

# Optimasi Prakiraan Cuaca Menggunakan Metode *Ensemble* pada Naïve Bayes dan C4.5

<http://dx.doi.org/10.28932/jutisi.v8i3.5455>

Riwayat Artikel

Received: 26 September 2022 | Final Revision: 30 November 2022 | Accepted: 30 November 2022

Creative Commons License 4.0 (CC BY – NC)



Vini Indri Yani<sup>✉ #1</sup>, Aradea<sup>\*2</sup>, Husni Mubarak<sup>#3</sup>

<sup>#</sup> Program studi Informatika, Universitas Siliwangi  
Jl. Siliwangi No. 24, Kota Tasikmalaya, 46115, Indonesia

<sup>1</sup>157006061@student.unsil.ac.id

<sup>3</sup> husni.mubarak@unsil.ac.id

<sup>\*</sup> Program studi Informatika, Universitas Siliwangi  
Jl. Siliwangi No. 24, Kota Tasikmalaya, 46115, Indonesia

<sup>2</sup>aradea@unsil.ac.id

<sup>✉</sup> Corresponding author: 157006061@student.unsi.ac.id

**Abstrak** — Peramalan cuaca merupakan hal yang penting bagi keberlangsungan hidup masyarakat luas. Oleh karena itu, akurasi dari peramalan cuaca haruslah tinggi. Berdasarkan hal itu maka dilakukan penelitian untuk meningkatkan akurasi peramalan cuaca dengan model *naïve Bayes* dan C4.5 kemudian dilakukan optimasi dengan metode *ensemble*. *Dataset* yang digunakan merupakan data cuaca hasil pengamatan dari BMKG Bandung selama 10 tahun. Akurasi pada proses *pretest* menunjukkan jika algoritma *naïve Bayes* memiliki akurasi sebesar yakni 49,45% dan algoritma C4.5 menghasilkan akurasi 41,24%, sementara pada proses *posttest* akurasi yang didapatkan adalah 49,76% untuk *bagging naïve Bayes*, 46,47% untuk *boosting naïve Bayes*, 45,76 untuk *bagging C4.5* dan 38,82% untuk C4.5.

**Kata kunci**— *Bagging; Boosting; C4.5; Ensemble; Naïve Bayes*.

## Optimization of Weather Forecast Using Ensemble Method on Naive Bayes and C4.5

**Abstract** — Weather forecasting is important for the survival of the wider community. Therefore, the accuracy of the weather forecast must be high. Based on this, a study was conducted to improve the accuracy of weather forecasting with the *naïve Bayes* and C4.5 models and then performed an optimization using the *ensemble* method. The dataset used is weather data observed from BMKG Bandung for 10 years. Accuracy in the *pretest* process shows that the *naïve Bayes* algorithm has an accuracy of 49.45% and the C4.5 algorithm produces 41.24% accuracy, while in the *posttest* process the accuracy obtained is 49.76% for *bagging naïve Bayes*, 46.47% for *boosting naïve Bayes*, 45.76 for *bagging C4.5* and 38.82% for C4.5.

**Keywords**— *Bagging; Boosting; C4.5; Ensemble; Naïve Bayes*.

## I. PENDAHULUAN

Penelitian prakiraan cuaca merupakan salah satu objek yang menarik untuk dikaji karena cuaca adalah salah satu hal yang berpengaruh dalam kehidupan sehari-hari sehingga akurasi yang baik pada prakiraan cuaca menjadi sangat dibutuhkan. Prakiraan cuaca dibuat dengan mengumpulkan data kuantitatif tentang kondisi atmosfer saat ini dan menggunakan pemahaman ilmiah tentang proses atmosfer untuk memproyeksikan bagaimana atmosfer akan berkembang [1]. 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% [2], sedangkan penelitian yang menggunakan metode *naïve Bayes* mendapatkan akurasi sebesar 82,65% [3] dan penelitian yang menggunakan algoritma C4.5 menghasilkan akurasi sebesar 88,89% [4]. Berdasarkan deskripsi hasil penelitian sebelumnya, algoritma C4.5 dan *naïve Bayes* menghasilkan nilai akurasi yang tinggi yakni 88,89% dan 82,65%.

Peramalan atau prakiraan yang bagus pada dasarnya harus memiliki nilai akurasi yang tinggi agar prediksi tersebut dapat digunakan, mendapatkan hasil yang baik dan sesuai dengan keadaan sebenarnya [5]. Perkembangan penelitian prakiraan cuaca selalu diharapkan memiliki akurasi yang lebih baik, cara-cara yang dapat dilakukan berupa menambah parameter input [6], menggabungkan beberapa metode [3], menggunakan metode yang belum pernah digunakan sebelumnya atau menggunakan metode optimasi [4].

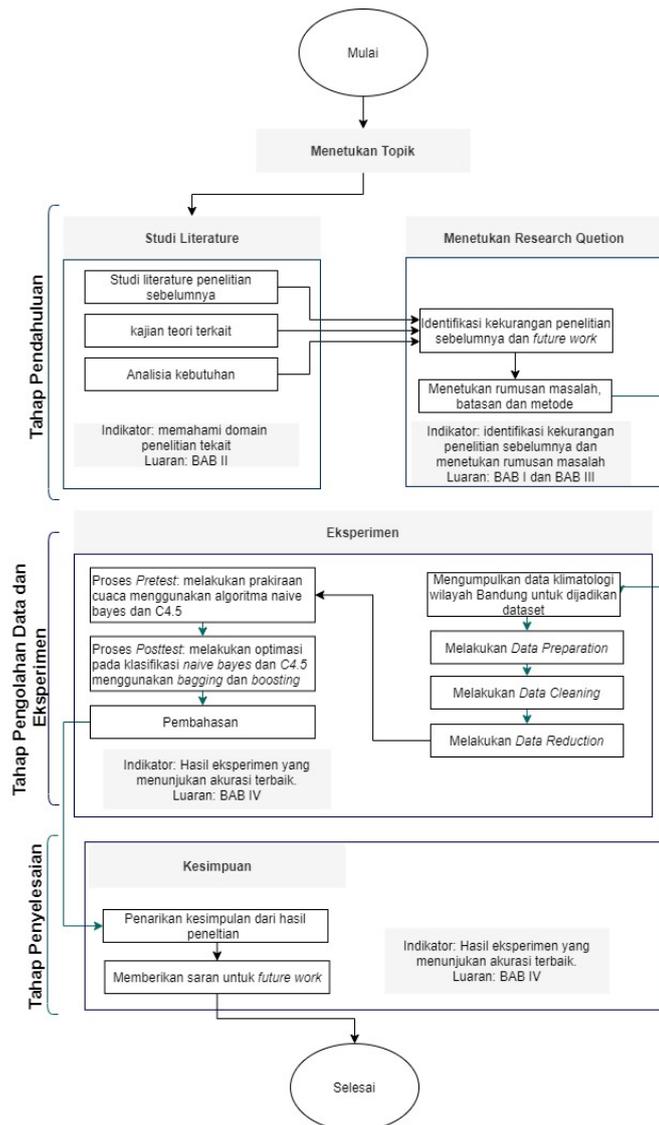
Optimasi adalah proses menyelesaikan suatu masalah tertentu supaya berada pada kondisi yang paling menguntungkan dari suatu sudut pandang tertentu. Masalah yang harus diselesaikan berkaitan erat dengan data-data yang dapat dinyatakan dalam satu atau beberapa variabel [7]. Optimasi juga dapat diartikan suatu usaha atau kegiatan untuk mendapatkan hasil terbaik dengan persyaratan yang diberikan [8]. Proses optimasi yang dilakukan memiliki objek yang jelas artinya terdapat objek-objek yang dioptimasi. Salah satu yang dapat dijadikan objek pada penelitian optimasi adalah algoritma klasifikasi. Ada beberapa metode yang dapat digunakan untuk mengoptimasi atau meningkatkan akurasi dari suatu klasifikasi, salah satunya metode *ensemble*.

Metode *ensemble* merupakan teknik yang menggabungkan prediksi dari berbagai model untuk memberikan prediksi yang lebih akurat [9]. Metode *ensemble* terdiri dari *boosting* dan *bagging*. Beberapa penelitian sebelumnya melakukan optimasi pada klasifikasi menggunakan metode *ensemble*, seperti penelitian yang mengoptimasi klasifikasi menggunakan *bagging* pada *naïve Bayes* [10], *neural network* [11] dan algoritma C4.5 [12], sementara itu penelitian lainnya menggunakan *Boosting* atau *AdaBoost* pada *naïve Bayes*, *support vector machine(SVM)* dan *decision tree* [5].

Berdasarkan paparan tersebut penelitian ini akan melakukan optimasi menggunakan metode *ensemble* (*bagging* dan *boosting*) pada prakiraan cuaca yang menggunakan algoritma *naïve Bayes* dan *algoritma C4.5*, karena berdasarkan hasil penelitian sebelumnya algoritma *naïve Bayes* dan C4.5 memiliki nilai akurasi yang cukup tinggi yaitu 82,65% [3] dan 88,89% [4]. Penelitian sebelumnya yang menggunakan *naïve Bayes* [3] dan C4.5 [4] relatif menggunakan dataset yang sedikit, oleh sebab itu penelitian ini menggunakan data cuaca selama 10 tahun, data tersebut dapat diunduh melalui <https://dataonline.bmkg.go.id>.

## II. METODE PENELITIAN

Langkah-langkah atau tahapan penelitian yang dilakukan penelitian ini digambarkan dengan Gambar 1. Pada Gambar 1 menunjukkan penelitian terdiri dari tiga tahap yakni tahap pendahuluan yang berupa studi literatur dan menentukan rumusan masalah, kemudian tahap pengolahan data dan eksperimen serta tahap penyelesaian berupa penarikan kesimpulan dari eksperimen.



Gambar 1. Tahapan Penelitian

### A. Tahap Pendahuluan

Hal pertama yang dilakukan adalah identifikasi masalah dengan melakukan studi literatur yaitu membaca dan mereview jurnal terkait prakiraan cuaca, sehingga dapat mengetahui kekurangan dan kelebihan pada penelitian-penelitian sebelumnya. Selanjutnya menentukan metode dan batasan dengan melakukan studi literatur dan studi pustaka untuk mencari solusi dari masalah yang ada.

### B. Tahap Eksperimen

#### 1. Tahap Pengumpulan Data

Hal pertama yang dilakukan pada tahap ini adalah mengumpulkan *dataset* yang dibutuhkan, data yang dikumpulkan berupa data klimatologi dari BMKG Bandung sebanyak 10 tahun.

#### 2. Tahap Preparation Data

Pengolahan awal data (*Preparation Data*) merupakan tahap awal pengolahan data, tahap ini perlu dilakukan untuk mendapatkan data yang berkualitas. Berikut merupakan beberapa teknik *preparation data* [4]:

- *Data Cleaning*  
Teknik ini menghilangkan data *noise*, data *incomplete* dan data *inconsistent*.

- *Data Reduction*

*Data reduction* adalah proses mereduksi *dataset* dengan cara mengurangi jumlah atribut yang tidak diperlukan agar lebih sedikit namun tetap bersifat informatif.

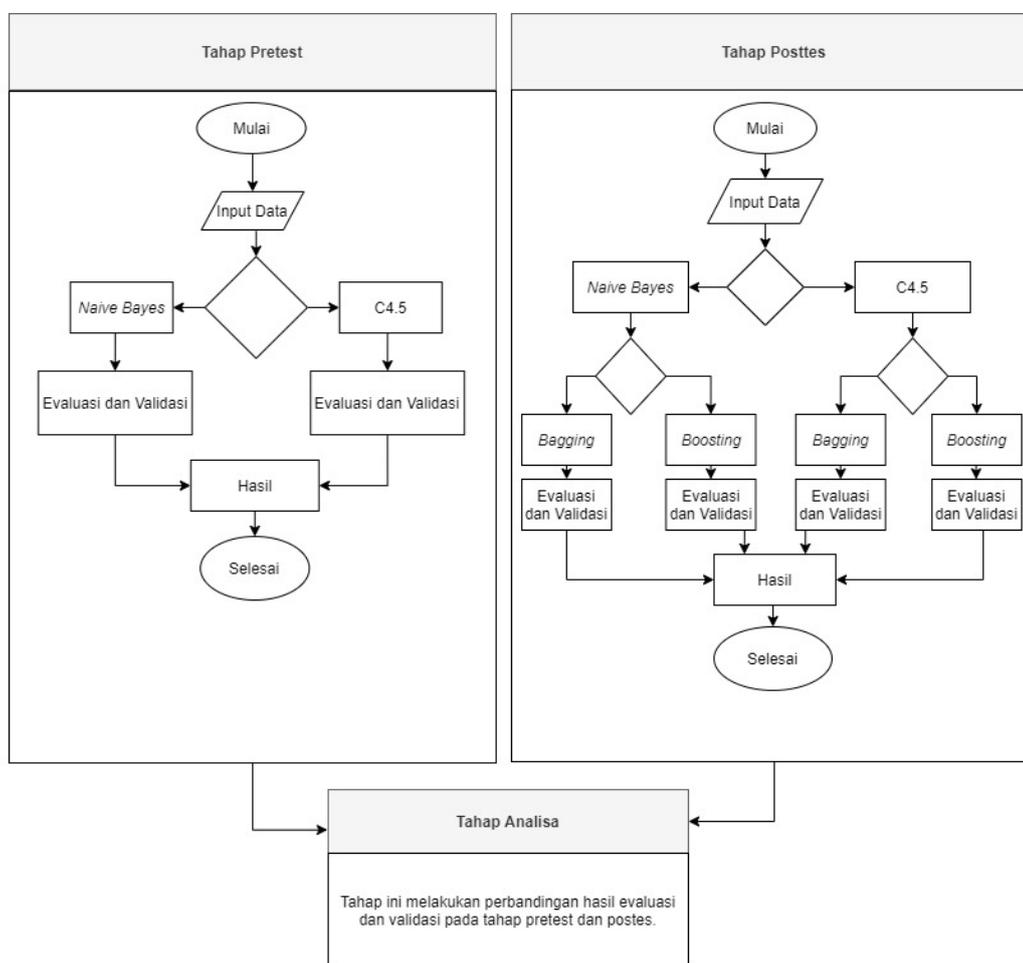
- Normalisasi

*Standardisation* atau *Z-Score normalization* merupakan salah satu teknik normalisasi data mining yang menentukan seberapa jauh suatu titik data menyimpang dari rata-ratanya. ini menghitung standar deviasi yang berada di bawah atau di atas rata-rata, mungkin di mana saja antara -3 dan +3 standar deviasi. teknik normalisasi *Z-Score normalization* dalam data mining bermanfaat untuk analisis data yang mengharuskan membandingkan suatu nilai dengan nilai *mean* (rata-rata), seperti temuan tes atau survei [13]. Berikut merupakan persamaan *Z-transformation*:

$$X_{Stand} = \frac{x - \text{mean}(x)}{\text{Standar deviasi}(x)} \quad (1)$$

### 3. Tahap Eksperimen

Proses selanjutnya dilakukan menggunakan *software RapidMiner* yang merupakan perangkat lunak yang bersifat terbuka (*open source*). RapidMiner memiliki kurang lebih 500 operator data mining, termasuk operator untuk input, output, data preprocessing dan visualisasi [14]. Gambar 2 menunjukkan tahap eksperimen pada penelitian ini.



Gambar 2. Tahap Eksperimen

Gambar 2. menunjukkan tahapan eksperimen yang dilakukan setelah melalui proses *preparation data*. Tahap eksperimen pada penelitian ini melakukan *pretest* terlebih dahulu karena menggunakan *dataset* yang berbeda dari penelitian sebelumnya, serta agar mendapatkan hasil eksperimen yang lebih valid.

- Tahap Pretest

Berdasarkan Gambar 2 pada tahap *pretest* dilakukan klasifikasi menggunakan algoritma *naïve Bayes* dan C4.5, kemudian dilakukan evaluasi menggunakan *Confusion Matrix* dan validasi menggunakan *K-folds Cross Validation*. Metode *K-fold Cross Validation* melakukan pengetesan terhadap *dataset* sebanyak *K*, data awal dibagi secara acak menjadi *K* subset saling eksklusif (berdiri sendiri) atau disebut dengan *fold*, dari ukuran yang kira-kira sama. Proses *training* dan *testing* dilakukan sebanyak *K* kali eksperimen. Pada setiap iterasi, dimana satu partisi digunakan sebagai data *testing* dan memanfaatkan sisa partisi lainnya sebagai data *training*. Eksperimen ini nilai *K* adalah 5 karena banyak iterasi akan menyebabkan banyak kemungkinan yang akan mempengaruhi nilai kebenaran dari prediksi yang dihasilkan.

Hasil perkiraan akurasi *cross validation* diperoleh dari jumlah keseluruhan klasifikasi yang benar dari iterasi *K*, dibagi dengan jumlah total *tuple* dalam data awal [15]. Sementara *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 [3]. Evaluasi menggunakan metode *confusion matrix* menghasilkan tiga luaran yakni akurasi, *precision* dan *recall*.

TABEL 1  
CONFUSION MATRIX

Correct Classification	Classified as	
	+	-
+	True positives	False negatives
-	False positives	True negatives

*Confusion Matrix* digunakan untuk evaluasi kinerja metode yang digunakan setelah klasifikasi. Untuk klasifikasi biner, skema matriks konfusi terlihat pada tabel 1. *Confusion matrix* adalah tabel yang digunakan untuk menentukan kinerja algoritma klasifikasi. *Confusion matrix* memvisualisasikan dan meringkas kinerja algoritma klasifikasi [16]. *Confusion matrix* terdiri dari empat karakteristik dasar yang digunakan untuk menentukan metrik pengukuran pengklasifikasi. Keempat nomor ini adalah:

1. TP (True Positive): jumlah dari data yang terklasifikasi di kelas yang benar.
2. TN (True Negative): jumlah dari data yang terklasifikasi di kelas sebaliknya yang benar.
3. FP (False Positive): jumlah data yang dianggap berada di kelas yang benar oleh sistem padahal seharusnya data tersebut berada di kelas yang sebaliknya.
4. FN (False Negative): jumlah data yang dianggap berada di kelas yang sebaliknya oleh sistem padahal seharusnya data tersebut berada di kelas yang benar [3].

*Performance metrics* suatu algoritma adalah akurasi, presisi, dan *recall* yang dihitung berdasarkan TP, TN, FP, dan FN yang disebutkan di atas:

- Akurasi algoritma direpresentasikan sebagai ratio data yang diklasifikasikan dengan benar (TP+TN) dengan jumlah total data (TP+TN+FP+FN).

$$\text{Akurasi} = \frac{TP+TN}{TP+TN+FP+FN} \quad (2)$$

- Presisi suatu algoritma direpresentasikan sebagai rasio data dengan penyakit yang diklasifikasikan dengan benar (TP) dengan total data yang diprediksi memiliki penyakit (TP+FP).

$$\text{Presisi} = \frac{TP}{TP+FP} \quad (3)$$

- *Recall* didefinisikan sebagai rasio data yang terklasifikasi di kelas yang benar (TP) dibagi dengan jumlah klasifikasi yang benar [16].

$$\text{Recall} = \frac{TP}{TP+FN} \quad (4)$$

- Tahap Posttest

Tahap ini merupakan tahap inti dari eksperimen, yakni melakukan *treatment* pada *pretest*. Pada penelitian ini, tahap *posttest* dilakukan proses *bagging* dan *boosting* pada *naïve Bayes* dan C4.5, lalu dilakukan evaluasi menggunakan *Confusion Matrix* dan validasi menggunakan *K-folds Cross Validation*.

- Tahap Analisis

Tahap terakhir adalah melakukan perbandingan akurasi tertinggi dan kenaikan yang terjadi setelah dilakukan *posttest*.

- Tahap Penyelesaian

Tahap ini dilakukan setelah proses eksperimen dilakukan, kemudian menarik kesimpulan dari hasil eksperimen dan memberikan saran untuk penelitian selanjutnya. Tahap akhir dari penelitian ini adalah pembuatan laporan yang disusun sesuai dengan ketentuan yang tercantum dalam pedoman penulisan Universitas Siliwangi Tasikmalaya.

### III. HASIL DAN PEMBAHASAN

Penelitian ini melakukan optimasi pada algoritma *naïve Bayes* dan C4.5 menggunakan metode *ensemble* dengan tujuan mengetahui metode atau algoritma terbaik untuk prakiraan cuaca. Proses klasifikasi, optimasi dan perhitungan akurasi dilakukan menggunakan *software* RapidMiner. *Hardware* yang digunakan adalah laptop DELL INSPIRON N4030 dengan spesifikasi:

1. Operating System: Windows 10 Pro 64-bit.
2. Processor: Intel® Core™ i5 CPU M 460 @2.53GHz
3. RAM/HDD: 4GB/500G

#### A. Pengumpulan Data

Data yang dikumpulkan merupakan data klimatologi dari BMKG Bandung dengan 12 atribut termasuk satu parameter kategorial, yakni tanggal, suhu minimum, suhu maksimum, suhu rata-rata, kelembaban rata-rata, lama penyinaran matahari, kecepatan angin maksimum, Data yang dijadikan data set adalah data dari tahun 2009-2018(10 tahun).

#### B. Data preparation

Data yang didapatkan sebanyak 3652 data yakni data klimatologi dari Januari 2009-Desember 2018. Terdapat beberapa data yang tidak lengkap atau hilang menyebabkan ketidakbenaran dan kekeliruan, oleh karena itu dilakukan *data cleaning* dan reduksi data.

##### 1. Data Cleaning

Penghapusan data *noise*, data tidak lengkap dan nilai atribut yang hilang menyebabkan data 3652 menjadi 2519 data atau berkurang sebanyak 1133 data.

##### 2. Reduksi Data

Proses ini menghapus variabel yang tidak mempengaruhi seperti *date* dan arah angin terbanyak. Arah angin terbanyak merupakan parameter yang berpengaruh pada proses terjadinya petir, tetapi dalam kategorial tidak ada. Penelitian ini adalah untuk melakukan optimasi. Oleh karena itu, variabel *date* dan arah angin terbanyak dihapus sehingga variabel input menjadi 9 dan satu atribut kategorial.

##### 3. Normalisasi

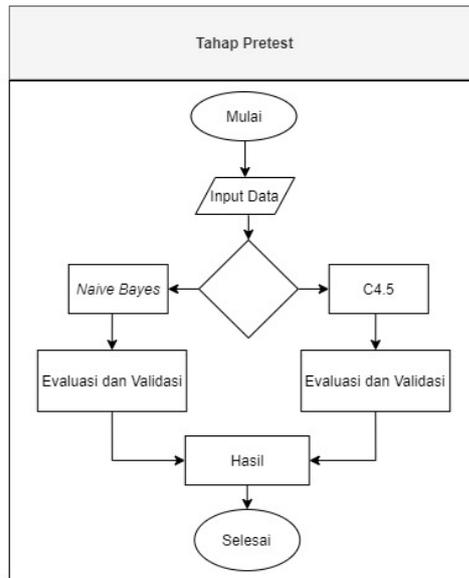
Proses normalisasi dilakukan untuk menjadikan *dataset* berada di-*range* angka yang sama

#### C. Eksperimen

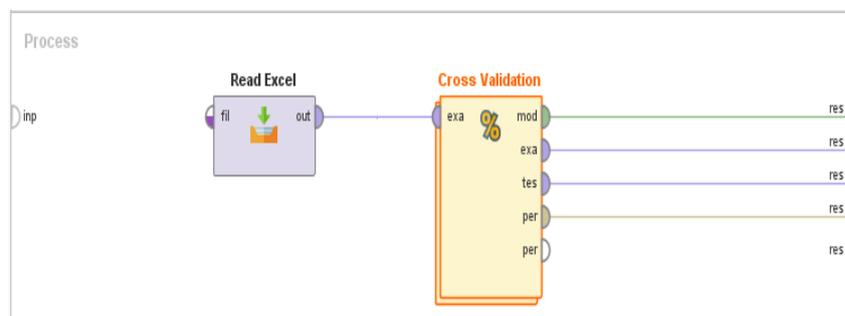
Proses eksperimen menggunakan *tools* RapidMiner sebagai *tools* yang digunakan untuk klasifikasi dan optimasi

##### 1. Pretest

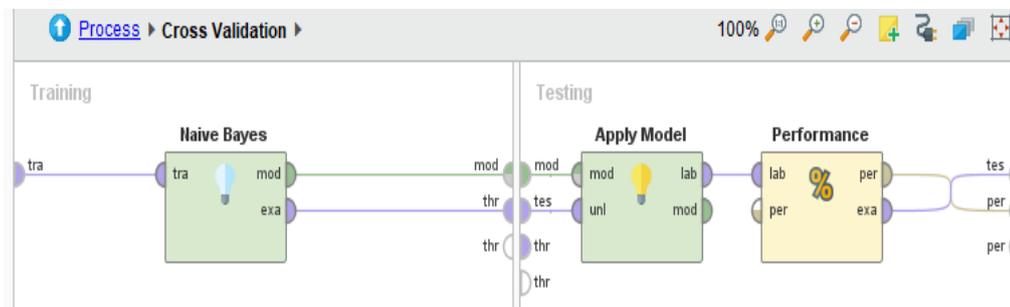
Hal yang dilakukan pada tahap *pretest* adalah mengklasifikasikan *dataset* menggunakan algoritma *naïve Bayes* dan C4.5, pada Gambar 3 digambarkan proses *pretest* dengan jelas. Langkah pertama yang dilakukan adalah menginput data kemudian melakukan klasifikasi data menggunakan algoritma yang telah ditentukan, melakukan evaluasi dan validasi kemudian hasilnya disimpan untuk dibandingkan dengan proses *posttest*.



Gambar 3. Pretest



Gambar 4. Rangkaian Klasifikasi dan Pengukuran Akurasi Algoritma Naive Bayes Bagian 1



Gambar 5. Rangkaian Klasifikasi dan Pengukuran Akurasi Algoritma Naive Bayes Bagian 2

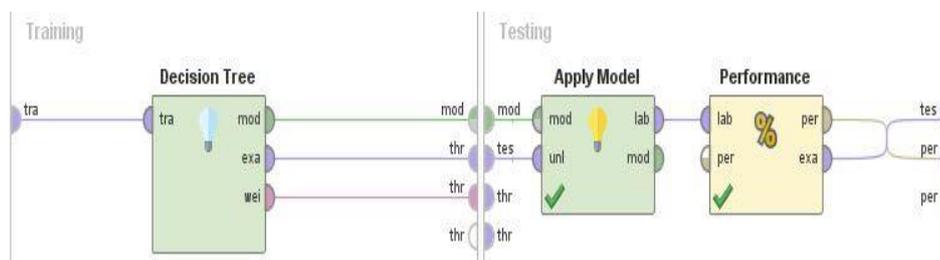
Gambar 4 menunjukkan atribut yang digunakan pada proses klasifikasi di RapidMiner. Atribut *Read Excel* berfungsi untuk membaca *dataset*, atribut *Cross Validation* berfungsi sebagai *K-folds* pada atribut ini ditentukan berapa nilai *K*. Gambar 5 menunjukkan isi atribut *Cross Validation* yang terdapat dua bagian yakni *training* dengan *naive Bayes* sebagai algoritma yang digunakan, sementara untuk *testing* menggunakan atribut *Performance* yang berfungsi seperti *confusion matrix*.

accuracy: 49.37% +/- 2.14% (micro average: 49.37%)

	true Tidak Hujan	true Hujan san...	true Hujan Ring...	true Hujan Sed...	true Hujan Lebat	class precision
pred. Tidak Huj...	674	166	86	38	8	69.34%
pred. Hujan sa...	188	382	231	86	23	41.98%
pred. Hujan Rin...	46	178	163	98	33	31.47%
pred. Hujan Se...	10	24	36	23	9	22.55%
pred. Hujan Le...	1	2	10	4	3	15.00%
class recall	73.34%	50.80%	30.99%	9.24%	3.95%	

Gambar 6. Akurasi dari Klasifikasi Naïve Bayes

Gambar 6 merupakan hasil dari proses validasi dan evaluasi dimana akurasi yang didapatkan oleh klasifikasi *naïve Bayes* adalah 49,37 %. Setiap kolom merupakan nilai keadaan sebenarnya cuaca, sementara setiap baris merupakan nilai prediksi dari cuaca. Sehingga dapat disimpulkan bahwa jika ditarik garis diagonal pada tabel menunjukkan nilai prediksi yang benar.



Gambar 7. Klasifikasi C4.5

Proses klasifikasi menggunakan C4.5 seperti proses pada Gambar 4 menggunakan atribut *Read Excel* berfungsi dan *Cross Validation*, namun pada atribut *Cross Validation* proses *training* menggunakan atribut *Decision Tree* seperti pada Gambar 7 karena eksperimen kedua menggunakan algoritma C4.5, sementara untuk proses *testing* menggunakan atribut yang sama dengan proses klasifikasi *naïve Bayes*.

accuracy: 41.24% +/- 4.09% (micro average: 41.24%)

	true Tidak Hujan	true Hujan san...	true Hujan Ring...	true Hujan Sed...	true Hujan Lebat	class precision
pred. Tidak Huj...	337	52	20	7	2	80.62%
pred. Hujan sa...	580	697	502	239	74	33.32%
pred. Hujan Rin...	0	3	4	1	0	50.00%
pred. Hujan Se...	2	0	0	2	0	50.00%
pred. Hujan Le...	0	0	0	0	0	0.00%
class recall	36.67%	92.69%	0.76%	0.80%	0.00%	

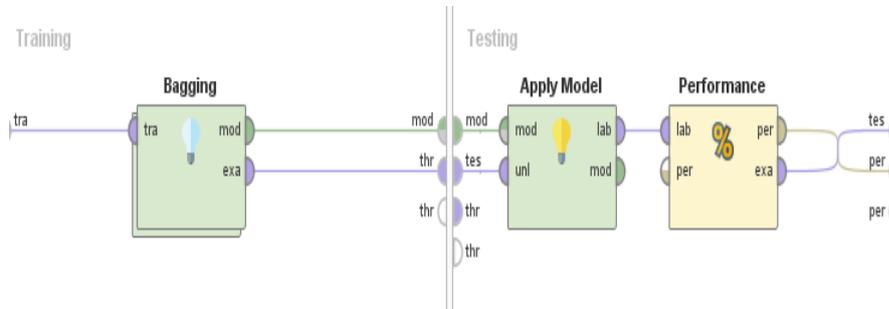
Gambar 8. Confusion Matrix pada RapidMiner

Gambar 8 menampilkan akurasi yang didapatkan oleh klasifikasi C4.5 adalah 41,24%, setiap kolom merupakan nilai keadaan sebenarnya cuaca, sementara setiap baris merupakan nilai prediksi dari cuaca. Sehingga dapat disimpulkan bahwa jika ditarik garis diagonal pada tabel menunjukkan nilai prediksi yang benar. Persentase *class recall* Hujan Lebat mendapatkan 0% karena tidak ada Hujan Lebat yang benar.

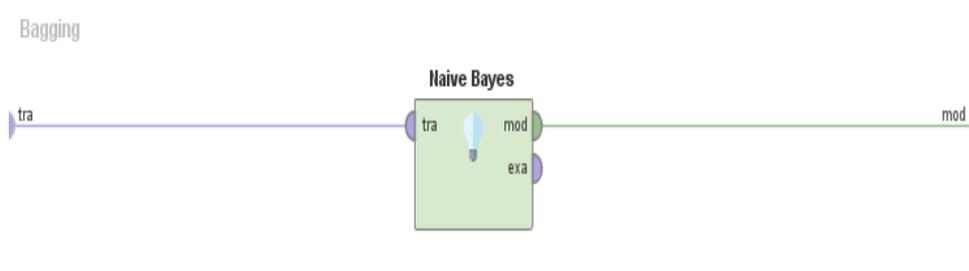
2. Posttest

Berikut ini merupakan proses *bagging* menggunakan *tools RapidMiner*.

Proses optimasi menggunakan *bagging* sama seperti pada Gambar 4, tapi pada atribut *Cross Validation* proses *training* menggunakan *Bagging* sebagai metode optimasi seperti pada gambar 9, sementara itu gambar 10 menunjukkan algoritma yang dioptimasi, pada proses ini algoritma yang dioptimasi adalah algoritma *naïve Bayes*.



Gambar 9. Detail Operator *Cross Validation*



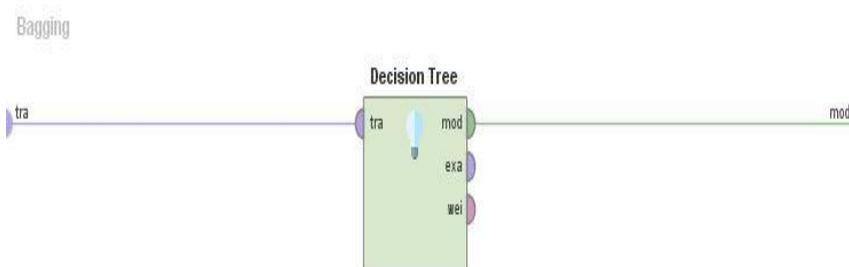
Gambar 10. Detail Operator *Bagging*

accuracy: 49.76% +/- 1.83% (micro average: 49.76%)

	true Tidak Hujan	true Hujan san...	true Hujan Ring...	true Hujan Sed...	true Hujan Lebat	class precision
pred. Tidak Huj...	673	167	84	38	8	69.38%
pred. Hujan sa...	192	388	231	91	26	41.81%
pred. Hujan Rin...	42	175	169	94	30	33.14%
pred. Hujan Se...	11	19	31	23	10	24.47%
pred. Hujan Le...	1	3	11	3	2	10.00%
class recall	73.23%	51.60%	32.13%	9.24%	2.63%	

Gambar 11. Detail Pengukuran Akurasi

Gambar 11 merupakan hasil dari proses validasi dan evaluasi menggunakan *tools RapidMiner*, dimana akurasi yang didapatkan oleh klasifikasi *naïve Bayes* yang telah dioptimasi oleh *bagging* adalah 49,76%. Hal ini menunjukkan adanya kenaikan akurasi sebanyak 0,31%. Berdasarkan Gambar 11 terdapat perubahan pada garis diagonal yang merupakan prediksi yang benar. Prediksi benar yang bertambah adalah Hujan Sangat Ringan bertambah 6, Hujan Ringan bertambah 6, sementara prediksi Tidak Hujan dan Hujan Lebat nilai kebenarannya berkurang 1.



Gambar 12. Detail Proses *Bagging C4.5*

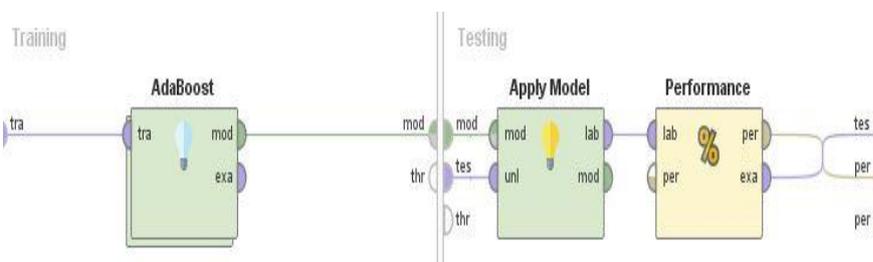
Proses optimasi *Bagging C4.5* sama seperti Gambar 9 tetapi proses *Bagging* dilakukan menggunakan atribut *Decision Tree* seperti pada Gambar 12 karena algoritma yang dioptimasi adalah *C4.5*

accuracy: 45.76% +/- 4.31% (micro average: 45.76%)

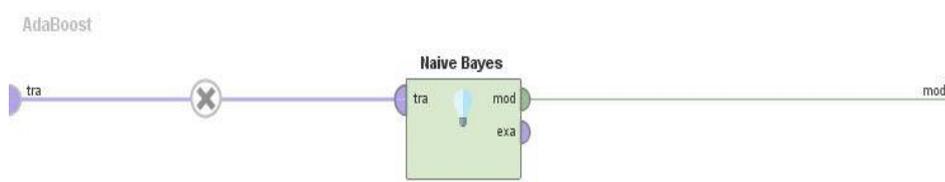
	true Tidak Hujan	true Hujan san...	true Hujan Ring...	true Hujan Sed...	true Hujan Lebat	class precision
pred. Tidak Huj...	485	83	39	14	2	77.85%
pred. Hujan sa...	429	661	481	231	74	35.23%
pred. Hujan Rin...	3	8	6	2	0	31.58%
pred. Hujan Se...	2	0	0	2	0	50.00%
pred. Hujan Le...	0	0	0	0	0	0.00%
class recall	52.77%	87.90%	1.14%	0.80%	0.00%	

Gambar 13. Detail Pengukuran Akurasi

Gambar 13 merupakan hasil dari proses validasi dan evaluasi dimana akurasi yang didapatkan oleh klasifikasi *C4.5* yang telah dioptimasi oleh *bagging* adalah 45,76%. *Bagging* pada *C4.5* menaikkan tingkat akurasi sebesar 4,52% meskipun masih belum ada prediksi Hujan Lebat yang benar. Perbaikan prediksi terbaik pada Tidak Hujan dengan bertambah 158 prediksi yang benar.



Gambar 14. Rangkaian Optimasi *Boosting* dan Pengukuran Akurasi



Gambar 15. Detail Proses *Boosting*

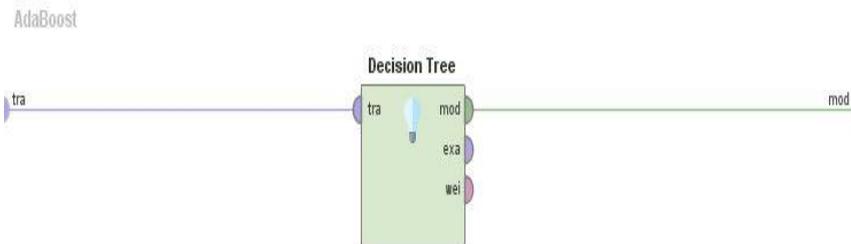
Proses optimasi menggunakan *boosting* sama seperti pada Gambar 4, tapi pada atribut *Cross Validation* proses *Training* menggunakan *AdaBoost* sebagai metode optimasi seperti pada Gambar 14. Gambar 15 menunjukkan algoritma yang dioptimasi, pada proses ini algoritma yang digunakan adalah algoritma *naive Bayes*.

accuracy: 46.47% +/- 5.08% (micro average: 46.47%)

	true Tidak Hujan	true Hujan san...	true Hujan Ring...	true Hujan Sed...	true Hujan Lebat	class precision
pred. Tidak Huj...	713	283	175	80	22	56.01%
pred. Hujan sa...	156	305	186	71	22	41.22%
pred. Hujan Rin...	41	147	135	76	24	31.91%
pred. Hujan Se...	8	14	22	18	7	26.09%
pred. Hujan Le...	1	3	8	4	1	5.88%
class recall	77.58%	40.56%	25.67%	7.23%	1.32%	

Gambar 16. Detail Pengukuran Akurasi

Gambar 16 merupakan hasil dari proses validasi dan evaluasi dimana akurasi yang didapatkan oleh klasifikasi *naïve Bayes* yang telah di *boosting* adalah 46,47%. Optimasi *boosting* tidak menaikkan akurasi, namun menurunkan akurasi sebesar -2.69%. Optimasi *boosting* memperbaiki prediksi tidak hujan dengan bertambahnya 39 prediksi yang benar, namun kebenaran pada prediksi lain menurun.



Gambar 17. Detail Proses Boosting

Proses optimasi C4.5 sama seperti Gambar 14 tetapi pada proses *AdaBoost* dilakukan menggunakan atribut *Decision Tree* seperti pada Gambar 17 karena pada proses ini algoritma yang dioptimasi adalah C4.5

accuracy: 38.82% +/- 4.70% (micro average: 38.82%)

	true Tidak Hujan	true Hujan san...	true Hujan Ring...	true Hujan Sed...	true Hujan Lebat	class precision
pred. Tidak Huj...	850	623	429	201	61	39.28%
pred. Hujan sa...	69	129	97	48	15	36.03%
pred. Hujan Rin...	0	0	0	0	0	0.00%
pred. Hujan Se...	0	0	0	0	0	0.00%
pred. Hujan Le...	0	0	0	0	0	0.00%
class recall	92.49%	17.15%	0.00%	0.00%	0.00%	

Gambar 18. Detail Pengukuran Akurasi

Gambar 18 merupakan hasil dari proses validasi dan evaluasi dimana akurasi yang didapatkan oleh klasifikasi C4.5 yang telah di *boosting* adalah 38,82%. Proses *Boosting* menyebabkan tiga *class recall* dan *class precision* bernilai nol yang artinya tidak ada prediksi untuk Hujan Ringan, Hujan Sedang dan Hujan Lebat sehingga sudah dapat dipastikan tidak ada prediksi yang benar. Hal tersebut menyebabkan nilai akurasi menurun sebanyak -2.42%.

### 3. Analisis

TABEL 2  
PERBANDINGAN AKURASI DARI HASIL EKSPERIMEN

	<i>Pretest</i>	<i>Posttest</i>		<i>Kenaikan</i>	
		<i>Bagging</i>	<i>Boosting</i>	<i>Bagging</i>	<i>Boosting</i>
<i>Naïve Bayes</i>	49,45%	49,76%	46,76%	0.31%	-2.69%
C4.5	41,24%	45,76%	38,82%	4.52%	-2.42%

Tabel 2 menunjukkan perbandingan akurasi yang dihasilkan oleh algoritma *naïve Bayes* dan C4.5, sebelum dan setelah dioptimasi menggunakan metode *ensemble* dengan bantuan *tools* RapidMiner. Metode Optimasi *Bagging* menaikkan akurasi sementara *Boosting* menurunkan nilai akurasi dari proses *Pretest*.

Akurasi pada proses *pretest* menunjukkan jika algoritma *naïve Bayes* memiliki akurasi sebesar yakni 49,45%, sementara algoritma C4.5 menghasilkan akurasi 41,24%. Akurasi pada proses *posttest* menunjukkan hasil yang sama, yakni algoritma *naïve Bayes* memiliki akurasi yang lebih baik, namun algoritma C4.5 mendapatkan kenaikan akurasi yang lebih baik daripada algoritma *naïve Bayes*.

### IV SIMPULAN

Hasil eksperimen menunjukkan jika optimasi menggunakan metode *ensemble bagging* pada *naïve Bayes* menaikkan akurasi sebesar 0,31%, sementara optimasi yang dilakukan menggunakan *boosting* membuat akurasi turun yakni sebesar -2,69%. Berbeda dengan algoritma *naïve Bayes*, algoritma C4.5 memiliki kenaikan akurasi yang lebih baik setelah dioptimasi menggunakan metode *ensemble*. Algoritma C4.5 memiliki kenaikan akurasi sebesar 4.52% setelah dioptimasi menggunakan metode *ensemble bagging*, sedangkan setelah dioptimasi menggunakan metode *boosting* mengalami penurunan akurasi sebesar -2.42%.

Metode Ensemble terbukti mempengaruhi klasifikasi. Berdasarkan Tabel 2. *Bagging* selalu meningkatkan akurasi, sedangkan *boosting* membuat akurasi turun. Hal ini terjadi karena perbedaan pemodelan data *training*. Pemodelan *bagging* bersifat *parallel* menyebabkan setiap model bersifat *independent*, sehingga dapat membuat akurasi menjadi lebih baik. Pemodelan *boosting* bersifat *sequential* sehingga membuat model baru tidak *independent*, sehingga mengakibatkan akurasi menurun. Hal tersebut terjadi karena *dataset* yang tersedia memiliki banyak *variance* yang nilainya hampir sama untuk setiap kategori, sehingga sulit untuk dimodelkan dengan konsep *boosting* yang bersifat *sequential*.

Hasil dari eksperimen menunjukkan jika *dataset* dari BMKG Bandung memiliki banyak *variance* yang nilainya hampir sama untuk setiap kategori, sehingga sulit untuk dimodelkan bahkan setelah dilakukan optimasi oleh *metode ensemble*. Penelitian selanjutnya mungkin dapat menambahkan *knowing base* dari pakar untuk mendapatkan akurasi yang lebih baik.

### DAFTAR PUSTAKA

- [1] K. Abhishek, M. Singh, S. Ghosh and A. Anand, "Weather forecasting model using Artificial Neural Network," *Procedia Technology*, vol. 4, no. 2212-0173, pp. 311-318, 2012.
- [2] I. W. Harmoko and N. AZ, "Prototipe Model Prediksi Peluang Kejadian Hujan Menggunakan Metode Fuzzy Logic Tipe Mamdani dan Sugeno," *TICOM*, vol. I, no. 1, pp. 59-16, 2012.
- [3] A. Subhan and A. Z. Fanani, "Penerapan Data Mining untuk Menentukan Potensi Hujan Harian dengan Menggunakan Algoritma Naive Bayes," Universitas Dian Nuswantoro, Semarang, 2017.
- [4] A. Novandya and I. Oktaria, "Penerapan Algoritma Klasifikasi Data Mining C4.5 Pada Dataset Cuaca Wilayah Bekasi," *Jurnal Format*, vol. 6, no. 2, pp. 98-106, 2017.
- [5] Y. Pratama, "Instituinal Repository UIN Sunan Kalijaga Yogyakarta," 03 Agustus 2016. [Online]. Available: <http://digilib.uin-suka.ac.id/id/eprint/21303>.
- [6] V. Kurniati, D. Triyanto and T. Rismawan, "Penerapan Logika Fuzzy dalam Sistem Prakiraan Cuaca Berbasis Mikrokontroler," *Jurnal Coding, Sistem Komputer Untan*, Vols. 05, No. 2, pp. 119-128, 2017.
- [7] D. C. Putri Buani, "Optimasi Algoritma Naïve Bayes dengan Menggunakan Algoritma Genetika," *Jurnal Evolusi*, vol. 4 Nomor 1, pp. 55-64, 2016.
- [8] A. Wanto, "Optimasi Prediksi Dengan Algoritma Backpropagation Dan Conjugate Gradient," *Jurnal Nasional Teknologi dan Sistem Informasi*, vol. 03 Nomor 3, pp. 370-380, 2018.
- [9] J. Brownlee, *Master Machine Learning Algorithms Discover How They Work and Implement Them From Scratch*, Melbourne: Machine Learning Mastery, 2017.
- [10] Z. Xiao-Dong, C. Sam and W. Fai, "Optimization of Bagging Classifiers Based on SBCB Algorithm," in *Proceedings of the Ninth International Conference on Machine Learning and Cybernetics*, Qingdao, 2010.

- [11] M. N. Shobary, "Optimasi Pengembangan Biaya Software dengan Perbandingan Neural Network dengan Optimasi Algoritma Genetika dan Bagging," *Jurnal INSTEK (Informatika Sains dan Teknologi*, pp. 21-30, 2018.
- [12] Mirqotussa'adah, M. A. Muslim and E. Sugiharti, "Penerapan Dizcretization dan Teknik Bagging Untuk Meningkatkan Akurasi Klasifikasi Berbasis Ensemble pada Algoritma C4.5 dalam Mendiagnosa Diabetes," *Lontar Komputer*, vol. VIII, no. 2, pp. 135-143, 2017.
- [13] K. Tolety, "Data Normalization Techniques in Data Mining Simplified 101," HEVO, 27 Mei 2022. [Online]. Available: <https://hevodata.com/learn/normalization-techniques-in-data-mining/#8>. [Accessed 16 November 2022].
- [14] D. Aprilla, D. A. Baskoro, L. Ambarwati and I. W. S. Wicaksana, *Belajar Data Mining dengan RAPIDMINER*, Jakarta, 2013.
- [15] A. I. Irawan, R. Saptono and A. Doewes, "Implementasi of Naive Bayes Classifier Method and Adaboost Algorithm for Predicton of Chronic Kidney Disease," 2016. [Online]. Available: [perpustakaan.uns.ac.id](http://perpustakaan.uns.ac.id). [Accessed 2019].
- [16] P. Singh, N. Singh, K. K. Singh and A. Singh, "Diagnosing of disease using machine learning," in *Machine Learning and the Internet of Medical Things in Healthcare*, Amsterdam, Elsevier, 2021, pp. 89-111.