Formosa Journal of Science and Technology (FJST)
Vol. 4, No. 2, 2025: 789-808

Comparison of C4.5 and C5.0 Methods for Classification of Bad Credit and Good Credit

Bonta Zirviera Cirgon
Institute Business and Multimedia ASMI
Corresponding Author: Bonta Zirviera Cirgon, zierviera@gmail.com

A R T I C L E I N F O

Keywords: Bad credit, fintech, classification, C4.5, C5.0

Received : 30, January
Revised : 13, February
Accepted: 26, February

A B S T R A C T

PT. BZC is a startup company engaged in Fintech urging on online loans. Since the last 2 years from 2016 to 2018 PT. BZC increased compared to those who added bad loans every year. This is due to better credit analysis than repairs that increase bad loans. The suitable method for overcoming this problem is based on a literature review that is using the C4.5 and C5.0 classification methods. Therefore, it is necessary to do the C4.5 and C5.0 methods first in this study so that it can know which method is the best. The results of the development of the model using this method obtained C4.5 method reached 99% and the accuracy rate of C5.0 method reached 100%.

ISSN-E: 2964-6804
DOI: https://doi.org/10.55927/fjst.v4i2.16
https://traformosapublisher.org/index.php/fjst

INTRODUCTION

PT. BZC is a startup company engaged in Fintech (Financial Technology) focusing on online lending. Prospective customers and customers can borrow money online through applications on smartphones or on their official website. Data obtained from PT. BZC is in the form of a SQL database containing customer data from 2016 to 2018. This customer data increases every year with the following details, data from 2016 has the number of prospective customers who apply for loans of around 11,640 and of that number around 5.86% or around 682 prospective customers have their applications accepted. In that year there were customers who experienced bad credit of around 2.4% or around 16 customers. Customers who reapplied for their loans (repeat loans) were around 422 customers and of that number around 29.4% or around 124 customers experienced bad credit. Data from 2017 showed that the number of prospective customers applying for loans was around 123,092 and of that number, around 10.872% or around 13,383 prospective customers had their applications accepted. In that year, customers who experienced bad credit were greater than the previous year, around 385 customers. Customers who reapplied for their loans (repeat loans) were also greater than the previous year, around 26,275 customers and of that number, around 4.59% or around 1,206 customers experienced bad credit. Data from 2018 had the largest number of prospective customers applying for loans, around 593,963 prospective customers and of that number, around 7.9978% or around 47,502 prospective customers had their applications accepted. In that year, customers who experienced bad credit were around 2,026 customers. Customers who reapplied for their loans (repeat loans) were around 100,753 and of that number, around 15,423 customers experienced bad credit. Based on data from 2016 to 2018, it is known that there are still customers who experience bad credit.

This causes the collection division to have difficulty in carrying out the collection process so that a credit analysis needs to be carried out to find out what kind of customers are likely to experience bad credit in the future. The goal is to minimize company losses due to unreturned funds and also to make it easier for the collection division to carry out the collection process. The appropriate method to overcome this problem based on literature reviews and previous research is to use the C4.5 and C5.0 methods. In a study conducted by Revathy and Lawrance in 2017 (Revathy & Lawrance, 2017) in their study it was stated that the level of prediction accuracy of the C4.5 method was 98.48% while the level of accuracy of the C5.0 method reached 99.49% with a model creation time in the C4.5 method of 0.02 seconds, while in the C5.0 method it was 0.01 seconds, with an error rate in the C4.5 method of 1.52% while the C5.0 method was 0.51%. In the previous year, namely 2014, a study was conducted by Singh and Giri (Singh & Giri, 2014) which became the reference for Revathy and Lawrance's research, saying that the level of prediction accuracy of the C5.0 method reached 99.6%. In the previous year, namely 2012, a study was conducted by Patil, Lathi and Chitre (Patil, Lathi, & Chitre, 2012) which became the reference for Revathy and Lawrance's research, saying that the prediction accuracy of C5.0 reached 99.6%. Then the previous year's study, namely 2009

which became the reference by Patil, Lathi and Chitre, was conducted by Zhu et al. (Zhu, Wang, Yan, & Wu, 2009), that the prediction accuracy of the C5.0 method using the first dataset and 10-fold cross validation reached more than 99.8%, while using the second dataset it dropped 4% to 95%. and research conducted by Niu et al. (Niu, Zong, Yan, & Zhao, 2009), stated that the accuracy results of C4.5 reached 97.8043% by using an increase of up to 20 seconds, 40% faster than not using an increase.

Based on the literature review, it is necessary to conduct a comparison between the C4.5 method and the C5.0 method first so that the best method in this study can be identified.

THEORETICAL REVIEW

Method C4.5

C4.5 method is a series of algorithms for classification problems in a machine and data set. With variable data values, where events are described by a collection of attributes and have one of an exclusive set of classes, the C4.5 method is a mapping of attribute values to classes that can be applied to classify new invisible events (Astuti, 2016).

To select the attribute as the root in C4.5 method, based on the highest gain value of the existing attributes. To calculate the gain Equation 1 (Taufiq, Nur, Setiawan, & Bachtiar, 2018) is used:

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

Where:

S : The set of cases
A : Attribute
n : Number of attribute attributes A
$|S_i|$: Proportion of S_i to S
$|S|$: Number of cases in S

In order to obtain the highest gain value from the attribute. The attribute with the highest information gain is chosen as the test attribute of a node. While the entropy value calculation can be seen in Equation 2 (Taufiq et al., 2018):

$$Entropy(S) = \sum_{i=1}^{n} - pi * log_2 pi \quad \ldots\ldots\ldots\ldots\ldots (2)$$

Where:

S : The set of cases
n : Number of partitions S
pi : Proportion from S_i to S

In Calculation *L*og2 in formula C4.5 use the calculation rules in Equation 3 below (Rifqo & Arzi, 2016):

$$log_2(X) = \frac{\ln(x)}{\ln 2} \quad \ldots\ldots\ldots\ldots\ldots (3)$$

Method C5.0

C5.0 method is one of the data mining methods that is especially applied to the decision tree technique. C5.0 is a refinement of the previous method formed by Ross Quinlan in 1987, namely ID3 and C4.5. The C5.0 method is a commercial version of C4.5 which is widely used in many data mining packages such as Clementine and RuleQuest. The results show that C5.0 increases memory usage by around 90%, faster than C4.5 (Wirdhaningsih, Ratnawati, Malang, Mining, & Tree, 2013).

The main difference between C4.5 and C5.0 is boosting, winnowing. Boosting is a technique for generating and combining several classifiers to improve prediction accuracy. Winnowing is a feature selection step which is carried out before modeling (Septiandi, 2016).

The decision tree development strategy using the C5.0 method is as follows (W, 2007) :

1. In the initial stages, the tree is described as a single node that represents training data.
2. If the entire sample contains the same class, then the node becomes a leaf and is labeled with that class.
3. If not, the algorithm using entropy-based measures (information gain) will choose attributes that will separate the sample into individual classes. These attributes become test or decision attributes at the node.
4. Branches are developed for each known value of the test attribute, and the sample is partitioned according to that branch.
5. The algorithm uses the same process recursively forms decision trees for samples on each partition. If an attribute has appeared on a node, then the attribute does not need to be considered at the derived node.
6. The recursive partition ends only when one of the following conditions is met:
    a. All samples at a particular node have the same class.
    b. There are no attributes left in the sample which can be further partitioned. In this case the majority vote is used. These nodes become leaf nodes and are labeled with classes that make up the majority in the existing sample.
    c. There are no samples for the test attribute branch. In this case, a leaf is formed with the majority of the class labeling the sample.

C5.0 method works by splitting the sample based on variables that provide the highest information gain. Each sub sample is defined by first split and then re-split which is usually based on different variables and the process will continue to repeat until the sub sample cannot be split (W, 2007).

The information gain measure is used to select test attributes at each node in the tree. This size is used to select attributes or node in the tree. The attribute with the highest information gain value will be selected as the parent for the next node. The formula used for information gain can be seen in Equation 4 (Wirdhaningsih et al., 2013):

$$I\ (S_1\ , S_2, \dots, S_m) = \ -\ \sum_{i=1}^{m} pi * \log_2(pi) \quad \dots\dots\dots\dots\dots\ (4)$$

with:
S = case set
m = number of samples
P_i = proportion of class

S is a set consisting of s sample data. The class attribute is m where it defines the classes in it, Ci (for i = 1, ..., m), si is the number of samples in S in class Ci. To classify the sample used, information is needed using the rules as above. Where pi is the proportion of classes in output as in the class Ci and is estimated by si / s (Wirdhaningsih et al., 2013).

Attribute A has a certain value {a1, a2, ..., av}. Attribute A can be used on the S partition into v subset, {S1, S2, ..., Sv}, where Sj contains a sample of S which is worth aj on A. If A is selected as the test attribute (for example the best attribute for split), then This subset will relate to the branch of the S. Sij set node is the number of samples in the Ci class in a subset Sj. To get information on the value of the subset of attribute A, the Equation 5 is used as follows (Wirdhaningsih et al., 2013):

$$E(A) = \sum_{j=i}^{y} \frac{(S_{1j},\ldots,S_{mj})}{S} I(S_{1j}, \ldots, S_{mj}) \quad \ldots\ldots\ldots\ldots\ldots (5)$$

with: $\frac{S_{1j}+S_{mj}}{S}$ = total subset j divided by the number of samples S

To get the next gain value used the Equation 6 formula below (Wirdhaningsih et al., 2013):

$$Gain(A) = I(S_1, S_2, \ldots, S_m) - E(A) \quad \ldots\ldots\ldots\ldots\ldots (6)$$

with:
A = attribute
S = case set
S_1 = number of samples

Decision Tree

The process in decision trees is to change the shape of data (tables) into tree models, change the tree model to rules, and simplify rules (Taufiq et al., 2018). There are three types of nodes in the decision tree, namely the root node, internal node, and leaf node or terminal node (Taufiq et al., 2018).

Confusion Matrix

Confusion Matrix is a method that uses a matrix table as in Table 1 where each column in the matrix is an example of a prediction class, while each row represents an actual event in the class (Taufiq et al., 2018).

Table 1. Model Confusion Matrix

True Classification	Classified as	
	+	-
+	*True Positives* (TP)	*False Negatives* (FN)
-	*False Positives* (FP)	*True Negatives* (TN)

The Confusion Matrix represents the level of accuracy of the classification process that has been carried out. Calculation of accuracy can be seen in Equation 7 below (Taufiq et al., 2018):

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} x100 \ldots\ldots\ldots\ldots\ldots (7)$$

Confusion Matrix represents the level of accuracy of the classification process that has been done. Calculation of accuracy can be seen in Equation 8 below (Kurniawan, Kurniawan, Informatika, & Surakarta, 2018):

$$Recall = \frac{TP}{TP+FN} x100 \ldots\ldots\ldots\ldots\ldots (8)$$

The calculation of the estimated proportion of positive cases that is true or precision, is formulated in Equation 9 below (Kurniawan et al., 2018):

$$Precision = \frac{TP}{TP+FP} x100 \ldots\ldots\ldots\ldots\ldots (9)$$

K-Fold Cross Validation

K-Fold Cross Validation which is one of the methods used to determine the average success of a system by looping by randomizing input attributes so that the system is tested for some random input attributes. K-Fold Cross Validation repeats k-times to divide a set of samples randomly into k sub-sets that are mutually independent, each repetition leaving one subset for testing and the other subset for training. The use of K = 5 or 10 can be used to estimate the level of error that occurs, because the training data at each fold is quite different from the original training data. Overall, 5 or 10-fold cross validation are both recommended and agreed upon (Banjarsari, Budiman, & Farmadi, 2016).

Table 2. The model form of 10-fold Cross Validation

n-*validation*	*Dataset's Partition*
1	
2	
3	
4	
5	
6	
7	
8	

9	
10	
Data *Test*	
Data *Training*	

Hypothesis: It is suspected that the C5.0 classification method has better accuracy than the C4.5 classification method.

Based on the identification of problems, research objectives, theoretical studies, studies from previous studies, a conceptual framework for research on customer segmentation at PT. BZC can be built to predict bad credit with classification techniques as shown in Figure 1 below:

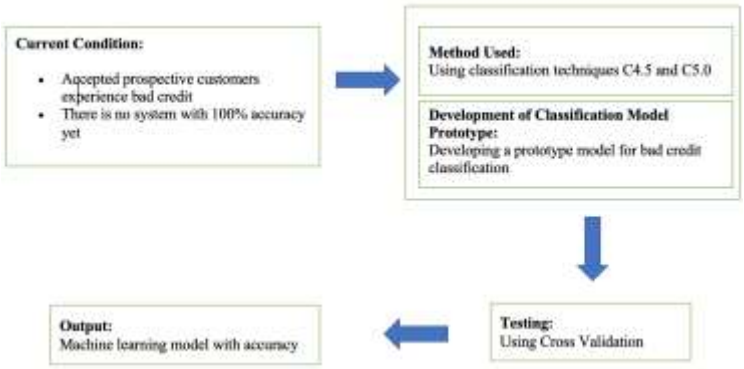


Figure 1. Conceptual Framework

METHODOLOGY

This study will compare the C4.5 method and the C5.0 method, then the results will be applied to the classification of PT. BZC customer data so that it can produce a more accurate prediction of bad credit. To find out, it is necessary to first test the C4.5 method and the C5.0 method so that we can find out the level of accuracy of the method. Based on the intent and scope of this study, this study was conducted using the experimental method. This method is carried out by researchers by manipulating conditions according to the needs of the problems faced in the study. By manipulating these conditions, the results of this study will produce a method with a higher level of accuracy than previous studies.

In this research step there are several stages as shown in the Figure 2 below:

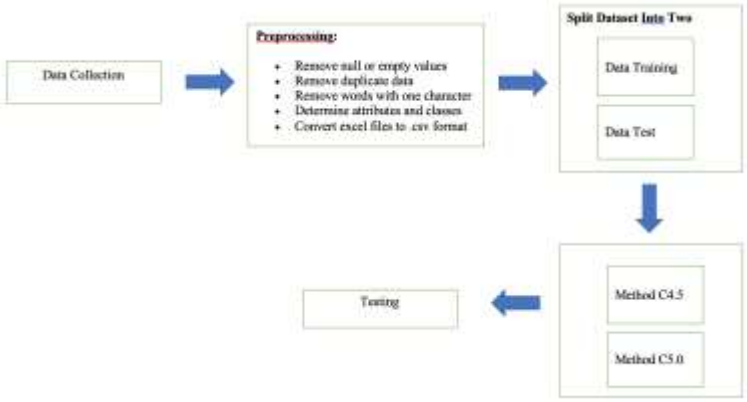


Figure 2. Research Steps

In this study, model testing was carried out, namely calculating and obtaining the rules in the C4.5 method model and the C5.0 method. The testing steps can be seen in the Figure 3 below.

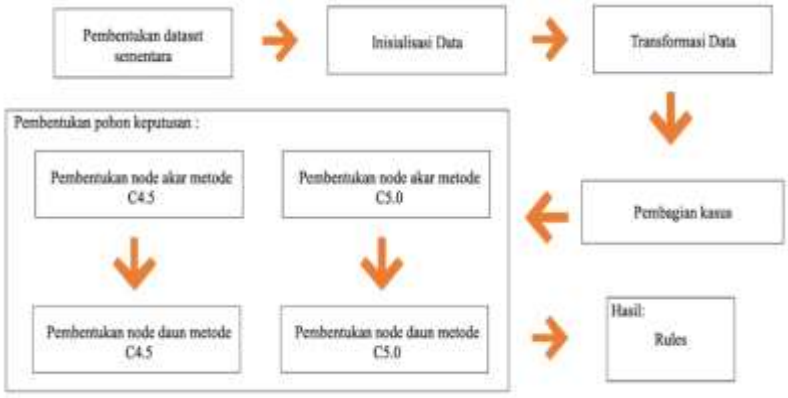


Figure 3. Manual Calculation Steps for Method C4.5 and Method C5.0

In the formation of the temporary dataset, 100 customer data in 2018 were taken as a test example can be seen in Figure 4 below:

NAMA LENGKAP	JENIS KELAMIN	UMUR	STATUS TEMPAT TINGGAL	PEMASUKAN PERBULAN	PENGELUARAN PERBULAN	ALASAN PINJAM	JUMLAH PINJAMAN	TANGGAL PINJAMAN	JATUH TEMPO	TENOR
Neni Suhertini	F	30	Rumah Sendiri	4500000.00	2500000.00	Lain-lain	1000000.00	2018-01-14	2018-02-12	30
anna nurul fathonah	F	36	Rumah Sendiri	8000000.00	4000000.00	Pendidikan	2000000.00	2018-01-17	2018-02-15	30
Eris ervanda	M	24	Rumah Orang Tua	3700000.00	2000000.00	Modal Usaha	1100000.00	2018-01-20	2018-02-18	30
abdullah	M	27	Rumah Orang Tua	3700000.00	500000.00	Lain-lain	1000000.00	2018-01-06	2018-03-08	30
Indah puspasari	F	42	Rumah Orang Tua	5000000.00	2500000.00	Pembelian Konsumen	1000000.00	2018-02-23	2018-03-24	30
Neni Suhertini	F	30	Rumah Sendiri	4500000.00	2500000.00	Modal Usaha	4000000.00	2018-02-27	2018-05-28	90
Muhamad Zamil	M	35	Rumah Orang Tua	6000000.00	5000000.00	Pembelian Konsumen	2000000.00	2018-03-02	2018-03-21	20
michael tansel	M	30	Rumah Orang Tua	30000000.00	20000000.00	Lain-lain	2000000.00	2018-03-02	2018-03-21	20
Amin Hartoko	M	35	Rumah Orang Tua	5000000.00	1000000.00	Pendidikan	1500000.00	2018-03-03	2018-03-27	25
juliandа nagoya	M	35	Rumah Sendiri	2800000.00	1000000.00	Pembelian Konsumen	1000000.00	2018-03-03	2018-04-01	30

Figure 4. Research Dataset

After the data is selected and the attributes that determine credit collectibility, the next stage is data initialization. The results of data initialization are as in Table 3 below:

Table 3. Data Initialization

Attribute	Category	Transformation
Loan Amount	1.000.000 – 3.000.000	JP1
	3.000.001 – 6.000.000	JP2
	6.000.001 – 8.000.000	JP3
Monthly Income	500.000 – 2.000.000	PMP1
	2.000.001 – 4.000.000	PMP2
	4.000.001 – 6.000.000	PMP3
	>6.000.000	PMP4
Monthly Expenditure	<3.000.000	PNP1
	>3.000.000	PNP2
Tenor	Short	T1
	Long	T2
Reasons to Borrow	Education	AP1
	Consumer Purchases	AP2
	Paying Debts	AP3
	Holiday	AP4
	Others	AP5
	Business Capital	AP6
	Medical Bills	AP7
	Motorcycle Purchase	AP8
	Vehicle Purchase	AP9
	Marriage Cost	AP10
Residence Status	Parents' Home	STT1
	Contract	STT2
	Kos	STT3
	Own Home	STT4
	Official House	STT5
Age	21-24	U1
	25-30	U2
	31-34	U3
	35-40	U4
	41-45	U5
	>45	U6
Gender	Man	JK1
	Woman	JK2

Data transformation is done by changing the numeric data type to interval and initializing the value or filling in characters that are too long on some attributes. After that, data transformation is carried out from Table 3. The results of data transformation can be seen as in Table 4 below.

Table 4. Data Transformation

Name	Loan Amount	Monthly Income	Monthly Expenditure	Age	Gender	Reasons to Borrow	Residence Status	Tenor	Collectibility
Nan Suhartini	JP2	PMP3	PNP1	U2	JK2	AP5	STT4	T1	Fluent
Anna Nurul	JP1	PMP4	PNP2	U4	JK2	AP1	STT4	T1	Fluent

Fathonah									
Eris ervanda	JP1	PMP2	PNP1	U1	JK1	AP6	STT1	T1	Fluent
abdullah	JP1	PMP2	PNP1	U2	JK1	AP5	STT1	T1	Fluent
Indah puspasari	JP1	PMP3	PNP1	U5	JK2	AP2	STT1	T1	Bad
Amin Hartoko	JP1	PMP3	PNP1	U4	JK2	AP1	STT1	T1	Fluent
Muhammad Jamil	JP1	PMP3	PNP2	U4	JK1	AP2	STT1	T1	Fluent
Michael Tansel	JP1	PMP2	PNP1	U2	JK1	AP5	STT1	T1	Fluent
…	…	…	…	…	…	…	…	…	…
Nurul Paridah	JP1	PMP3	PNP1	U2	JK2	AP7	STT1	T1	Fluent

In the decision tree method, the determination of the initial node is done through attribute calculations, entropy of all cases and gain of each attribute. The division of cases is described in Table 5 below.

Table 5. Division of Cases

Attribute	**Value**	**Sum**	**Fluent**	**Bad**
		100	63	37
Loan Amount	JP1	96	63	33
	JP2	4	0	4
Monthly Income	PMP1	1	0	1
	PMP2	33	23	10
	PMP3	40	24	16
	PMP4	26	16	10
Monthly Expenditure	PMP1	76	47	29
	PMP2	24	16	8
Tenor	T1	12	10	2
	T2	88	53	35
Reasons to Borrow	AP1	26	18	8
	AP2	13	8	3
	AP3	3	1	2
	AP4	3	2	1
	AP5	25	17	8
	AP6	19	12	7
	AP7	9	5	4
	AP8	1	0	1
	AP9	0	0	0
	AP10	1	0	1
Residence Status	STT1	64	44	20
	STT2	4	1	3
	STT3	0	0	0
	STT4	32	18	14
	STT5	0	0	0
Age	U1	31	21	10
	U2	28	21	7
	U3	22	9	13

	U4	11	11	0
	U5	2	0	2
	U6	6	1	5
Gender	JK1	55	32	13
	JK2	45	31	14

In determining the root node, it is necessary to know the gain value of each attribute. Table 6 below is the gain value of all attributes.

Table 6. Root Node Calculation

Attribute	Value	Sum	Fluent	Bad	Information Gain	Gain
		100	63	37	0.95	-
Loan Amount	JP1	96	63	33	0.9	0.08
	JP2	4	0	4	0	
Monthly Income	PMP1	1	0	1	0	0.02
	PMP2	33	23	10	0.88	
	PMP3	40	24	16	0.97	
	PMP4	26	16	10	0.97	
Monthly Expenditure	PNP1	76	47	29	0.96	0.01
	PNP2	24	16	8	0.92	
Tenor	T1	12	10	2	0.65	0.02
	T2	88	53	35	0.97	
Reasons to Borrow	AP1	26	18	8	0.89	0.04
	AP2	13	8	3	0.92	
	AP3	3	1	2	0.93	
	AP4	3	2	1	0.93	
	AP5	25	17	8	0.91	
	AP6	19	12	7	0.95	
	AP7	9	5	4	0.96	
	AP8	1	0	1	0	
	AP9	0	0	0	0	
	AP10	1	0	1	0	
Residence Status	STT1	64	44	20	3.12	-1.4
	STT2	4	1	3	0.81	
	STT3	0	0	0	0	
	STT4	32	18	14	0.99	
	STT5	0	0	0	0	
Age	U1	28	21	8	0.82	0.11
	U2	28	21	7	0.81	
	U3	22	9	13	0.98	
	U4	11	11	0	0	
	U5	2	0	2	0	
	U6	6	1	5	0.58	
Gender	JK1	55	32	13	4.16	-1.75
	JK2	45	31	14	0.89	

Based on the table above, it is known that the attribute with the highest gain is the Umur attribute, which s 0.11. Thus, the Umur attribute becomes the root node. There are six values of the UMR attribute, namely, U1 (21 - 24), U2 (25 - 30), U3 (31 - 34), U4 (35 - 40), U5 (41 - 45), U6 (> 45). From the results above, attributes U4 and U5 have an entropy value of 0 so they do not branch, but attributes U1, U2, U3 and U6 can still be recalculated. So it can be described that the temporary decision tree looks like the following Figure 5.

Figure 5. Root Node Formation

Next, recalculate the gain and entropy values using the same formula until the last node.

This study is divided into two stages, namely the training and testing stages. The training stage is used to create a classification model that is used to classify whether customers are included in the current credit or bad credit group, at this stage it will be compared so that the best method is obtained, while in the testing stage to classify whether customers are included in the bad credit group or not. Testing of both models was carried out using 10-folds Cross Validation by repeating 10 times. This can be seen in Figure 6 below:

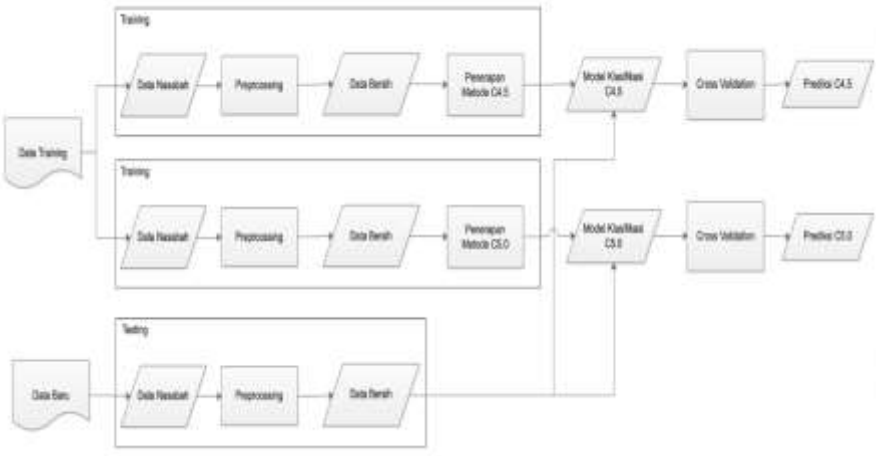


Figure 6. System Design Chart

RESULTS

Implementation Stage

At this stage, the implementation stages will be explained from the implementation of data mining to the implementation of the C4.5 method and the C5.0 method. These stages can be seen in Figure 7 below.

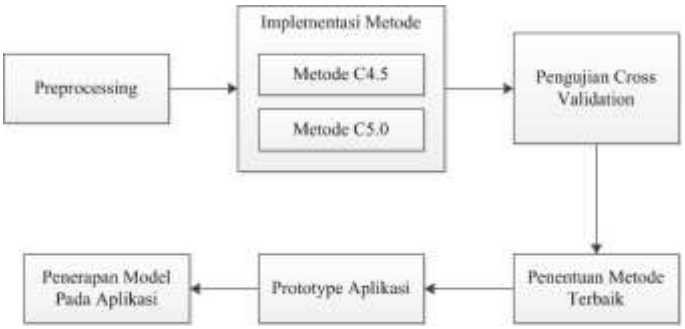


Figure 7. Implementation Stage

Pre-processing Stage

There are several stages in the pre-processing process which will be described in Figure 8 below:

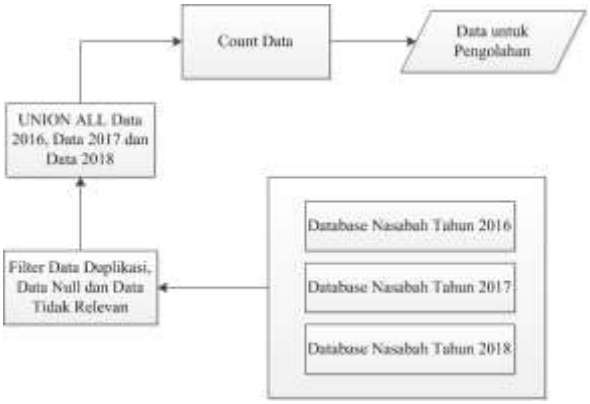


Figure 8. Data Formation Process

Based on the image above, the data formation process can be described as follows:

1. First, take customer data from 2016, customer data from 2017 and customer data from 2018 in the same database. As in Figure 9 below:

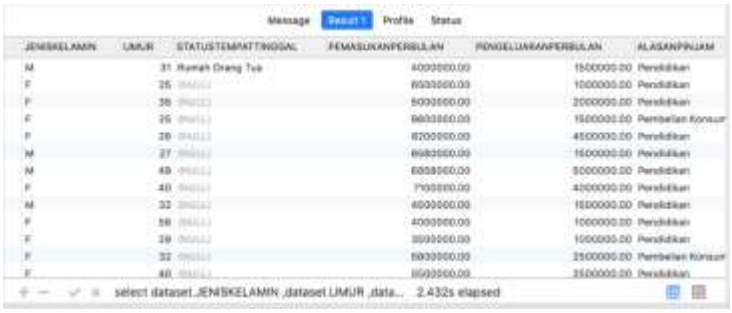


Figure 9. Data Before Cleaning

2. After obtaining the data, each duplicate data is filtered, null data and irrelevant data will be removed or not used as in Figure 10.

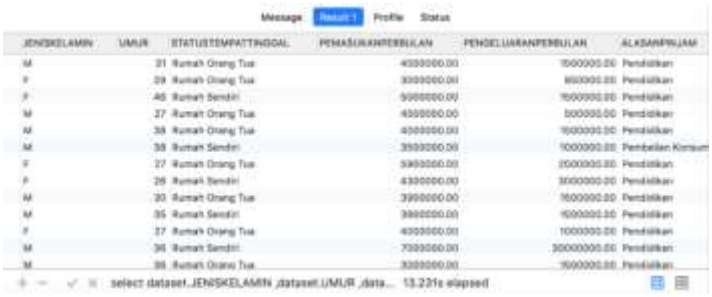


Figure 10. Data After Cleaning

3. After the data is clean, union all is performed or combining the 2016 data, 2017 data and 2018 data into one.
4. After the data has become one, the calculation of the amount of data is carried out.
5. After the process is complete, the data is ready to be processed in .csv format.

Formation of Training Data and Testing Data

At this stage is the stage of forming data into several portions consisting of 100, 1000, 5000, 10000 and 14846 as well as the formation of training data and testing data. In the formation of data, the C5.0 Method uses RStudio tools in the form of console commands, as in Figure 11 follows:

```
#Data Sample
df1 <- Dataset[order(runif(nrow(Dataset))),]
set.seed(1993)
df1.100 <- df1[1:100,]
df1.1000 <- df1[1:1000,]
df1.5000 <- df1[1:5000,]
df1.10000 <- df1[1:10000,]
df1.14846 <- df1[1:14846,]
```

Figure 11. Data Formation for C5.0 Method

Then the process of forming training data and testing data uses two methods, namely the Rapidminer tool for the C4.5 method and the RStudio tool for the C5.0 method as in Figure 12 and Figure 13:

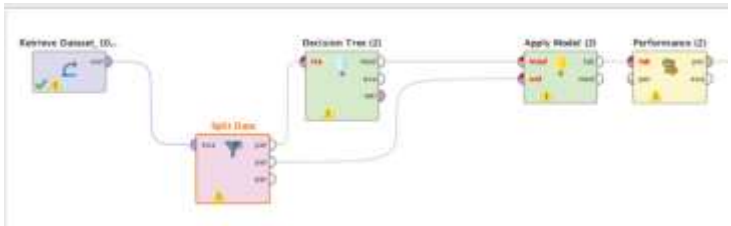


Figure 12. Formation of Training Data and Testing Data Method C4.5

Figure 13. Sharing Training Data and Testing Data Using Rapidminer

Divide the data into 80 training data and 20 testing data, then split the data by 10 while crossing the data during the Cross Validation testing process as in Figure 14 and Figure 15.

```
#split dataset into data test and data training
set.seed(1993)
#80 data training : 20 data testing
df1.train <- df1[1:11877,]
df1.test <- df1[11878:14846,]
```

Figure 14. Sharing Training Data and Testing Data Using RStudio

```
library(plyr)
set.seed(2019)
C50.folds <- split(df1.14846, cut(sample(1:nrow(df1.14846)),10))
errs.C50 <- rep(NA, length(C50.folds))
acc.C50 <- rep(NA, length(C50.folds))
pre.C50 <- rep(NA, length(C50.folds))
rec.C50 <- rep(NA, length(C50.folds))

#start_time <- Sys.time()
for (i in 1:length(C50.folds)) {
  test.C50 <- ldply(C50.folds[i], data.frame)
  train.C50 <- ldply(C50.folds[-i], data.frame)
```

Figure 15. Formation of Training Data and Testing Data in Cross Validation Method C5.0

Cross Validation Testing Method C4.5

Implement the C4.5 method by using Cross Validation cross testing then dividing the data into 80:20 for training data and testing data as in Figure 16 and Figure 17.

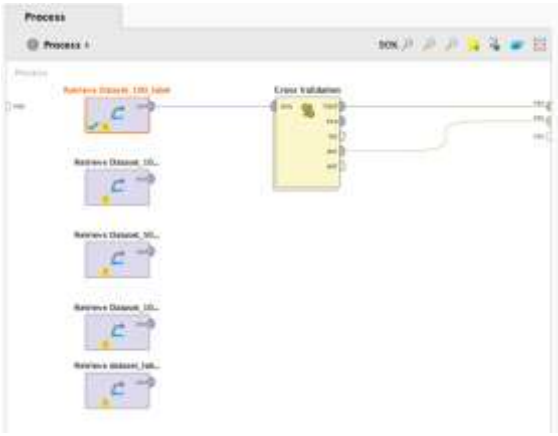


Figure 16. Testing Using Cross Validation Method C4.5

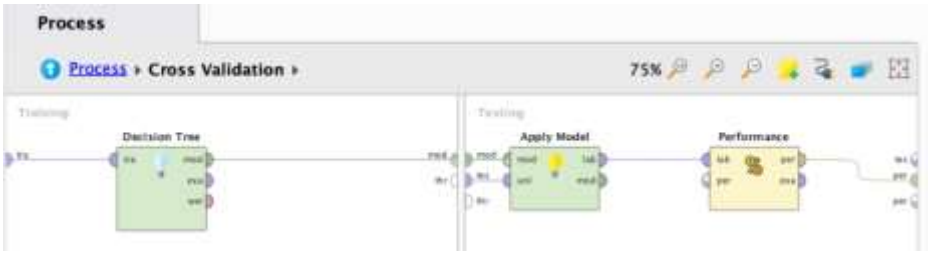


Figure 17. Cross Validation Testing Using Training Data and Testing Data Method C4.5

Test results of the C4.5 method using 10000 data samples

Based on Figure 18, it can be seen that the accuracy of the 10000 data samples reaches 95.45%, the precision reaches 95.46% and the recall reaches 99.99%.

accuracy: 95.45% +/- 0.05% (micro average: 95.45%)

	true Lancar	true Macet	class precision
pred. Lancar	9544	454	95.46%
pred. Macet	1	0	0.00%
class recall	99.99%	0.00%	

Figure 18. Test Results of Method C4.5 Sample Data 10000

Test results of the C4.5 method using 14846 data samples

Based on Figure 19, it can be seen that the accuracy of the 14846 data samples reaches 94.97%, the precision reaches 94.97% and the recall reaches 100%.

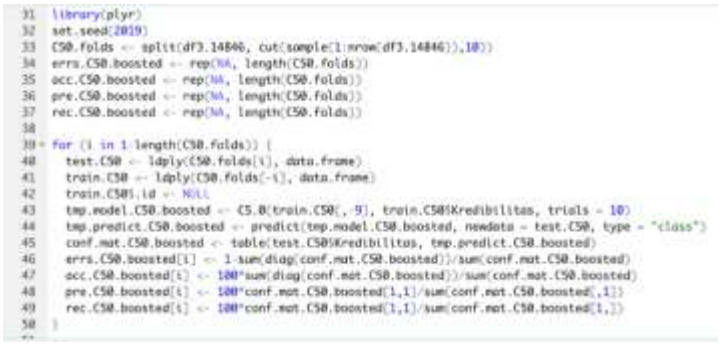
accuracy: 94.97% +/- 0.04% (micro average: 94.97%)

	true Lancar	true Macet	class precision
pred. Lancar	14115	748	94.97%
pred. Macet	0	0	0.00%
class recall	100.00%	0.00%	

Figure 19. Test Results of Method C4.5 Sample Data 14846

Cross Validation Testing Method C5.0

The process of implementing the C5.0 method with Cross Validation cross testing as shown in Figure 20.



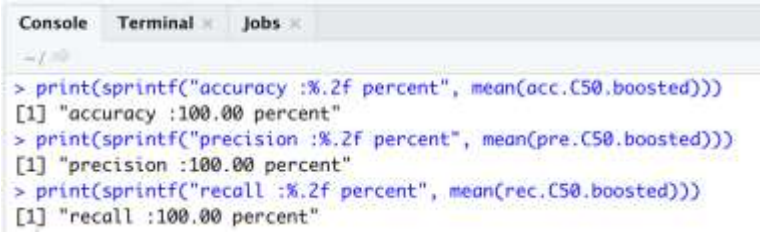
```
library(plyr)
set.seed(2019)
C50.folds <- split(df3.14846, cut(sample(1:nrow(df3.14846)),10))
errs.C50.boosted <- rep(NA, length(C50.folds))
acc.C50.boosted <- rep(NA, length(C50.folds))
pre.C50.boosted <- rep(NA, length(C50.folds))
rec.C50.boosted <- rep(NA, length(C50.folds))

for (i in 1:length(C50.folds)) {
  test.C50 <- ldply(C50.folds[i], data.frame)
  train.C50 <- ldply(C50.folds[-i], data.frame)
  train.C50$.id <- NULL
  tmp.model.C50.boosted <- C5.0(train.C50[,-9], train.C50$Kredibilitas, trials = 10)
  tmp.predict.C50.boosted <- predict(tmp.model.C50.boosted, newdata = test.C50, type = "class")
  conf.mat.C50.boosted <- table(test.C50$Kredibilitas, tmp.predict.C50.boosted)
  errs.C50.boosted[i] <- 1-sum(diag(conf.mat.C50.boosted))/sum(conf.mat.C50.boosted)
  acc.C50.boosted[i] <- 100*sum(diag(conf.mat.C50.boosted))/sum(conf.mat.C50.boosted)
  pre.C50.boosted[i] <- 100*conf.mat.C50.boosted[1,1]/sum(conf.mat.C50.boosted[,1])
  rec.C50.boosted[i] <- 100*conf.mat.C50.boosted[1,1]/sum(conf.mat.C50.boosted[1,])
}
```

Figure 20. Cross Validation Testing of C5.0 Method Using Training Data and Testing Data

Test results of the C5.0 method using 10000 data samples

Based on Figure 21, it can be seen that the accuracy of the 10000 data samples reaches 100%, the precision reaches 100% and the recall reaches 100%.

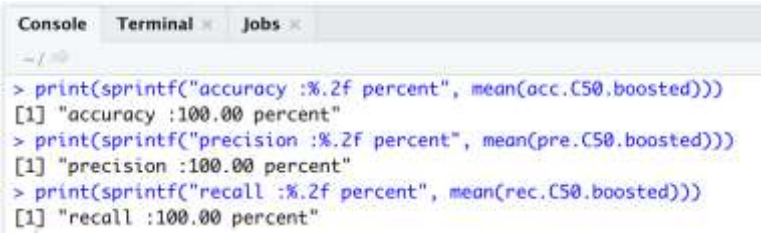


```
Console  Terminal  Jobs
> print(sprintf("accuracy :%.2f percent", mean(acc.C50.boosted)))
[1] "accuracy :100.00 percent"
> print(sprintf("precision :%.2f percent", mean(pre.C50.boosted)))
[1] "precision :100.00 percent"
> print(sprintf("recall :%.2f percent", mean(rec.C50.boosted)))
[1] "recall :100.00 percent"
```

Figure 21. Test Results of C5.0 Method Sample Data 10000

Test results of the C5.0 method using 14846 data samples

Based on Figure 22, it can be seen that the accuracy of the 14846 data samples reaches 100%, the precision reaches 100% and the recall reaches 100%.



```
Console  Terminal  Jobs
> print(sprintf("accuracy :%.2f percent", mean(acc.C50.boosted)))
[1] "accuracy :100.00 percent"
> print(sprintf("precision :%.2f percent", mean(pre.C50.boosted)))
[1] "precision :100.00 percent"
> print(sprintf("recall :%.2f percent", mean(rec.C50.boosted)))
[1] "recall :100.00 percent"
```

Figure 22. Test Results of C5.0 Method Sample Data 14846

Comparison Results

After testing the C4.5 method and the C5.0 method, based on the data in Table 7, it can be seen that the accuracy of the C5.0 method is better than the accuracy of the C4.5 method with an accuracy reaching 100%.

Table 7. Comparison Results

Metode	Sampel Data	*Accuracy (%)*	*Precision (%)*	*Recall (%)*
C4.5	100	99	98.99	100
	1000	95.70	98.99	99.69
	5000	96.22	96.22	100
	10000	95.45	95.46	99.99
	14846	94.97	94.97	100
C5.0	100	97	97	100
	1000	100	100	100
	5000	100	100	100
	10000	100	100	100
	14846	100	100	100

Implementation Model to Application

The application of the C5.0 Method model to the Bad Credit Classification Application prototype can be seen in Figure 23.

Figure 23. Result Prototype of Bad Credit Classification Application

DISCUSSION

In the comparison of Method C4.5 and Method C5.0 using two different platforms, namely Method C4.5 using Rapidminer while Method C5.0 using RStudio, this is because only RStudio has implemented Method C5.0 on its platform while on several other platforms it has not been found.

CONCLUSIONS AND RECOMMENDATIONS

Some conclusions that can be obtained from the research that has been conducted are as follows:

1. Accuracy in the C4.5 method and the C5.0 method can reach up to more than 95%.

2. The best accuracy value shows that the C5.0 method is more accurate than the C4.5 method.
3. By using the C5.0 method classification model, it can detect customer profiles that are included in the bad credit category.
4. Prototype of the bad credit classification application program designed based on the application of the decision tree model from the C5.0 method.
5. The results of this program help to analyse what customer profiles are included in the bad credit category.

FURTHER STUDY

In this study, there are limitations of the problem and there are still many shortcomings in the system developed in this study. Several things that can be used as references or considerations in further research are:

1. Further research needs to be conducted on the C5.0 method with the hope that it is not limited to accuracy and rule models.
2. Further research needs to be conducted using other datasets that may produce different accuracies.
3. From a technical programming perspective, it is necessary to develop a prototype of the bad credit classification application so that it can be connected to the database and can upload data in the form of excel format files or .csv format.

ACKNOWLEDGMENT

The author would like to thank the supervisor and colleagues who have helped and provided moral support so that this writing can be completed.

REFERENCES

Astuti, P. (2016). Komparasi Penerapan Algoritma C45, KNN dan Neural Network Dalam Proses Kelayakan Penerimaan Kredit Kendaraan Bermotor. *Faktor Exacta*, *9*(1), 87–101.

Banjarsari, M. A., Budiman, I., & Farmadi, A. (2016). Penerapan K-Optimal Pada Algoritma Knn Untuk Prediksi Kelulusan Tepat Waktu Mahasiswa Program Studi Ilmu Komputer Fmipa Unlam Berdasarkan Ip Sampai Dengan Semester 4. *Klik - Kumpulan Jurnal Ilmu Komputer*, *2*(2), 159–173. https://doi.org/10.20527/KLIK.V2I2.26

Kurniawan, D. A., Kurniawan, Y. I., Informatika, P. S., & Surakarta, U. M. (2018). Aplikasi Prediksi Kelayakan Calon Anggota Kredit Menggunakan Algoritma Naïve Bayes. *Jurnal Teknologi Informasi Dan Ilmu Komputer (JTIIK)*, *5*(4), 455–464.

Niu, Z., Zong, L., Yan, Q., & Zhao, Z. (2009). Auto-recognizing DBMS workload based on C5.0 algorithm. *Proceedings - 2009 2nd International Workshop on Knowledge Discovery and Data Mining, WKKD 2009*, 777–780. https://doi.org/10.1109/WKDD.2009.185

Patil, P. N., Lathi, R., & Chitre, V. (2012). Comparison of C5 . 0 & CART Classification algorithms using pruning technique. *International Journal of Engineering Research & Technology*, *1*(4), 1–5.

Revathy, R., & Lawrance, R. (2017). Comparative Analysis of C4.5 and C5.0 Algorithms on Crop Pest Data. *International Journal of Innovative Research in Computer and Communication Engineering*, *5*(1), 50–58. Retrieved from www.ijircce.com

Rifqo, M. H., & Arzi, T. (2016). Implementasi Algoritma C4.5 untuk Menentukan Calon Debitur dengan Mengukur Tingkat Risiko Kredit pada Bank Bri Cabang Curup. *Jurnal Pseudocode*, *III*(2), 83–90.

Septiandi, R. (2016). *Analisis Data Keterlambatan Bahan Baku Berdasarkan Pernyataan Mengenai Tesis Dan Sumber Informasi Serta Pelimpahan Hak Cipta* *.

Singh, S., & Giri, M. (2014). Comparative Study Id3, Cart And C4.5 Decision Tree Algorithm: A Survey. *International Journal of Advanced Information Science and Technology (IJAIST) ISSN*, *3*(7), 47–52. https://doi.org/10.15693/ijaist/2014.v3i7.47-52

Taufiq, I., Nur, A., Setiawan, N. Y., & Bachtiar, F. A. (2018). *Prediksi Kredit Macet Berdasarkan Preferensi Nasabah Menggunakan Metode Klasifikasi C4 . 5 pada Koperasi Simpan Pinjam Mitra Raya Wates*. *2*(12).

W, Y. Y. (2007). Perbandingan Performansi Algoritma Decision Tree C5 . 0 , Cart ,. *Seminar*, *2007*(Snati), 0–3.

Wirdhaningsih, K. P., Ratnawati, D. E., Malang, U. B., Mining, D., & Tree, D. (2013). Penerapan algoritma decision tree c5.0 untuk peramalan forex. *Doro Jurnal*, *2*(8), 1–6.

Zhu, X., Wang, J., Yan, H., & Wu, S. (2009). Research and application of the improved algorithm C4.5 on decision tree. *Proceedings of the International Symposium on Test and Measurement*, *2*, 184–187. https://doi.org/10.1109/ICTM.2009.5413078