JURNAL RESISTOR
Vol. 2 No 2 – Oktober 2019
p-ISSN 2598-7542 (Print), e-ISSN 2598-9650 (Online)
Available Online at :
http://jurnal.stiki-indonesia.ac.id/index.php/jurnalresistor

PENANGANAN DATA *MISSING VALUE* PADA KUALITAS PRODUKSI JAGUNG DENGAN MENGGUNAKAN METODE *K-NN IMPUTATION* PADA ALGORITMA C4.5

Moch. Lutfi[1], Mochamad Hasyim[2]

[1,2] Universitas Yudharta Pasuruan
Jl. Yudharta No.7, Kembangkuning, Sengonagun, Purwosari, Pasuruan, Jawa Timur 67162

Email: moch.lutfi@yudharta.ac.id[1], hasyim@yudharta.ac.id[2]

Received : Agustus, 2019	Accepted : October, 2019	Published : October, 2019

Abstract

Corn is a staple crop for Indonesian people because most of his life is from the agriculture sector. To increase the productivity of corn, another thing to be aware of is looking at the quality of the corn products. Through empirical observations and observations, this research explores and extracts data through the concept of data mining so that neglected data becomes useful. Determining the quality of maize production is an important task to assist farmers in determining the classification process. Missing value is a problem in maintaining a quality data. Missing value can be caused by several things, one of which is caused by an error at the time of data entry. Missing value will be a problem when the amount of data in large quantities, so it is very influential in the survey results. Therefore on this research proposed K-NN imputation method to handle missing value data. The results showed the accuracy of the C 4.5 algorithm classification process on the corn production dataset that experienced a missing value accuracy value of 92.90%. Whereas if done with special handling using the method K-NN imputation on the handling process missing value best value at k = 5 of 94.50% with this that the proposed method increases significantly.

Keywords: quality of corn production, Data Mining, missing value, K-NN imputation, C 4.5

Abstrak

Jagung adalah tanaman pokok bagi masyarakat indonesia karena sebagian besar hidupnya dari sektor pertanian. Untuk meningkatkan produktivitas jagung, hal lain yang harus diperhatikan adalah melihat kualitas produk jagung tersebut. Melalui pengamatan dan observasi secara empiris, penelitian ini menggali dan mengekstrak suatu data dengan melalui konsep data mining sehingga data yang terabaikan menjadi bermanfaat. Menentukan kualitas produksi jagung merupakan tugas penting untuk membantu para petani dalam menentukan proses klasifikasi. *Missing value* merupakan masalah dalam menjaga suatu kualitas data. *Missing value* dapat disebabkan oleh beberapa hal, salah satunya diakibatkan oleh kesalahan pada saat entri data. *Missing value* akan menjadi masalah ketika jumlah data dalam jumlah besar, sehingga sangat berpengaruh sekali terhadap hasil survey. Oleh karena itu pada penelitian ini diusulkan metode *K-NN imputation* untuk menangani data *missing value*. Hasil penelitian menunjukan akurasi proses klasifikasi algoritma C4.5 pada dataset produksi jagung yang mengalami *missing value* nilai akurasi sebesar 92.90%. Sedangkan jika dilakukan dengan penanganan khusus menggunakan metode *K-NN imputation* pada proses penanganan *missing value* nilai terbaik pada k=5 sebesar 94.50% dengan ini bahwa metode yang diusulkan meningkat secara signifikan.

Kata kunci: ***Kualitas Produksi Jagung, Data Mining, Missing Value, K-NN Imputation, C4.5.***

1. PENDAHULUAN

Jagung adalah tanaman pokok bagi masyarakat indonesia karena sebagian besar hidupnya dari sektor pertanian. Namun beberapa tahun kedepan lahan pertanian akan semakin menyempit akibat pertumbuhan sektor

industri yang perkembang pesat sehingga mengurangi produktivitas hasil pertanian. Untuk meningkatkan produktivitas jagung, hal lain yang harus diperhatikan adalah peningkatan kualitas produk jagung tersebut. Namun kenyataan-nya, permasalahan kualitas pada biji jagung sampai saat ini masih menjadi sebuah persoalan[1]. Sehingga perlu adanya metode untuk menyelesaikan masalah penentuan kualitas produksi jagung salah satunya dengan algoritma C4.5.

Decision Tree adalah salah satu pendekatan yang paling popular dalam klasifikasi [2]. *Decision Tree* dipakai diberbagai disiplin ilmu seperti bidang medis, statistik, *machine learning*, *pattern recognation*, dan data mining. Para peneliti telah mengembangkan berbagai metode *Decision Tree* dalam menyelesaikan dataset yang ada.

Namun C4.5 juga dapat menyebabkan *over-fitting* yang disebabkan oleh *noisy data* dan *irrelevant feature*[3]. *Over fitting* dapat menyebabkan tingkat akurasi yang buruk dalam proses klasifikasi. Selain *over fitting*, yang menjadi tantangan di bidang algoritma C4.5 adalah *imbalanced data, missing value* dan tingkat akurasi.

Missing value merupakan masalah dalam menjaga suatu kualitas data. *Missing value* dapat disebabkan oleh beberapa hal, salah satunya diakibatkan oleh kesalahan pada saat entri data. *Missing value* akan menjadi masalah ketika jumlah data dalam jumlah besar, sehingga sangat berpengaruh sekali terhadap hasil survey. *Missing value* dapat menyebabkan tingkat keakuratan suatu data menjadi berkurang dan menurunnya kualitas data pada saat akan dilakukan pengolahan data lanjut, seperti proses klasifikasi. Oleh karena itu, diperlukan penanganan khusus untuk mengatasi *missing value* ini.

Ada beberapa penelitian tentang penanganan missing value yaitu Metode yang diusulkan Erna Sri Rahayu, [6] dinamakan *Average Gain* (AG), dimana AG adalah metode split atribut menggunakan *average gain* yang dikalikan dengan selisih misklasifikasi. Setelah proses split atribut dilanjutkan dengan teknik *pruning*. Teknik *pruning* yang digunakan yaitu *threshold pruning* dan *cost complexity pruning*. Metode AG yang diintegrasikan dengan *threshold pruning* dan *cost complexity pruning* selanjutnya dalam penelitian ini disebut AG_Pruning. Minakhsi [4] melakukan penelitian pada dataset yang mengandung *missing values*. Teknik Imputasi yang digunakan ada tiga, yaitu *litwise deletion*, *mean imputation* dan *K-NN imputation*, kemudian dilakukan klasifikasi menggunakan algoritma C4.5. Song [8] *K-NN imputation* mampu meningkatkan akurasi. Hasil dari penelitian ini menunjukkan bahwa keenam dataset secara rata-rata menghasilkan akurasi prediksi menggunakan algoritma C4.5 meningkat sebesar 6% untuk data yang lengkap *(K-NN imputation)* dibandingkan data yang tidak lengkap (data yang masing mengandung *missing values*).

Dalam penelitian ini dilakukan penanganan data *missing value* dengan metode K-NN *imputation* pada algoritma C4.5 untuk menentukan kualitas produksi jagung.

2. KAJIAN PUSTAKA DAN METODE PENELITIAN

2.1 Algoritma C4.5

C4.5 adalah algoritma klasifikasi *supervised learning* untuk membentuk pohon keputusan *(Decision Tree)* dari sebuah dataset. Adapun tahapan dalam membuat sebuah pohon keputusan dalam algoritma C4.5[9] yaitu:

1. Mempersiapkan data *training*. Data *training* biasanya diambil dari data histori yang pernah terjadi sebelumnya atau disebut data masa lalu dan sudah dikelompokkan dalam kelas-kelas tertentu.
2. Menghitung akar dari pohon. Akar akan diambil dari atribut yang akan terpilih, dengan cara menghitung nilai *gain* dari masing-masing atribut, nilai *gain* yang paling tinggi yang akan menjadi akar pertama. Sebelum menghitung nilai *gain* dari atribut, hitung dahulu nilai *entropy*. Untuk menghitung nilai *entropy* digunakan rumus :

$$Entropy(S) = \sum_{i=1}^{n} - pi * log_2 pi \quad (2.1)$$

Keterangan:

S = Himpunan kasus
n = Jumlah partisi S
pi = Proporsi S_i terhadap S

Kemudian menghitung nilai *gain* menggunakan rumus :

$$Gain(S, A) = Entropy(s) \sum_{i=1}^{n} \frac{|S_i|}{|S|} * Entropy(S_i) \quad (2.2)$$

Keterangan:

S = Himpunan kasus
A = Atribut
n = Jumlah partisi atribut A
$|S_i|$ = Proporsi S_i terhadap S

$|S|$ = Proporsi kasus dalam S

3. Ulangi langkah ke-2 dan langkah ke-3 hingga semua *record* terpartisi.
4. 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.

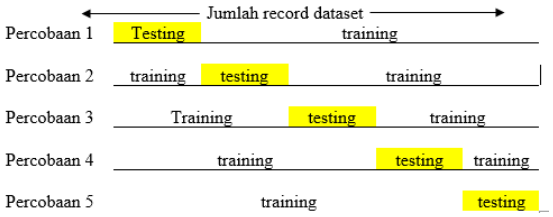
2.2 K-NN Imputation

Pada *K-NN Imputation* pengisian *missing value* dilakukan dengan memperhitungkan jarak vektor antar atribut. Adapun algoritmanya[10] adalah:

1. Dataset dipisah menjadi 2 bagian:
   - D_m yang berisi atribut dengan minimal satu *missing value.*
   - $D_k$yang berisi atribut dengan value yang lengkap.
2. Untuk setiap vektor di D_m
   a. Pisah vektor menjadi 2 yaitu: vektor observed (x_o) dan missing (x_m) dengan x=[x_o: x_m]
   b. Hitung jarak antara x_o dengan seluruh vektor dari D_k, yang dihitung hanya vektor dr D_k yang sebaris dengan x_o.
   - Untuk *categorical attributes*, pengisian missing value dengan modus dari nilai hasil *K-NN imputation*.
   - Untuk *continuous atrributes*, pengisian missing value dengan mean atau median dari nilai hasil *K-NN imputation*.

2.3 Cross Validation

Cross Validation merupakan metode statistik mengevaluasi dengan membagi data menjadi dua segmen: satu digunakan untuk training dan yang lain digunakan untuk memvalidasi model/*testing*. Dalam *cross validasi*, *training* set dan testing set diatur sedemikian rupa sehingga setiap data pernah menjadi *training* set dan testing set. Bentuk *cross-validation* adalah *k-fold validation*. Misalnya: *5-fold validation* berarti bahwa *record* dataset dibagi 4 subset menjadi *training set* dan 1 subset sebagai *testing set.* Ilustrasi *5-fold validation* dapat dilihat pada gambar 2.1.



Gambar 2.1 Pembagian Dataset untuk 5-Fold Validation

2.4 Confusion Matrix

Confusion Matrix merupakan tabel klasifikasi yang mengandung informasi hasil perhitungan sistem secara keseluruhan [11]. Pengukuran data di evaluasi melalui akurasi, *presisi* dan *recall*. Hasil pengukuran direpresentasikan ke dalam sebuah tabel klasifikasi untuk memudahkan pembacaan. Akurasi adalah jumlah prosentase dari klasifikasi system yang tepat. *Presisi* merupakan ukuran dari akurasi suatu *class* yang telah di klasifikasi oleh sistem. Sedangkan *recall* adalah presentase data dengan nilai positif dari hasil klasifikasi yang nilainya juga positif. Adapun perhitungannya yaitu :

Akurasi = (TP+TN) / (TP+TN+FP+FN) (2.3)

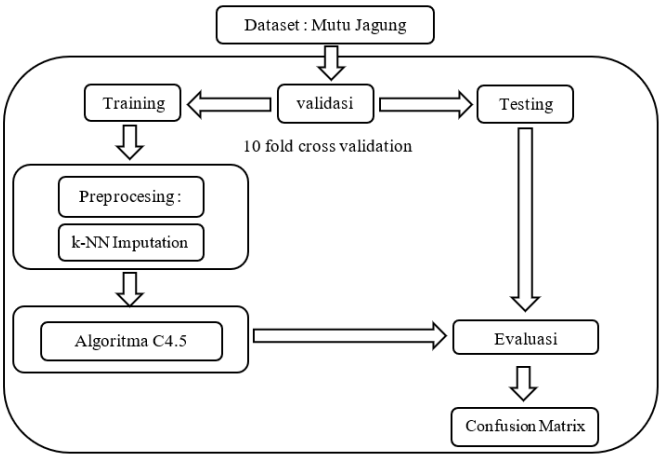
Presisi = TP / (TP+FN) (2.4)

Recall = TP / (TP+FP*)* (2.5)

		Predicted	
		Positive	Negative
Actual	Positive	TP	FN
	Negative	FP	TN

2.5 Design Sistem

Metode yang diusulkan pada penelitian ini adalah penerapan *K-NN imputation* untuk meningkatkan akurasi proses klasifikasi pada algoritma klasifikasi C4.5. Untuk menggambarkan alur sistem yang diusulkan dapat dijelaskan dalam gambar 2.2 sebagaimana berikut:



Gambar 2.2 Design sistem secara umum

Pada gambar 2.2 diatas design sistem secara umum dijelaskan bahwa dataset jagung di inputkan kemudian data di pecah menjadi dua bagian atau validasi dengan 10 fold validation yaitu *training* dan *testing*. Sebelum data di *testing* data terlebih dahulu di *preprocessing* menggunakan metode *K-NN imputation* karena dataset mengandung *missing value*, setelah dataset sudah terisi dan tidak mengalami *missing value* maka data tersebut di *training* dengan algoritma C4.5. Tahap selanjutnya data di *testing* untuk di evaluasi keakuratan data menggunakan *confusion matrix*.

2.6 Dataset

Dataset yang digunakan pada penelitian ini, yaitu data produksi jagung dinas pertanian kabupaten bojonegoro. Dataset yang dipilih mempunyai karakteristik *missing value* yang beragam. Pada dataset ini mempunyai observasi sebanyak 1000 data, dengan total *missing value* sebanyak 49 *record*, berikut tabel data *missing value*.

Tabel 2.1 Attribut *Missing Value* Dataset mutu produksi jagung

No	Atribut	Missing Value (MV)
1	VARIATAS	3
2	PANJANG	26
3	BENTUK	6
4	WARNA	2
5	RASA	3
6	MUSIM	7
7	HAMA	2
	Total MV	**49**

3. ANALISIS DAN PEMBAHASAN

3.1 Pengisian *Missing Value*

Pengisian *missing value* pada dataset Mutu Jagung dengan menggunakan *software* R Studio. Imputasi *missing values* dilakukan dengan software R Studio sedangkan *package* yang digunakan adalah VIM (*Visualization and Imputation of Missing Values*) versi 4.6.0. dengan perintah sebagai berikut:

Untuk menentukan directori dataset yang tersimpan:

- setwd("C:/Users/MochLutfi/Documents/latihan")

Untuk membaca file bertipe CVS:

- jagung<-read.csv("jagung.csv")

Melihat struktur data frame:

- str(jagung)

Melihat attribut dataset:

- head(jagung)

Konversi data missing value string ke numerik:

- jagung$Variatas[jagung$Variatas ==""] = NA
- jagung$Bentuk[jagung$Bentuk ==""] = NA
- jagung$Warna[jagung$Warna ==""] = NA
- jagung$Rasa[jagung$Rasa ==""] = NA
- jagung$Musim[jagung$Musim ==""] = NA
- jagung$Hama[jagung$Hama ==""] = NA
- jagung$Panjang[jagung$Panjang ==""] = NA

Melihat seluruh isi dataset:

- summary(jagung)

Mengaktifkan librari K-NN imputation:

- library(VIM)

Input data missing value dengan metode K-NN:

- jagung5<-kNN (jagung,variable=c("Variatas","Panjang","Bentuk","Warna","Rasa","Musim","Hama"),k=5)

Melihat struktur dataset yang sudah terinput dengan K-NN:

- str(jagung5)

Melihat attribut dataset yang sudah terinput dengan K-NN:

- head(jagung5)

Melihat isi seluruh dataset yang sudah terinput dengan K-NN:

- summary(jagung5)

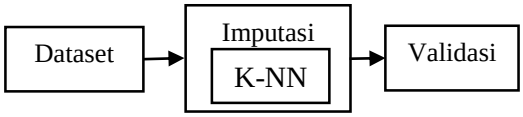
Menentukan direktori tempat untuk penyimpanan hasil export:

- setwd("C:/Users/Moch Lutfi/Documents/latihan")

Export dataset ke dalam tipe file csv yang sudah terisi value dengan K-NN:

- write.csv(jagung5, file = "jagung_K5.csv")

Pada eksperimen kedua penanganan data *missing value* dengan *tools* rapidminer tahap pertama melakukan pembacaan dataset kemudian melakukan *preprocessing* data dengan operator *impute missing value* dan metode yang digunakan *k-nearest neigbor*.



Setelah dilakukan proses input *value* tahap selanjutnya dilakukan validasi dengan *cross validation* dan penerapan terhadap algoritma C4.5 yang digunakan sebagai proses klasifikasi mutu produksi jagung. Sedangkan dalam perhitungan manual dibawah ini hasil dari imputasi menggunakan *tools* bantu rapidminer menggunakan operator *impute missing value* dengan metode *K-Nearest Neighbor* sedangkan nilai yang K yang digunakan k1, k2, k3, k4 dan k5 sebagai evaluasi kinerja metode yang diusulkan.

Tahap selanjutnya setelah terisi data yang mengandung *missing value* di hitung dan diproses dengan metode C4.5 sejauh mana kinerja metode tersebut jika terapkan pada dataset mutu jagung.

3.2 Implementasi Split Atribut

Metode C4.5 yang diusulkan pada penelitian ini diterapkan pada dataset mutu jagung diambil sebanyak 100 data dari 1000 data hasil dari penanganan *missing value* yang digunakan sebagai contoh perhitungan manual algoritma C4.5 dengan langkah-langkah sebagai berikut:

Tabel 3.1 dataset yang sudah terisi value

No	Variatas	Panjang	Bentuk	Warna	Rasa	Teknik	Musim	Hama	PH	Mutu
1.	Hibrida	9.3	Oval	Merah	pulen	Tumpang Sari	Hujan	tikus	5.2	Grade-C
2.	Pioner	8.9	Oval	merah	sangat-pulen	Tumpang Sari	Hujan	Ulat grayak	5.2	Grade-C
3.	Arjuna	9	Oval	Merah	sangat-pulen	Tumpang Gilir	Hujan	tikus	5.2	Grade-D
4.	Hibrida	9.2	Oval	Merah	pulen	Tumpang Gilir	Hujan	Ulat grayak	5.2	Grade-D
5.	Bima	8.5	Pipih	Merah	pulen	Tumpang Gilir	Hujan	tikus	5.2	Grade-C
6.	BISI	8.9	kecil	Kuning	sangat-pulen	Tumpang Sari	Hujan	penggerek-batang jagung	5.2	Grade-D
7.	Hibrida	9.3	Oval	Merah	pulen	Tumpang Sari	Kemarau	Ulat grayak	5.5	Grade-C
8.	Pioner	9.1	Oval	merah	sangat-pulen	Tumpang Sari	Kemarau	Penggerek tongkol	5.5	Grade-C
9.	…..	…..	…..	…..	…..	…..	…..	…..	…..	…..
100.	Arjuna	9.3	Oval	Merah	sangat-pulen	Tumpang Sari	Hujan	tikus	5.2	Grade-C

3.3 Menghitung Entropy

Dimulai dari *node* akar, maka harus dihitung terlebih dahulu *entropy* untuk node akar (semua data) terhadap kelas.

$$Entropy(S) = \left(-\left(\frac{31}{100}\right) \times \log 2\left(\frac{31}{100}\right)\right) + \left(-\left(\frac{49}{100}\right) \times \log 2\left(\frac{49}{100}\right)\right) + \left(-\left(\frac{20}{100}\right) \times \log 2\left(\frac{20}{100}\right)\right) = 1.492461891$$

Tabel 3.2 perhitungan entropi

total kasus	**grade-B**	**grade-C**	**grade-D**	**entropi**
100	31	49	20	1.492461891

Setelah mendapatkan entropi dari keseluruhan kasus, lakukan analis pada setiap atribut, nilai-nilainya, dan hitung entropinya.

3.3.1 Menghitung *Entropy* tiap *class* pada semua atribut Variatas.

$$Entropy(Hibrida) = \left(-\left(\frac{7}{23}\right) \times \log 2\left(\frac{7}{23}\right)\right) + \left(-\left(\frac{10}{23}\right) \times \log 2\left(\frac{10}{23}\right)\right) + \left(-\left(\frac{6}{23}\right) \times \log 2\left(\frac{6}{23}\right)\right) = 1.550494982$$

$$Entropy(Pioner) = \left(-\left(\frac{6}{14}\right) \times \log 2\left(\frac{6}{14}\right)\right) + \left(-\left(\frac{8}{14}\right) \times \log 2\left(\frac{8}{14}\right)\right) + \left(-\left(\frac{0}{14}\right) \times \log 2\left(\frac{0}{14}\right)\right) = 0$$

$$Entropy(Arjuna)$$
$$= \left(-\left(\frac{7}{21}\right) \times \log 2\left(\frac{7}{21}\right)\right) + \left(-\left(\frac{8}{21}\right) \times \log 2\left(\frac{8}{21}\right)\right) + \left(-\left(\frac{6}{21}\right) \times \log 2\left(\frac{6}{21}\right)\right)$$
$$= 1.575114591$$
$$Entropy(Bisi) = \left(-\left(\frac{4}{15}\right) \times \log 2\left(\frac{4}{15}\right)\right) + \left(-\left(\frac{7}{15}\right) \times \log 2\left(\frac{7}{15}\right)\right) + \left(-\left(\frac{5}{15}\right) \times \log 2\left(\frac{5}{15}\right)\right)$$
$$= 1.54994164$$
$$Entropy(Kumala)$$
$$= \left(-\left(\frac{3}{13}\right) \times \log 2\left(\frac{3}{13}\right)\right) + \left(-\left(\frac{10}{13}\right) \times \log 2\left(\frac{10}{13}\right)\right) + \left(-\left(\frac{0}{13}\right) \times \log 2\left(\frac{0}{13}\right)\right) = 0$$
$$Entropy(Bima)$$
$$= \left(-\left(\frac{4}{14}\right) \times \log 2\left(\frac{4}{14}\right)\right) + \left(-\left(\frac{6}{14}\right) \times \log 2\left(\frac{6}{14}\right)\right) + \left(-\left(\frac{4}{14}\right) \times \log 2\left(\frac{4}{14}\right)\right)$$
$$= 1.556656707$$

3.3.2 Menghitung *Entropy* tiap *class* pada atribut Panjang

Karena *value* nya *numerik* pada attribut panjang maka di kategorikan menjadi attribut <=9 dan >9 dengan tujuan *value numerik* menjadi nominal kategori.

$$Entropy(\leq 9) = \left(-\left(\frac{21}{37}\right) \times \log 2\left(\frac{21}{37}\right)\right) + \left(-\left(\frac{33}{37}\right) \times \log 2\left(\frac{33}{37}\right)\right) + \left(-\left(\frac{9}{37}\right) \times \log 2\left(\frac{9}{37}\right)\right)$$
$$= 1.107096357$$
$$Entropy(\leq 9) = \left(-\left(\frac{10}{63}\right) \times \log 2\left(\frac{10}{63}\right)\right) + \left(-\left(\frac{15}{63}\right) \times \log 2\left(\frac{15}{63}\right)\right) + \left(-\left(\frac{11}{63}\right) \times \log 2\left(\frac{11}{63}\right)\right)$$
$$= 1.354058564$$

3.3.3 Menghitung *Entropy* tiap *class* pada atribut Bentuk

$$Entropy(Oval)$$
$$= \left(-\left(\frac{22}{70}\right) \times \log 2\left(\frac{22}{70}\right)\right) + \left(-\left(\frac{36}{70}\right) \times \log 2\left(\frac{36}{70}\right)\right) + \left(-\left(\frac{12}{70}\right) \times \log 2\left(\frac{12}{70}\right)\right)$$
$$= 1.454363793$$
$$Entropy(Pipih)$$
$$= \left(-\left(\frac{4}{14}\right) \times \log 2\left(\frac{4}{14}\right)\right) + \left(-\left(\frac{6}{14}\right) \times \log 2\left(\frac{6}{14}\right)\right) + \left(-\left(\frac{4}{14}\right) \times \log 2\left(\frac{4}{14}\right)\right)$$
$$= 1.556656707$$
$$Entropy(Kecil)$$
$$= \left(-\left(\frac{5}{16}\right) \times \log 2\left(\frac{5}{16}\right)\right) + \left(-\left(\frac{7}{16}\right) \times \log 2\left(\frac{7}{16}\right)\right) + \left(-\left(\frac{4}{16}\right) \times \log 2\left(\frac{4}{16}\right)\right)$$
$$= 1.546179692$$

3.3.4 Menghitung *Entropy* tiap *class* pada atribut Warna

$$Entropy(Merah)$$
$$= \left(-\left(\frac{26}{80}\right) \times \log 2\left(\frac{26}{80}\right)\right) + \left(-\left(\frac{37}{80}\right) \times \log 2\left(\frac{37}{80}\right)\right) + \left(-\left(\frac{17}{80}\right) \times \log 2\left(\frac{17}{80}\right)\right)$$
$$= 1.516327151$$
$$Entropy(Kuning)$$
$$= \left(-\left(\frac{2}{7}\right) \times \log 2\left(\frac{2}{7}\right)\right) + \left(-\left(\frac{2}{7}\right) \times \log 2\left(\frac{2}{7}\right)\right) + \left(-\left(\frac{3}{7}\right) \times \log 2\left(\frac{3}{7}\right)\right)$$
$$= 1.556656707$$

$$Entropy(Putih)$$
$$= \left(-\left(\frac{3}{13}\right) \times \log 2 \left(\frac{3}{13}\right)\right) + \left(-\left(\frac{10}{13}\right) \times \log 2 \left(\frac{10}{13}\right)\right) + \left(-\left(\frac{0}{13}\right) \times \log 2 \left(\frac{0}{13}\right)\right) = 0$$

3.3.5 Menghitung *Entropy* tiap *class* pada atribut Rasa

$$Entropy(Pulen)$$
$$= \left(-\left(\frac{14}{50}\right) \times \log 2 \left(\frac{14}{50}\right)\right) + \left(-\left(\frac{26}{50}\right) \times \log 2 \left(\frac{26}{50}\right)\right) + \left(-\left(\frac{10}{50}\right) \times \log 2 \left(\frac{10}{50}\right)\right) = 1.469182539$$

$$Entropy(Sangat - Pulen)$$
$$= \left(-\left(\frac{17}{50}\right) \times \log 2 \left(\frac{17}{50}\right)\right) + \left(-\left(\frac{23}{50}\right) \times \log 2 \left(\frac{23}{50}\right)\right) + \left(-\left(\frac{10}{50}\right) \times \log 2 \left(\frac{10}{50}\right)\right) = 1.508894705$$

3.3.6 Menghitung *Entropy* tiap *class* pada atribut Teknik

$$Entropy(Tumpang\ sari)$$
$$= \left(-\left(\frac{15}{52}\right) \times \log 2 \left(\frac{15}{52}\right)\right) + \left(-\left(\frac{28}{52}\right) \times \log 2 \left(\frac{28}{52}\right)\right) + \left(-\left(\frac{9}{52}\right) \times \log 2 \left(\frac{9}{52}\right)\right) = 1.436235453$$

$$Entropy(Tumpang\ gilir)$$
$$= \left(-\left(\frac{16}{48}\right) \times \log 2 \left(\frac{16}{48}\right)\right) + \left(-\left(\frac{21}{48}\right) \times \log 2 \left(\frac{21}{48}\right)\right) + \left(-\left(\frac{11}{48}\right) \times \log 2 \left(\frac{11}{48}\right)\right) = 1.537203882$$

3.3.7 Menghitung *Entropy* tiap *class* pada atribut Musim

$$Entropy(Kemarau)$$
$$= \left(-\left(\frac{16}{48}\right) \times \log 2 \left(\frac{16}{48}\right)\right) + \left(-\left(\frac{25}{48}\right) \times \log 2 \left(\frac{25}{48}\right)\right) + \left(-\left(\frac{7}{48}\right) \times \log 2 \left(\frac{7}{48}\right)\right) = 1.423548142$$

$$Entropy(Hujan)$$
$$= \left(-\left(\frac{15}{52}\right) \times \log 2 \left(\frac{15}{52}\right)\right) + \left(-\left(\frac{24}{52}\right) \times \log 2 \left(\frac{24}{52}\right)\right) + \left(-\left(\frac{13}{52}\right) \times \log 2 \left(\frac{13}{52}\right)\right) = 1.532205578$$

3.3.8 Menghitung *Entropy* tiap *class* pada atribut Hama

$$Entropy(Ulat\ grayak)$$
$$= \left(-\left(\frac{8}{21}\right) \times \log 2 \left(\frac{8}{21}\right)\right) + \left(-\left(\frac{10}{21}\right) \times \log 2 \left(\frac{10}{21}\right)\right) + \left(-\left(\frac{3}{21}\right) \times \log 2 \left(\frac{3}{21}\right)\right) = 1.441166544$$

$$Entropy(Tikus)$$
$$= \left(-\left(\frac{0}{18}\right) \times \log 2 \left(\frac{0}{18}\right)\right) + \left(-\left(\frac{15}{18}\right) \times \log 2 \left(\frac{15}{18}\right)\right) + \left(-\left(\frac{3}{18}\right) \times \log 2 \left(\frac{3}{18}\right)\right) = 0$$

$$Entropy(Penggerek\ batang\ jagung)$$
$$= \left(-\left(\frac{3}{19}\right) \times \log 2 \left(\frac{3}{19}\right)\right) + \left(-\left(\frac{7}{19}\right) \times \log 2 \left(\frac{7}{19}\right)\right) + \left(-\left(\frac{9}{19}\right) \times \log 2 \left(\frac{9}{19}\right)\right) = 1.461838199$$

$$Entropy(Belalang\ kembara)$$
$$= \left(-\left(\frac{20}{20}\right) \times \log 2 \left(\frac{20}{20}\right)\right) + \left(-\left(\frac{0}{20}\right) \times \log 2 \left(\frac{0}{20}\right)\right) + \left(-\left(\frac{9}{20}\right) \times \log 2 \left(\frac{9}{20}\right)\right) = 0$$

3.3.9 Menghitung *Entropy* tiap *class* pada atribut ph

$$Entropy(5.2) = \left(-\left(\frac{8}{28}\right) \times \log 2\left(\frac{8}{28}\right)\right) + \left(-\left(\frac{8}{28}\right) \times \log 2\left(\frac{8}{28}\right)\right) + \left(-\left(\frac{12}{28}\right) \times \log 2\left(\frac{12}{28}\right)\right) = 1.556656707$$

$$Entropy(5.5) = \left(-\left(\frac{5}{24}\right) \times \log 2\left(\frac{5}{24}\right)\right) + \left(-\left(\frac{13}{24}\right) \times \log 2\left(\frac{13}{24}\right)\right) + \left(-\left(\frac{6}{24}\right) \times \log 2\left(\frac{6}{24}\right)\right) = 1.450582008$$

$$Entropy(9.1) = \left(-\left(\frac{18}{48}\right) \times \log 2\left(\frac{18}{48}\right)\right) + \left(-\left(\frac{28}{48}\right) \times \log 2\left(\frac{28}{48}\right)\right) + \left(-\left(\frac{2}{48}\right) \times \log 2\left(\frac{2}{48}\right)\right) = 1.175283587$$

Table 3.3 Tabel Informasi Entropi

node	attribut	nilai	sum(nilai)	(grade-B)	(grade-C)	(grade-D)	entropi	gain
1	Variatas	hibrida	23	7	10	6	1.550494982	
		pioner	14	6	8	0	0	
		arjuna	21	7	8	6	1.575114591	
		bisi	15	4	7	5	1.54994164	
		kumala	13	3	10	0	0	
		bima	14	4	6	4	1.556656707	
	panjang	<=9	37	21	33	9	1.107096357	
		>9	63	10	15	11	1.354058564	
	bentuk	oval	70	22	36	12	1.454363793	
		pipih	14	4	6	4	1.556656707	
		kecil	16	5	7	4	1.546179692	
	warna	merah	80	26	37	17	1.516327151	
		kuning	7	2	2	3	1.556656707	
		putih	13	3	10	0	0	
	rasa	pulen	50	14	26	10	1.469182539	
		sangat-pulen	50	17	23	10	1.508894705	
	teknik	tumpang sari	52	15	28	9	1.436235453	
		tumpang gilir	48	16	21	11	1.537203882	
	musim	kemarau	48	16	25	7	1.423548142	
		hujan	52	15	24	13	1.532205578	
	hama	ulat grayak	21	8	10	3	1.441166544	
		tikus	18	0	15	3	0	
		penggerek tongkol	22	0	17	5	0	

		penggerek - batang jagung	19	3	7	9	1.461838199	
		belalang kembara	20	20	0	0	0	
	ph	5.2	28	8	8	12	1.556656707	
		5.5	24	5	13	6	1.450582008	
		9.1	48	18	28	2	1.175283587	

3.4 Menghitung *Gain Rasio*

Dimulai dari *entropy*, maka harus dihitung terlebih dahulu *gain* dari *node* akar (semua data).

$$Gain(Varietas) = 1.492461891 - \left(\left(\frac{23}{100}\right) \times 1.550494982 + \left(\frac{14}{100}\right) \times 0 + \left(\frac{21}{100}\right) \times 1.5775114591 + \left(\frac{15}{100}\right) \times 1.54994164 + \left(\frac{13}{100}\right) \times 0 + \left(\frac{14}{100}\right) \times 1.556656707\right) = 0.354650796$$

$$Gain(Panjang) = 1.492461891 - \left(\left(\frac{37}{100}\right) \times 1.107096357 + \left(\frac{63}{100}\right) \times 1.354058564\right) = 0.229779343$$

$$Gain(Bentuk) = 1.492461891 - \left(\left(\frac{70}{100}\right) \times 1.454363793 + \left(\frac{14}{100}\right) \times 1.556656707 + \left(\frac{16}{100}\right) \times 1.546179692\right) = 0.009086546$$

$$Gain(Warna) = 1.492461891 - \left(\left(\frac{80}{100}\right) \times 1.516327151 + \left(\frac{7}{100}\right) \times 1.556656707 + \left(\frac{13}{100}\right) \times 0\right) = 0.170434201$$

$$Gain(Rasa) = 1.492461891 - \left(\left(\frac{50}{100}\right) \times 1.469182539 + \left(\frac{50}{100}\right) \times 1.508894705\right) = 0.003423269$$

$$Gain(Teknik) = 1.492461891 - \left(\left(\frac{52}{100}\right) \times 1.436235453 + \left(\frac{48}{100}\right) \times 1.537203882\right) = 0.007761592$$

$$Gain(Musim) = 1.492461891 - \left(\left(\frac{48}{100}\right) \times 1.423548142 + \left(\frac{52}{100}\right) \times 1.532205578\right) = 0.012411882$$

$$Gain(Hama) = 1.492461891 - \left(\left(\frac{21}{100}\right) \times 1.441166544 + \left(\frac{18}{100}\right) \times 0 + \left(\frac{22}{100}\right) \times 0 + \left(\frac{19}{100}\right) \times 1.461838199 + \left(\frac{20}{100}\right) \times 0\right) = 0.912067659$$

$$Gain(ph) = 1.492461891 - \left(\left(\frac{28}{100}\right) \times 1.556656707 + \left(\frac{24}{100}\right) \times 1.450582008 + \left(\frac{48}{100}\right) \times 1.175283587\right) = 0.144322209$$

Table 3.4 Tabel Informasi *Gain*

node	attribut	nilai	sum(nilai)	(grade-B)	(grade-C)	(grade-D)	entropi	gain
1	Variatas	hibrida	23	7	10	6	1.550494982	

		Pioneer	14	6	8	0	0	
		Arjuna	21	7	8	6	1.575114591	
		Bisi	15	4	7	5	1.54994164	
		Kumala	13	3	10	0	0	
		Bima	14	4	6	4	1.556656707	
								0.354650796
	panjang	<=9	37	21	33	9	1.107096357	
		>9	63	10	15	11	1.354058564	
								0.229779343
	bentuk	Oval	70	22	36	12	1.454363793	
		Pipih	14	4	6	4	1.556656707	
		Kecil	16	5	7	4	1.546179692	
								0.009086546
	warna	Merah	80	26	37	17	1.516327151	
		Kuning	7	2	2	3	1.556656707	
		Putih	13	3	10	0	0	
								0.170434201
	rasa	Pulen	50	14	26	10	1.469182539	
		sangat-pulen	50	17	23	10	1.508894705	
								0.007761592
	teknik	tumpang sari	52	15	28	9	1.436235453	
		tumpang gilir	48	16	21	11	1.537203882	
								0.007761592
	musim	Kemarau	48	16	25	7	1.423548142	
		Hujan	52	15	24	13	1.532205578	
								0.007761592
	hama	ulat grayak	21	8	10	3	1.441166544	
		Tikus	18	0	15	3	0	
		penggerek tongkol	22	0	17	5	0	
		penggerek - batang jagung	19	3	7	9	1.461838199	
		belalang kembara	20	20	0	0	0	
								0.912067659
	ph	5.2	28	8	8	12	1.556656707	
		5.5	24	5	13	6	1.450582008	
		9.1	48	18	28	2	1.175283587	
								0.144322209

Dari hasil perhitungan attribut diatas nilai *gain* tertinggi adalah hama yaitu sebesar 0.912067659 dan pada *value* pengerek batang jagung memiliki nilai *entropy* yang paling tinggi sebesar 1.461838199 di bandingkan dengan nilai *entropy* lainnya, dengan demikian penggerek batang jagung menjadi cabang.

berdasarkan pembentukan pohon keputusan node 1 (*root node*), node 1.1 akan dianalisis lebih lanjut. Untuk mempermudah maka dari attribut penggerek batang jagung difilter, dengan mengambil data yang memiliki attribut **hama = penggerek batang jagung** sehingga jadilah tabel 3.5 seperti di bawah ini.

Tabel 3.5 Filter Data

no	Variatas	Panjang	Bentuk	Warna	Rasa	Teknik	Musim	Hama	PH	Mutu
1	BISI	8.9	kecil	Kuning	sangat-pulen	Tumpang Sari	Hujan	penggerek-batang jagung	5.2	Grade-D
2	Bima	10	Pipih	Merah	pulen	Tumpang Gilir	Kemarau	penggerek-batang jagung	5.5	Grade-D
3	Kumala	9.5	Oval	putih	pulen	Tumpang Gilir	Hujan	penggerek-batang jagung	9.1	Grade-B
4	Bima	9.1	Pipih	Merah	pulen	Tumpang Sari	Kemarau	penggerek-batang jagung	9.1	Grade-C
5	BISI	8.9	kecil	Kuning	sangat-pulen	Tumpang Sari	Hujan	penggerek-batang jagung	5.2	Grade-D
6	Arjuna	8.9	Oval	Merah	sangat-pulen	Tumpang Sari	Hujan	penggerek-batang jagung	5.2	Grade-D
7	BISI	9.2	kecil	Merah	sangat-pulen	Tumpang Gilir	Hujan	penggerek-batang jagung	5.2	Grade-D
8	BISI	9.5	kecil	Kuning	sangat-pulen	Tumpang Sari	Kemarau	penggerek-batang jagung	5.5	Grade-B
9	Hibrida	9.1	Oval	Merah	pulen	Tumpang Gilir	Hujan	penggerek-batang jagung	9.1	Grade-D
10	Pioner	9.5	Oval	merah	sangat-pulen	Tumpang Gilir	Kemarau	penggerek-batang jagung	9.1	Grade-B
11	Pioner	9.4	Oval	merah	sangat-pulen	Tumpang Gilir	Hujan	penggerek-batang jagung	5.2	Grade-C
12	Arjuna	7.5	Oval	Merah	sangat-pulen	Tumpang Gilir	Kemarau	penggerek-batang jagung	5.5	Grade-D
13	Pioner	9.3	Oval	merah	sangat-pulen	Tumpang Sari	Kemarau	penggerek-batang jagung	5.5	Grade-C
14	Pioner	9.3	Oval	merah	sangat-pulen	Tumpang Sari	Kemarau	penggerek-batang jagung	5.5	Grade-C
15	Hibrida	9.3	Oval	Merah	pulen	Tumpang Gilir	Kemarau	penggerek-batang jagung	9.1	Grade-D
16	Bima	9.4	Pipih	Merah	pulen	Tumpang Gilir	Hujan	penggerek-batang jagung	5.2	Grade-C
17	Arjuna	8.9	Oval	Merah	sangat-pulen	Tumpang Sari	Hujan	penggerek-batang jagung	5.2	Grade-D
18	Kumala	9.5	Oval	putih	pulen	Tumpang Sari	Kemarau	penggerek-batang jagung	5.5	Grade-C
19	Hibrida	9	Oval	Merah	pulen	Tumpang Sari	Hujan	penggerek-batang jagung	9.1	Grade-C

Dari data tersebut kemudian dianalisis dan dihitung lagi *entropy* pada atribut hama penggerek batang jagung serta setiap atribut *entropy*-nya, sehingga hasil dapat dilihat pada tabel 3.6. Setelah itu tentukan dan pilih atribut yang memiliki *gain rasio* tertinggi untuk dibuat sebagai *node* berikutnya.

$$Entropy(Penggerek\ batang\ jagung)$$
$$= \left(-\left(\frac{3}{19}\right) \times \log 2\left(\frac{3}{19}\right)\right)$$
$$+ \left(-\left(\frac{7}{19}\right) \times \log 2\left(\frac{7}{19}\right)\right)$$
$$+ \left(-\left(\frac{9}{19}\right) \times \log 2\left(\frac{9}{19}\right)\right)$$
$$= 1.461838199$$

Tabel 3.6 Filter Data

penggerek-batang jagung	grade B	grade C	grade D	Entropi
19	3	7	9	1.4618

Tahap perhitungan berikutnya ulangi proses untuk setiap cabang sampai semua kasus pada cabang memiliki kelas yang sama atau nilai *entropy* bernilai nol. Sehingga pada tabel keputusan diatas, atribut dengan nilai *Gain Ratio* tertinggi terpilih sebagai akar (*root*) dan atribut yang mempunyai nilai *Gain Ratio* lebih rendah akan menjadi cabang (*branch*) dibawah akar (*root*).

3.5 Pembahasan Hasil Eksperimen

Pada proses *K-NN imputation* dilakukan eksperimen dengan cara menentukan k=1 sampai k=5 untuk semua dataset yang dipakai dalam penelitian ini. Setelah terbentuk dataset lengkap hasil *K-NN imputation*, maka langkah selanjutnya mengolah dataset baru hasil *K-NN imputation* dengan algoritma C4.5.

Dalam parameter *K-NN imputation* yang di experimenkan adalah bagian parameter nilai K pada *k-nearest neighbor*. Hal ini dilakukan untuk mencari akurasi terbaik dalam proses klasifikasi dengan algoritma C4.5. Untuk parameter nilai K=1 sampai k=5 diimplementasikan untuk mencari nilai signifikan dalam menentukan akurasi terbaik.

Eksperimen k=1 sampai k=5 diatas digunakan untuk mengisi nilai *missing value* pada dataset mutu jagung sedangkan eksperimen klasifikasi algoritma C4.5 dilakukan dengan *criteria gain rasio* dan parameter *minimal gain* tetap sebesar 0,1 serta pada proses *pruning* tetap untuk nilai *maximal dept* 20 dan nilai *confidence* 0,25. Hasil dari eksperimen diatas ditunjukkan dalam tabel dibawah ini.

Tabel 3.7 Nilai akurasi menggunakan algoritma C4.5

METODE	AKURASI
C4.5	92.90%

Setelah mengetahui akurasi dari algoritma C4.5 dengan tanpa penangganan data *missing* tahap selanjutnya dilakukan penanganan data *missing value* dengan *K-NN imputation* pada dataset yang dipilih dengan menggunakan Algoritma C4.5 sebagai perbandingan nilai akurasi yang terbaik dari metode yang diusulkan. Tabel Nilai akurasi dataset penelitian menggunakan metode *K-NNI* dengan menentukan nilai K berbeda.

Tabel 3.8 Nilai Akurasi algoritma C4.5 + k-nn imputation

METODE		AKURASI
C4.5 + k-nn imputation	k1	93.70%
	k2	94.00%
	k3	94.30%
	k4	94.10%
	k5	94.20%

Hasil *Confution matrik* pengukuran kinerja metode C4.5 dengan penanganan data *missing value* menggunakan *K-NN imputation* ditunjukkan pada Tabel 3.9

Tabel 3.9 Akurasi confusion matriks

accuracy: 94.30%				
	True Grade-C	True Grade-D	True Grade-B	class precision
Pred. Grade-C	457	15	21	92.70%
Pred. Grade-D	4	187	7	94.44%
Pred. Grade-B	8	2	299	96.76%
class recall	97.44%	91.67%	91.44%	

Dari tabel di atas bisa disimpulkan bahwa hasil akurasi menggunakan metode C4.5 dengan penanganan data *missing value* menggunakan *K-NN imputation* adalah sebagai berikut:

a. Nilai TP (*True Positive)* dan nilai selain TP disebut dengan FN (*False Negative)*.
b. Sedangkan *class recall* merupakan kolom yang berisi besar nilai klasifikasi yang tepat. Misalnya *class recall* yang tepat pengklasifikasiannya adalah sebagai berikut:

$$TP = \frac{TP}{TP + FP} = \frac{457}{469} = 0.9744x100 = 97.44\%$$

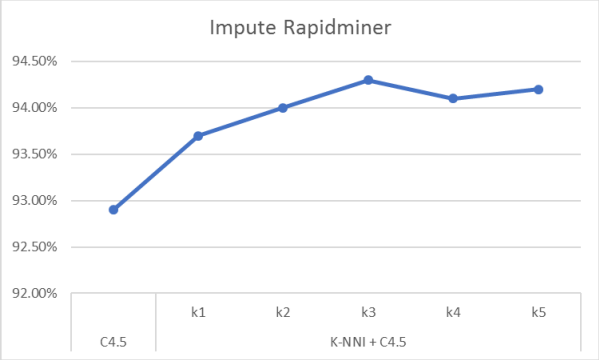
c. Sedangkan *class precision* merupakan baris yang berisi besar nilai klasifikasi yang tepat. Misalnya *class precision* yang tepat pengklasifikasiannya adalah sebagai berikut:

$$TP = \frac{TP}{TP + FN} = \frac{457}{493} = 0.927x100 = 92.70\%$$

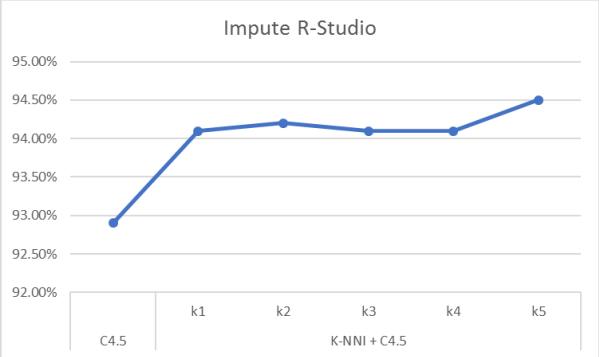
d. Sedangkan *accuracy* merupakan persentase hasil klasifikasi yang benar yang bisa didapatkan dengan cara berikut:

$$Acc = \frac{TP + TN}{TP + TN + FP + FN} = \frac{943}{1000}$$
$$= 0.943x100 = 94.30\%$$

Grafik akurasi klasifikasi perbandingan algoritma C4.5 dengan metode *K-NN imputation* untuk menangani *missing value* dengan software rapidminer dan R Studio ditampilkan dalam grafik dibawah ini.



Gambar 3.1 Grafik akurasi dengan tools rapidminer



Gambar 3.2 Grafik akurasi dengan tools R Studio

Pada grafik diatas dapat dijelaskan bahwa :

a. Pada dataset mutu jagung yang mengandung *missing value* di implementasikan pada algoritma C4.5 nilai akurasi sebesar **92.90%.**
b. Untuk *imputation value* dengan software rapidminer nilai akurasi terbaik terjadi jika dilakukan dengan penanganan khusus menggunakan metode *K-NN imputation* pada proses penanganan *missing value*-nya, dengan penentukan nilai parameter k yang berbeda yaitu nilai k=1 sebesar **93.70 %**, k=2 **s**ebesar **94.00%**, nilai k=3 sebesar **94.30%**, nilai k=4 sebesar **94.10%** dan nilai k=5 sebesar **94.20%.**
c. Sedangkan *imputation value* dengan software R Studio nilai akurasi terbaik dengan penentukan nilai parameter k yang berbeda yaitu pada nilai k=5 sebesar **94.50%** dan utuk nilai k=1, k=3 dan k=4 dengan nilai rata-rata **s**ebesar **94.10%**.

4. KESIMPULAN DAN SARAN

4.1 Kesimpulan

Berdasarkan hasil pembahasan di bab sebelumnya dapat ditarik kesimpulan bahwa :

Metode *K-NN imputation* dapat meningkatkan akurasi proses klasifikasi algoritma C4.5 pada dataset yang dipilih yaitu hasil produksi jagung, sebelumnya diterapkan langsung ke dalam algoritma C4.5 jika tidak ada penanganan data *missing* hasil akurasi sebesar 92.90%, Sedangkan nilai akurasi terbaik terjadi jika dilakukan dengan penanganan khusus menggunakan metode *K-NN imputation* pada proses penanganan *missing value*-nya nilai terbaik pada k=5 sebesar 94.50%.

4.2 Saran

Saran untuk pengembangan penelitian selanjutnya berdasarkan hasil penelitian ini adalah :

1. Untuk proses imputasi, dapat menggunakan metode lainnya, seperti *K-Means* atau Algoritma Genetika.
2. Untuk proses split atribut, dapat menggunakan metode split atribut lainnya.
3. Untuk *pruning* dapat menggunakan metode *pruning* lainnya.
4. Mengkombinasikan ketiga hal diatas, agar dapat meningkatkan akurasi.

DAFTAR PUSTAKA

[1] M. A. Bustomi and Z. Dzulfikar, "Analisis Distribusi Intensitas RGB Citra Digital untuk Klasifikasi Kualitas Biji Jagung menggunakan Jaringan Syaraf Tiruan," *Fis. Dan Apl.*, vol. 10, no. 3, pp. 127–132, 2014.

[2] L. Rokach and O. Maimon, *Data Mining With Decision Trees - Theory and Applications*. 2015.

[3] T. Wang, Z. Qin, Z. Jin, and S. Zhang, "Handling over-fitting in test cost-sensitive decision tree learning by feature selection, smoothing and pruning," *Journal of Systems and Software*, vol. 83, no. 7. pp. 1137–1147, 2010.

[4] M. Malarvizhi and A. Thanamani, "K-NN Classifier Performs Better Than K-Means Clustering in Missing Value Imputation," *IOSR J. Comput. Eng.*, vol. 6, no. 5, pp. 12–15, 2012.

[5] G. E. A. P. A. Batista and M. C. Monard, "A study of k-nearest neighbour as an imputation method," *Front. Artif. Intell. Appl.*, vol. 87, pp. 251–260, 2002.

[6] E. S. Rahayu, R. Satria, and C. Supriyanto, “Penerapan Metode Average Gain , Threshold Pruning dan Cost Complexity Pruning untuk Split Atribut pada Algoritma C4 . 5,” *J. Intell. Syst.*, vol. 1, no. 2, pp. 91–97, 2015.

[7] C. J. Mantas and J. Abellán, “Credal-C4.5: Decision tree based on imprecise probabilities to classify noisy data,” *Expert Syst. Appl.*, vol. 41, no. 10, pp. 4625–4637, 2014.

[8] Q. Song, M. Shepperd, X. Chen, and J. Liu, “Can K-NN imputation improve the performance of C4.5 with small software project data sets? A comparative evaluation,” *J. Syst. Softw.*, vol. 81, no. 12, pp. 2361–2370, 2008.

[9] D. T. Larose, *Discovering Knowledge in Data an introduction to data mining*. 2005.

[10] E. Acuña and C. Rodriguez, “The Treatment of Missing Values and its Effect on Classifier Accuracy,” *Classif. Clust. Data Min. Appl.*, no. 1995, pp. 639–647, 2004.

[11] M. Sokolova and G. Lapalme, “A systematic analysis of performance measures for classification tasks,” *Inf. Process. Manag.*, vol. 45, no. 4, pp. 427–437, 2009.