list-style-type: none"> 1. Tidak hujan 2. Hujan ringan 3. Hujan sedang 4. Hujan lebat 5. Hujan sangat lebat
10.	PENERAPAN LOGIKA FUZZY DALAM SISTEM PRAKIRAAN CUACA BERBASIS MIKROKONTROLER , Velma Kurniati, Dedi Triyanto, Tedy Rismawan (2017)	Logika Fuzzy	<ol style="list-style-type: none"> 1. Suhu 2. Kelembaban 3. Tekanan udara 	<ol style="list-style-type: none"> 1. Hujan 2. Cerah 3. Cerah Berawan
11.	Penerapan Algoritma Klasifikasi Data Mining C4.5 pada Dataset Cuaca Wilayah Bekasi , Adhika Novandya, Isni Oktria (2017)	Algoritma C4.5	<ol style="list-style-type: none"> 1. Date 2. Time 3. Desc 4. Temp 	<ol style="list-style-type: none"> 1. Clear 2. Sunny 3. Partly cloudy 4. Patchy rain nearby

Tabel 2.1 Matrix Penelitian Terkait Prakiraan Cuaca (Lanjutan 6)

NO	Judul, Penulis	Ruang Lingkup Penelitian		
		Metode	Input	Output
			5. Rain 6. Wind 7. Dir 8. Could 9. Humidity 10. Pressure	5. Heavy rain shower 6. Light rain shower
12.	<i>Weather Prediction Based on Fuzzy Logic Algorithm for Supporting General Farming Automation System</i> , Aris Pujud Kurniawan. Agung Nugroho Jati, Fairuz Azmi (2017)	Logika Fuzzy	1. Soil 2. Raindrops	1. Clear 2. Cloudy 3. Rainy
13.	PENERAPAN DATA MINING UNTUK MENENTUKAN POTENSI HUJAN HARIAN DENGAN MENGGUNAKAN ALGORITMA NAIVE BAYES , Ahmad Subhan, Ahmad Zainul Fanani (2017)	<i>Naïve Bayes</i>	1. Temperature MAX 2. Temperature MIN 3. Temperature Mean 4. Hr. Med 5. Wind Max 6. Wind Mean 7. Pressure SLP	1. Hujan 2. Tidak Hujan

Tabel 2.1 Matrix Penelitian Terkait Prakiraan Cuaca (Lanjutan 7)

NO	Judul, Penulis	Ruang Lingkup Penelitian		
		Metode	Input	Output
			8. Pressure STN 9. Vis 10. Prec	
14.	<i>Rainfall Forecasting in Bandung Regency using C4.5 Algorithm</i> , Joko Azhari Suyatno, Fhira Nhita, Aniq Atiqi Rohmawati (2018)	Algoritma C4.5	1. Temperature 2. Rainfall 3. Solar Radiation 4. Humidity 5. Wind Velocity 6. Evaporation	1. No rainfall 2. Low rainfall 3. Medium rainfall 4. Heavy rainfall 5. Very heavy rainfall
15.	OPTIMASI PRAKIRAAN CUACA MENGGUNAKAN METODE ENSEMBLE PADA ALGORITMA NAÏVE BAYES DAN C4.5 , Vini Indri Yani (2020)	Algoritma Naïve Bayes dan C4.5	1. Suhu Minimum 2. Suhu Maksimum 3. Suhu Rata-rata 4. Kelembaban Rata-rata 5. Curah Hujan 6. Lama Penyinaran Matahari	1. Tidak Hujan 2. Hujan Sangat Ringan 3. Hujan Ringan 4. Hujan Sedang 5. Hujan Lebat 6. Hujan Sangat Lebat

Tabel 2.1 Matrix Penelitian Terkait Prakiraan Cuaca (Lanjutan 8)

NO	Judul, Penulis	Ruang Lingkup Penelitian		
		Metode	Input	Output
			7. Kecepatan Angin Maksimum 8. Kecepatan Maksimum Arah Angin 9. Kecepatan Angin Rata-rata	

Tabel 2.2 Matrix Penelitian Terkait Metode Optimasi

No.	Judul/Penulis	Metode	Metode Optimasi	Kenaikan Akurasi
1.	OPTIMIZATION OF BAGGING CLASSIFIERS BASED ON SBCB ALGORITHM , XIAO-DONG ZENG, SAM CHAO, FAI WONG (2010)	<i>Naïve Bayes</i>	<i>Bagging dan SBCB (Selecting Base Classifiers on Bagging)</i>	Naik 1-2%
2.	PENERAPAN TEKNIK BAGGING PADA ALGORITMA KLASIFIKASI UNTUK	Algoritma C.45	<i>Bagging</i>	Naik 1-2%

Tabel 2.2 Matrix Penelitian Terkait Metode Optimasi (Lanjutan 1)

No.	Judul/Penulis	Metode	Metode Optimasi	Kenaikan Akurasi
3.	Penerapan Dizcretization dan Teknik Bagging Untuk Meningkatkan Akurasi Klasifikasi Berbasis Ensemble pada Algoritma C4.5 dalam Mendiagnosa Diabetes , Mirqotussa'adaha, Much Aziz Muslima, Endang Sugiharti, Budi Prasetyo, Siti Alimah (2017)	Algoritma C4.5	<i>Bagging</i>	Naik sekitar 5%
4.	PENERAPAN METODE ENSEMBLE UNTUK MENINGKATKAN KINERJA ALGORITME KLASIFIKASI PADA IMBALANCED DATASET , Yoga Pristyanto (2019)	<i>Naïve Bayes</i> <i>Decision Tree</i>	<i>AdaBoost</i>	Naïve Bayes: Naik 1-2% Decision Tree: Naik sekitar 1%

2.2 Tinjauan Pustaka

2.2.1 Prakiraan Cuaca

Cuaca merupakan keadaan udara pada suatu waktu yang singkat dan tempat yang sempit. Kondisi cuaca di suatu tempat dapat ditentukan oleh beberapa factor seperti suhu udara, kelembaban udara, intensitas cahaya matahari, arah angin, kecepatan angin dan sebagainya. Faktor-faktor yang ada tersebut menjadikan cuaca keesokan harinya dapat diprediksi. Prakiraan atau prediksi adalah suatu kegiatan yang meramalkan suatu variabel dimasa mendatang yang berdasarkan pada data ataupun variabel pada masa lampau (T.P & Indah Sari, 2017) (Rozi & Sukmana, 2016). Berdasarkan penjabaran di atas prakiraan cuaca dapat diartikan proses prediksi keadaan cuaca dimasa mendatang menggunakan variabel atau faktor cuaca ada.

2.2.2 *Naïve Bayes*

Naïve Bayes adalah algoritma sederhana namun kuat untuk pemodelan prediktif (Brownlee, 2017). Algoritma *Naïve Bayes* memprediksi peluang di masa depan berdasarkan pengalaman di masa sebelumnya sehingga dikenal sebagai Teorema Bayes. Algoritma *Naïve Bayes* merupakan sebuah metoda klasifikasi menggunakan metode probabilitas dan statistik yg dikemukakan oleh ilmuwan Inggris Thomas Bayes (INFORMATIKALOGI, 2017). Teorema *Bayes* menyediakan cara untuk dapat menghitung probabilitas hipotesis yang diberikan pengetahuan kita sebelumnya (Brownlee, 2017).

Persamana Teorema *Naïve Bayes*

$$P(C|X) = \frac{P(C|X)P(C)}{P(X)} \quad (2.1)$$

- x = Data dengan class yang belum diketahui
 c = Hipotesis data merupakan suatu class spesifik
 $P(c/x)$ = Probabilitas hipotesis berdasar kondisi (posteriori probability)
 $P(c)$ = Probabilitas hipotesis (prior probability)
 $P(x/c)$ = Probabilitas berdasarkan kondisi pada hipotesis
 $P(x)$ = Probabilitas c

Penjabaran lebih lanjut dari rumus tersebut adalah $(C|X_1, \dots, X_n)$ dengan menggunakan aturan perkalian sebagai berikut ini:

$$P(C|X_1, \dots, X_n) = P(C)P(X_1, \dots, X_n|C) \quad (2.2)$$

$$= P(C)P(X_1|c)(X_2, \dots, X_n|C, X_1) \quad (2.3)$$

$$= P(C)P(X_1|c)P(X_2|C, X_1)(X_3, \dots, X_n|C, X_1, X_2) \quad (2.4)$$

$$= P(C)P(X_1|c)P(X_2|C, X_1)P(X_3|C, X_1, X_2) \dots P(X_n|C, X_1, X_2, \dots, X_{n-1}) \quad (2.5)$$

Hasil penjabaran tersebut menyebabkan semakin banyak X, maka akan semakin kompleks yang hampir mustahil untuk dianalisa satu persatu. Solusi yang dapat dilakukan adalah dengan menggunakan independensi yang sangat tinggi, yakni masing-masing parameter saling bebas satu sama lain (INFORMATIKALOGI, 2017). Berdasarkan asumsi tersebut, maka berlaku suatu persamaan sebagai berikut:

$$P(c|X_1, \dots, X_n) = P(C) \prod_{i=1}^n P(X_i|C) \quad (2.6)$$

$$P(c|X) = P(x_1|c)P(x_2|c) \dots P(x_n|c)P(c) \quad (2.7)$$

Persamaan-persamaan diatas merupakan model Teorema *Naïve Bayes* untuk dalam proses klasifikasi, sementara untuk klasifikasi dengan data *continue* menggunakan rumus *Densitas Gauss*:

$$P = (X_i = x_i | Y_i = y_i) = \frac{1}{\sqrt{2\pi} \sigma_{ij}} x e^{-\left(\frac{(x_i - u_{ij})^2}{2\sigma^2}\right)} \quad (2.8)$$

Keterangan:

P = Peluang

X_i = Atribut ke-i

x_i = Nilai atribut ke-i

Y_i = Kelas yang dicari

y_i = Sub kelas Y yang dicari

u = Mean, menyatakan rata-rata dari seluruh atribut

σ = Devisiasi standar, menyatakan varian dari seluruh atribut

Persamaan untuk *mean* dan devisiasi standar

$$mean(x) = \frac{1}{n} x \sum_{i=1}^n x_i \quad (2.9)$$

$$standar\ deviasi(x) = \sqrt{\frac{1}{n-1} x \sum_{i=1}^n (x_i - mean(x))^2} \quad (2.10)$$

Tahapan yang dilakukan dalam proses klasifikasi *naïve bayes* dengan data *continue* adalah

1. Menentukan *class probabilities*.
2. Menentukan *conditional probabilities*
3. Melakukan *prediction*
4. Melakukan *final prediction*

2.2.3 Algoritma C4.5

Algoritma C4.5 adalah algoritma yang mengubah data menjadi pohon keputusan dan aturan-aturan keputusan (*rules*) (Novandya & Oktaria, 2017).

Algoritma C4.5 merupakan algoritma pohon keputusan yang bisa mengatasi atribut kontinyu, sehingga menjadikan algoritma ini sebagai salah satu algoritma klasifikasi dan prediksi yang populer (Suyatno, Nhita, & Rohmawati, 2018). Ada beberapa tahapan dalam membangun sebuah pohon keputusan C4.5 yaitu:

- a) Menyiapkan dataset
- b) Menentukan akar pohon dengan menghitung nilai *gain* dari masing-masing atribut, nilai *gain* yang paling tinggi akan menjadi akar dari pohon keputusan. Namun, sebelum menghitung *gain* dari atribut, tentukan dahulu nilai entropi. Berikut persamaan untuk mencari *entropy* dan *gain*:

$$Entropy(S) = \sum_{i=1}^n -p_i \times \log_2 p_i \quad (2.11)$$

Di mana:

S = Himpunan kasus

n = Jumlah partisi S

p_i = Proporssi S_i terhadap S

$$Gain(S, A) = Entropy(S) - \sum_{i=1}^n \frac{|S_i|}{|S|} \times Entropy(S_i) \quad (2.12)$$

S = Himpunan kasus

A = Atribut

n = Jumlah partisi atribut A

$|S_i|$ = Jumlah kasus pada partisi ke i

$|S|$ = Jumlah dalam kasus S

Algoritma C4.5 merupakan pengembangan dari algoritma ID3. Oleh karena pengembangan tersebut algoritma C4.5 mempunyai prinsip dasar kerja yang sama dengan algoritma ID3. Namun, pada algoritma C4.5 pemilihan atribut dilakukan dengan menggunakan *gain ratio* (Informatikalogi, 2017). Berikut merupakan persamaan untuk *gain ratio*:

$$\text{gain ratio}(a) = \frac{\text{gain}(a)}{\text{split}(a)} \quad (2.13)$$

a = atribut

$\text{gain}(a)$ = information gain pada atribut a

$\text{split}(a)$ = split information pada atribut a

Atribut dengan nilai *gain ratio* tertinggi dipilih sebagai atribut test untuk simpul. Pendekatan ini merupakan normalisasi pada *information gain* dengan menggunakan apa yang disebut sebagai *split information*. *Split information* menyatakan *entropy* atau informasi potensial dengan rumus:

$$\text{SplitInfo}(S, A) = - \sum_{i=1}^n \frac{S_i}{S} \log_2 \frac{S_i}{S} \quad (2.14)$$

S = ruang (data) *sample* yang digunakan untuk *training*

A = Atribut

S_i = Jumlah sample untuk atribut i

- c) Lakukan Langkah b secara berulang hingga semua *record* terpartisi
- d) Proses partisi pohon keputusan akan berhenti saat:

- Semua record dalam simpul N mendapat kelas yang sama
- Tidak ada atribut di dalam *record* yang dipartisi lagi
- Tidak ada *record* di dalam cabang yang kosong.

2.2.4 Metode Ensemble

Ensemble method adalah algoritma *machine learning* yang kuat dan lebih maju. *Ensemble Method* merupakan teknik yang menggabungkan prediksi dari berbagai model untuk memberikan prediksi yang lebih akurat (Brownlee, 2017). Terdapat dua metode *ensemble* yang paling banyak digunakan yaitu *Bagging* dan *Random Forest* yang merupakan salah satu algoritma paling kuat yang ada. Kemudian, terdapat *Boosting* dan *AdaBoost* yang dapat memperbaiki prediksi yang lebih lemah.

a. *Bagging*

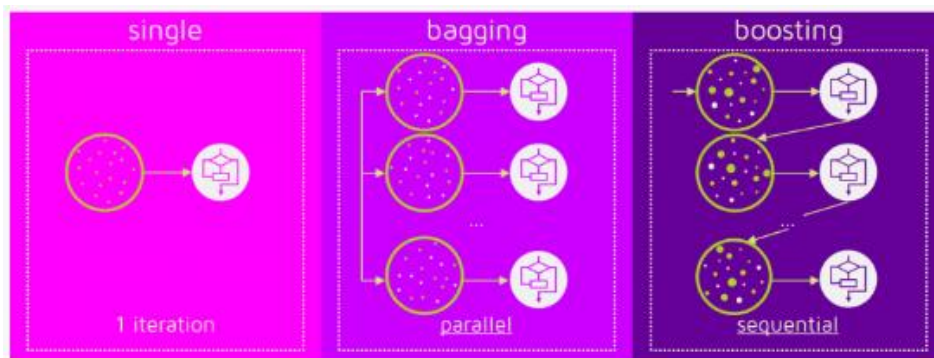
Bagging (*Agregasi Bootstrap*) merupakan metode ensemble yang sederhana dan sangat kuat. *Bagging* umumnya adalah prosedur umum yang dapat digunakan untuk mengurangi varians untuk algoritma yang memiliki varian tinggi (Brownlee, 2017). Seperti *decision tree*, klasifikasi dan *regression trees* (CART).

b. *Boosting*

Boosting adalah metode ensemble yang berupaya memperkuat klasifikasi dari metode klasifikasi yang lemah. *Boosting* merupakan metode yang membuat *classifier* kuat dari *classifier* yang lemah. Teknik ini dilakukan dengan membangun model dari data training, kemudian membuat yang kedua

model yang mencoba untuk memperbaiki kesalahan dari model pertama. Model ditambahkan hingga *training set* diprediksi dengan sempurna (Brownlee, 2017). Metode ini sering digunakan dalam mesin peningkat gradien stokastik. *Adaptive Boosting (adaboost)* merupakan salah satu dari beberapa varian pada algoritme boosting. Algoritme *AdaBoost* dari Freund dan Schapire (1995) merupakan algoritme penguat praktis pertama, dan tetap menjadi salah satu yang paling banyak digunakan dan dipelajari, dengan aplikasi di berbagai bidang (Pristyanto, 2019).

Perbedaan antara *bagging* dan boosting adalah *Bagging* memiliki tahap pelatihan bersifat *parallel*, artinya setiap model bersifat *independent*, sedangkan *boosting* bersifat *sequential*, setiap model dilatih dengan mempertimbangkan keberhasilan pengklasifikasi sebelumnya. (Quantdare, 2016).



Gambar 2.1 Perbedaan Tahap Pelatihan *Bagging* dan *Boosting*

Sumber: <https://quantdare.com/what-is-the-difference-between-bagging-and-boosting/>

Persamaan untuk bagging adalah

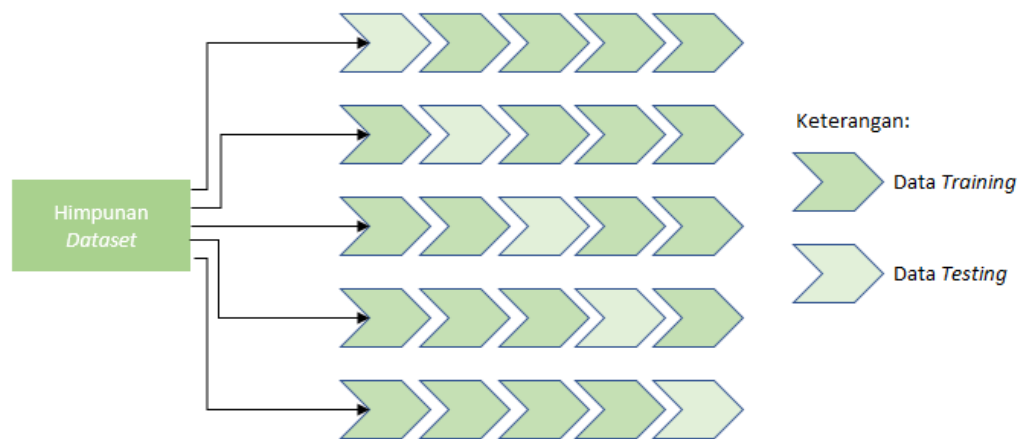
$$e = \frac{1}{N} \sum_{e=1}^N e_i \quad (2.15)$$

Persamaan untuk boosting adalah

$$e = \frac{1}{N} \sum_{e=1}^N w e_i \quad (2.16)$$

2.2.5 Metode Validasi

Penelitian ini menggunakan metode *K-Fold cross-validation* sebagai metode validasi. Metode ini melakukan pengtesan terhadap *dataset* sebanyak K .



Gambar 2.2 *K-Folds Cross Validation*

Gambar 2.2 merupakan contoh pengtesan dataset jika ‘ K ’ bernilai 5, maka dilakukan pengtesan sebanyak lima kali dengan pemilihan data *training* dan data *testing* secara acak.

2.2.6 Metode Evaluasi

Penelitian ini menggunakan *confusion matrix* sebagai metode evaluasi. *confusion matrix* merupakan metode yang mempresentasikan hasil evaluasi model dengan menggunakan tabel matriks. Jika *dataset* terdiri dari dua kelas, maka kelas pertama dianggap positif dan kelas kedua dianggap negatif (Subhan

& Fanani, 2017). Evaluasi menggunakan metode *confusion matrix* menghasilkan tiga luaran yakni akurasi, *precision* dan *recall*.

Tabel 2.3 *Confusion Matrix*

<i>Correct Classification</i>	<i>Classified as</i>	
	+	-
+	<i>True positives</i>	<i>False negatives</i>
-	<i>False positives</i>	<i>True negatives</i>

Tabel 2.3 menggambarkan bentuk dari *confusion matrix*. *True positive (TP)* pada table diatas merupakan jumlah *record* positif dalam *dataset* yang diklasifikasi positif. *False positive (FP)* adalah jumlah record negatif dalam *dataset* yang diklasifikasi positif. *True negative (TN)* merupakan jumlah *record* negative dalam dataset yang diklasifikasi positif. *False negative (FN)* adalah jumlah *record* positif yang diklasifikasi negatif (Subhan & Fanani, 2017).

Berikut merupakan persamaan model *confusion matrix*:

$$\text{Akurasi} = \frac{TP+TN}{TP+TN+FP+FN} \quad (2.17)$$

$$\text{Recall} = \frac{TP}{TP+FN} \quad (2.18)$$

$$\text{Presisi} = \frac{TP}{TP+FP} \quad (2.19)$$

Keterangan:

Akurasi adalah presentase jumlah *record* data yang diklasifikasi dengan benar oleh algoritma. *Recall* merupakan rasio prediksi benar positif dibandingkan dengan keseluruhan data yang benar positif. *Presisi* adalah rasio prediksi benar

positif dibandingkan dengan keseluruhan hasil yang diprediksi positif (Arthana, 2019).

2.2.7 RapidMiner

RapidMiner sebelumnya dikenal sebagai YALE (*Yet Another Learning Environment*), mulai dikembangkan pada tahun 2001 oleh Ralf Klinkenberg, Ingo Mierswa, dan Simon Fischer dari Unit Kecerdasan Buatan Universitas Teknik Dortmund (Deutsh, 2010).

RapidMiner merupakan perangkat lunak yang bersifat terbuka (open source). RapidMiner adalah sebuah solusi untuk melakukan analisis terhadap data mining, text mining dan analisis prediksi. RapidMiner menggunakan berbagai teknik deskriptif dan prediksi dalam memberikan wawasan kepada pengguna sehingga dapat membuat keputusan yang paling baik (C, Baskoro, Ambarwati, & Wicaksono, 2013).