Prediction Analysis of Kartu Jakarta Pintar (KJP) Awardees in Vocational High School XYZ Using C4.5 Algorithm

Rina Damiaza and Devi Fitrianah

***Abstract*—The purpose of this paper is to analyze data and predict students who have the potential to acquire Kartu Jakarta Pintar (KJP) scholarships. KJP is a financial assistance program from the government of Special Capital Region of Jakarta granted to Jakarta citizens who are poor to allow them to receive proper education until graduating from senior high school or vocational high school. One problem arising in its implementation is the provision of KJP that is not on target, which is due to uneven distribution of information and inaccurate decision making. To overcome this problem, it is necessary to analyze the data regarding previous KJP awardees, using data mining techniques. The method applied in the current study is a decision tree with C4.5 algorithm. Decision tree is used to help simplify the decision-making process, so that decisions can interpret the solution of the problem. Thus, based on data analysis of KJP awardees, the level of accuracy will be obtained, which proves the C4.5 algorithm can be used as an alternative method in determining prospective KJP awardees.**

***Index Terms*—Kartu Jakarta Pintar, data mining, decision Tree, C4.5 algorithm.**

I. Introduction

Every human being has the right to receive proper education. A good quality education with qualified facilities and infrastructure requires great cost to be implemented. However, not all students can afford the required costs, especially students from poor families. Therefore, financial assistance in the form of scholarships is needed to support optimal education.

Vocational High School XYZ is a school that gets a policy from the government to annually distribute scholarships in the form of Kartu Jakarta Pintar (KJP) for poor students. KJP is a financial assistance program from the government of Special Capital Region of Jakarta, granted to poor citizens of Jakarta to allow them to receive proper education until graduating from senior high school or vocational high school. During the distribution of KJP at the Vocational High School XYZ, there were several crucial problems, one of which was the provision of scholarship funds that were not on target. This is caused by the dissemination of uneven information and inaccurate decision making. Hence, Vocational High School XYZ needs help to determine eligible students to receive the KJP scholarship. Based on these problems, a prediction analysis is needed on the data of KJP awardees to support decision making.

Several previous studies have been conducted to assist decision making in the selection process of prospective scholarship awardees by predicting data from previous awardees [1], [2]. The prediction of prospective awardees can be done using classification techniques. Classification is the process of finding a model or function that describes and distinguishes data classes or concepts [3]. In 2018, A. Wibowo and D. Fitrianah used classification techniques to predict Seleksi Nasional Masuk Perguruan Tinggi Negeri (SNMPTN) acceptance for high school students in Indonesia [4]. Herein, predictions will be made on previous KJP scholarship registrant data at Vocational High School XYZ using classification techniques with a C4.5 decision tree algorithm. The contribution of the current study is to provide recommendations for determining the prospective awardees of KJP scholarships using data mining techniques.

II. Literature Review

A. Classification

Classification is a data mining technique that is used to predict classes or properties of each data instance. Prediction models make it possible to predict unknown variable values based on the values of known variables. Classification is also called supervised learning because data classes are predetermined [5]. Classification method can be used for various cases, as M. Sadikin and F. Alfiandi did in his research conducting a comparative study of classification method to predict risk potential on customer candidate data [6]. The classification process has two phases, wherein the training data are analyzed by the classification algorithm, and the second phase is classification process, wherein the test data are used to estimate the accuracy of the classification model or classifier [7]. The main components of the classification process include [1]:

1) Class, a dependent variable that represents the label of classification results.
2) Predictor, an independent variable of a model based on the characteristics of the data attributes classified.
3) Training dataset, a complete set of data that contains classes and predictors to be trained, so that the model can group into the right class.

Manuscript received May 19, 2019; revised December 12, 2019.

The authors are with the Department of Informatics Engineering, Faculty of Computer Science, Universitas Mercu Buana, Jakarta 11650, Indonesia (e-mail: 41515010088@student.mercubuana.ac.id, devi.fitrianah@mercubuana.ac.id).

4) Testing dataset, which contains new data that will be grouped by the model to determine the accuracy of the prepared model.

B. C4.5 Algorithm

The C4.5 algorithm is a well-known decision tree algorithm. The C4.5 algorithm was first introduced by Quinlan (1996) as an improved version of ID3 [8]. The decision trees generated by the C4.5 algorithm can be used for classification and for this reason, it is also referred to as a statistical classifier [9]. Classification statements on decision tree are found in its branches, and classes or segments are found in the leaves.

There are several steps in creating a decision tree using the C4.5 algorithm [2].

1) Prepare training data. Training data is usually obtained from historical data that have been grouped into certain classes.
2) Determine the root of the tree. The root will be taken from the selected attribute by calculating the gain value of each attribute; the highest gain value will be the first root. Before calculating the gain value of an attribute, the entropy value is first calculated. To calculate the entropy value, the formula is used:

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

3) Then, calculate the gain value using the formula:

$$Gain(S, A) = Entropy(S) - \sum_{i=1}^{n} \frac{|Si|}{|S|} * Entropy(Si) \quad (2)$$

4) Repeat the second step until all records are partitioned.
5) The decision tree partition process will stop when all records in N node get the same class, there is no attribute in the record that is being partitioned, and there are no records in an empty branch.

C. Decision Tree

Decision tree is a very popular technique in data mining because of its simplicity and transparency. Decision tree is a flowchart-like tree structure, where each internal node denotes a test on an attribute, each branch represents an outcome of the test, and each leaf node holds a class label [10]. Leaf nodes give a classification that applies to all instances that reach the leaf, a set of classifications, or a probability distribution over all possible classifications. To classify an unknown instance, it is routed down the tree according to the values of the attributes tested in successive nodes, and when a leaf is reached, the instance is classified according to the class assigned to the leaf [11]. C4.5 decision tree is the very first fundamental supervised machine learning classification algorithm, which is extensively implemented and typically achieves very good performance in prediction [12].

Tree complexity is influenced by several metrics, which include the total number of nodes, total number of leaves, tree depth, and total number of attributes used [13]-[16]. Usually, classification trees are graphically represented as hierarchical structures, making them easier to interpret than through other techniques. If the classification tree becomes complicated, the graphical representation becomes useless.

D. Related Works Regarding to Scholarship Recommendation

There are some previous studies related to awardee recommendation. In 2015, J. K. Alhassan and S. A. Lawal [17] applied a decision-tree-based classification technique to develop a software that would be applicable in the disbursement of scholarship for postgraduate studies. In this study, the system was tested using data from the National Information Technology Development Agency (NITDA) and proved that the classification technique based on the decision tree was able to predict whether an applicant would succeed in getting a scholarship.

Then, in 2015, Sumarlin [18] showed that the application of the k-nearest neighbor algorithm as a decision making for awardees achieved good results. In this study, cross validation, confusion matrix, and ROC curve were used to measure the performance of the k-nearest neighbor algorithm and obtained an accuracy rate of 85.56% with the area under curve (AUC) value of 0.958.

Later, in 2017, M. S. Juliardi and D. E. Cahyani [19] classified the data of Bidikmisi scholarship applicants at Sebelas Maret University using C4.5 algorithm. In this study, there are two phases of the classification method used to determine the class of data, which are the learning phase and classification phase. In the learning phase, the data of applicants from previous years is used to build a classification model, and in the classification phase, the data of applicants in the current year is used to be grouped into "accepted" or "rejected." The results show that the C4.5 algorithm can be used as an alternative method to help the selection process of Bidikmisi scholarship recipients with an accuracy rate of 79.80%.

III. Methodology

The overall methodology is shown in Fig. 1. There are several steps including data collection, pre-processing data, algorithm implementation, model testing using specified application, and evaluation and model validation.

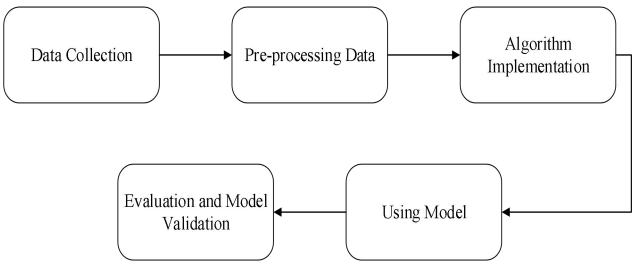

Fig. 1. Block diagram of methodology.

A. Data Collection

The dataset used in this study includes the data of 1011 KJP awardees in Vocational High School XYZ within 4 years (2015–2018). Example of the dataset, used, is shown in Table I.

B. Pre-processing Data

Pre-processing is an important step used to convert raw data into a format that allows data mining techniques to be

applied and to improve data quality [20]. The initial data obtained still have a lot of shortcomings, including many duplicates, and attributes requiring simplification. Therefore, pre-processing is needed so that the data used provide good results. In this study, pre-processing was manually done. There are two types of pre-processing used in this study, which are data cleaning and data transformation.

TABLE I: DATA COLLECTION SAMPLE

Attribute	Description	Value
1	One or both parents have died	Yes No
2	Work status of parents	Employed Unemployed
3	Parent income	Sufficient Insufficient
4	Number of family members	Int
5	Home ownership status	Owned Not Owned
6	The number of vehicles owned is more than one	Yes No
7	Decision	Accepted Rejected

1) Data Cleaning

At this stage, data cleaning from duplication is performed on the KJP scholarship registrant data each year. The results found that as many as 572 data in 2017 were duplicates of all registered data in 2016. The results of duplication checking also found 107 data in 2018 were duplicates of registrant data in 2017. Because all registrant data in 2016 have duplicates in the 2017 data, the 2016 registrant data is not used in this study.

2) Data Transformation

At this stage, the simplification of the value of each attribute is done by assigning it a code label. For example: "One or both parents have died" attribute is simplified to the "A1" code label, and "Yes" attribute is simplified to the "Y" code label. This is done to make the data easier to be processed. The conversion value of each attribute is shown in Table II.

TABLE II: DATA TRANSFORMATION

Value	Value Code
Yes	Y
No	N
Employed	Y
Unemployed	N
Sufficient	Y
Insufficient	N
Int	-
Owned	Y
Not Owned	N
Yes	Y
No	N

The dataset used after pre-processing can be seen in Table III.

TABLE III: PRE-PROCESSING DATA

Description	A1	A2	A3	A4	A5	A6	A7
Sample 1	N	Y	N	5	Y	N	Rejected
Sample 2	N	Y	N	4	N	N	Rejected
Sample 3	N	N	N	4	N	N	Accepted
Sample 4	Y	N	N	2	N	N	Accepted
Sample 5	N	Y	N	4	N	N	Accepted
Sample 6	Y	Y	N	3	N	N	Accepted
Sample 7	N	Y	N	3	N	N	Accepted
Sample 8	N	Y	Y	5	Y	Y	Rejected
Sample 9	N	Y	N	5	N	N	Rejected
Sample 10	N	Y	N	3	Y	N	Rejected
…	…	…	…	…	…	…	…
Sample 1011	N	Y	N	5	N	N	Rejected

Description:
A1 = One or both parents have died
A2 = Work status of parents
A3 = Parent income
A4 = Number of family members
A5 = Home ownership status
A6 = The number of vehicles owned is more than one
A7 = Decision

TABLE IV: THE RESULT OF ENTROPY AND INFORMATION GAIN VALUES

Node	Attribute	Value	Total	Accepted	Rejected	Entropy	Information Gain
1	Total		1011	682	329	0.910178806	
	A1						0.029887265
		Y	163	143	20	0.537070712	
		N	848	539	309	0.946264412	
	A2						0.014235446
		Y	670	421	249	0.951924367	
		N	341	261	80	0.785951352	
	A3						0.036349287
		Y	48	9	39	0.69621226	
		N	963	673	290	0.882682715	
	A4						0.047830565
		2	25	16	9	0.942683189	
		3	161	70	91	0.987692509	
		4	299	201	98	0.912620555	
		5	341	241	100	0.872889076	
		6	124	105	19	0.617854358	
		7	50	42	8	0.634309555	
		8	8	5	3	0.954434003	
		10	3	2	1	0.918295834	
	A5						0.06451356
		Y	209	82	127	0.96629588	
		N	802	600	202	0.814229083	
	A6						0.002274587
		Y	155	114	41	0.83348509	
		N	856	568	288	0.921379645	

C. Algorithm Implementation

The algorithm implementation is done by calculating *entropy* and *information gain* values. Later, the *entropy* and *information gain* values obtained are used to make the decision tree nodes. Calculating *entropy* values is done using formula (1).

$$Entropy(Total) = \left(-\frac{682}{1011} * \log_2\left(\frac{682}{1011}\right)\right) + \left(-\frac{329}{1011} * \log_2\left(\frac{329}{1011}\right)\right) = 0.910178806$$

Entropy (Total) is obtained by calculating 1011 cases with 682 "accepted" values and 329 "rejected" values. Then, the *entropy* value of each attribute value is calculated. The *information gain* value of each attribute is calculated using formula (2).

$$Gain(Total, A1) = 0.910178806 - \left(\left(\frac{163}{1011} * 0.537070712\right) + \left(\frac{848}{1011} * 0.946264412\right)\right) = 0.029887265$$

The result of *entropy* and *information gain* calculation can be seen on Table IV.

D. Using Model

After the data have been collected and processed, the model is tested using RapidMiner. The explanation of each operator used in RapidMiner is shown in Fig. 2, Fig. 3, and Fig. 4.

- Read Excel Operator: This operator is used to import data for testing from Microsoft Excel to RapidMiner. After that, the user is asked to define the spreadsheets to be used and select cells to import.
- Optimize Parameter (Grid) Operator: The optimize parameter operator is called wrapper operator because a subprocess can be built within this operator. In this case, we optimize the accuracy of the decision tree model's parameters by identifying the best combination of the *criterion* to be used for splitting of the attributes and the *minimal gain*. The *criterion* selection defines the criterion by which attributes are selected for splitting, and the *minimal gain* governs the threshold, which is requested to allow the next split.

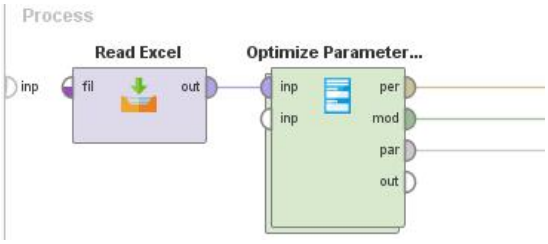

Fig. 2. Operator.

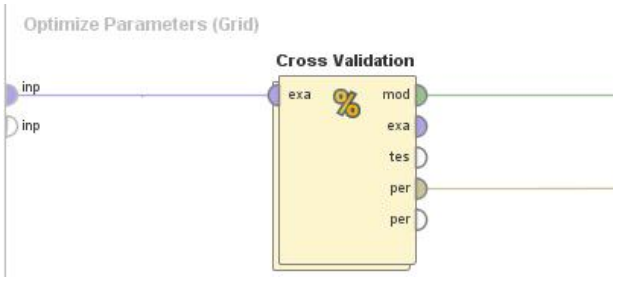

Fig. 3. Operator in optimize parameter (grid).

- Cross Validation Operator: The cross validation operator is a nesting operator that has two subprocesses, which can be embedded into it: one for training the model and one for testing the model [21]. The dataset is partitioned into *k* subsets. The process is repeated *k* times, each of the *k* subsets used once as the test data.
- Decision Tree Operator: Decision tree is a tree that is formed from a collection of nodes. This operator is used to make decisions with the C4.5 algorithm. In this operator, users can set *criterion*, *maximal depth*, *confidence*, *minimal gain*, *minimal leaf size*, *minimal size for split*, and *number of prepruning alternatives*.
- Apply Model Operator: This operator is used to apply a model that has been trained on data training. The dataset to be applied to the model must have the same attributes as the dataset used to generate the model.
- Performance Operator: The performance operator is an evaluation operator that can be used for all types of learning tasks. This operator automatically determines the type of learning task and calculates the value of the most common performance criteria for that type.

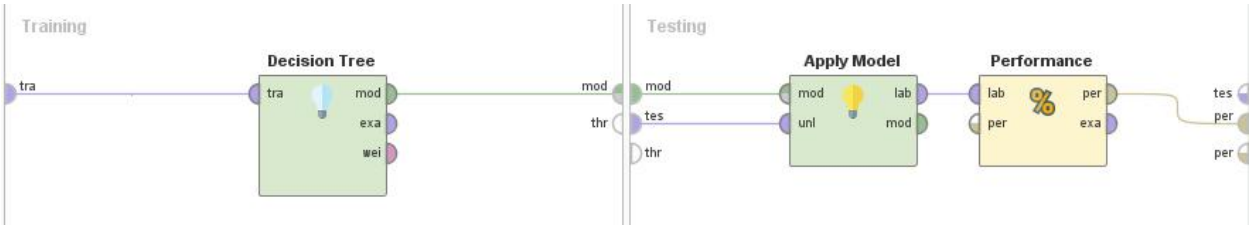

Fig. 4. Operator in cross validation.

E. Evaluation and Model Validation

Evaluation is needed to analyze and measure the accuracy of the results obtained by using confusion matrix. The calculation of confusion matrix is performed based on true positive predictions (True Positive), wrong positive predictions (False Positive), correct negative predictions (True Negative), and wrong negative predictions (False Negative) [22]. Model validation is done using the *k*-fold cross validation technique. In this technique, the dataset is randomly divided into a number of *k*-pieces; then a number of *k*-experiments are performed, wherein each experiment uses *k*-partition data as testing data, and the rest of the partition is

used as training data. In this paper, *k*-fold cross validation technique is used with a value of $k = 10$.

IV. Results and Discussion

Based on results of the calculation of *entropy* and *information gain* values shown in Table IV, A5 is selected as the root node of the decision tree with the highest *information gain* value, which is 0.06451356. To generate the next node, it is also adjusted to the highest *information gain* value by removing the previously selected attribute. A4 is selected as the next decision tree node with *information gain* value of 0.047830656, followed by A3 with *information gain* value of 0.036349287. The node selection continues until all attributes have classes. If all attributes already have a class, then a decision tree is formed as shown in Fig. 5.

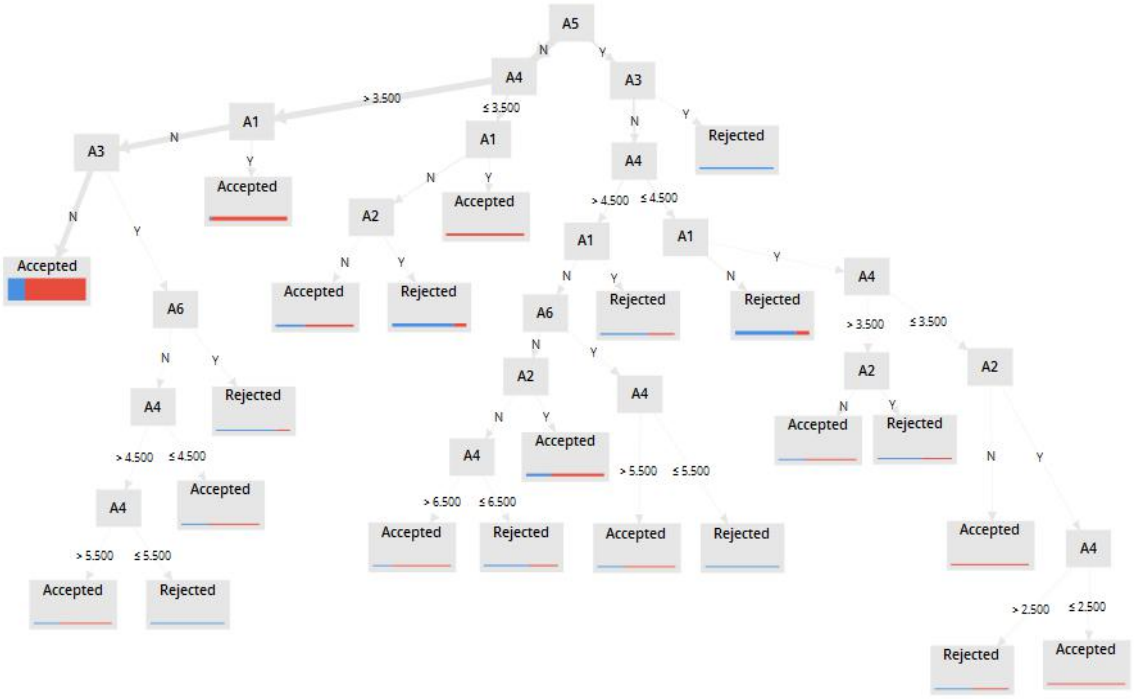

Fig. 2. Decision tree.

The decision tree that has been formed produces *rules* that are used to determine the KJP awardees recommendations. There are 22 *rules* that are formed, and can be seen as follows:

1) *If* A5 = N *And* A4 > 3.500, A1 = N, A3 = N *Then* A7 = Accepted.
2) *If* A5 = N *And* A4 > 3.500, A1 = N, A3 = Y, A6 = N, A4 > 4.500, A4 > 5.500 *Then* A7 = Accepted.
3) *If* A5 = N *And* A4 > 3.500, A1 = N, A3 = Y, A6 = N, A4 > 4.500, A4 ≤ 5.500 *Then* A7 = Rejected.
4) *If* A5 = N *And* A4 > 3.500, A1 = N, A3 = Y, A6 = N, A4 ≤ 4.500 *Then* A7 = Accepted.
5) *If* A5 = N *And* A4 > 3.500, A1 = N, A3 = Y, A6 = Y *Then* A7 = Rejected.
6) *If* A5 = N *And* A4 > 3.500, A1 = Y *Then* A7 = Accepted.
7) *If* A5 = N *And* A4 ≤ 3.500, A1 = N, A2 = N *Then* A7 = Accepted.
8) *If* A5 = N *And* A4 ≤ 3.500, A1 = N, A2 = Y *Then* A7 = Rejected.
9) *If* A5 = N *And* A4 ≤ 3.500, A1 = Y *Then* A7 = Accepted.
10) *If* A5 = Y *And* A3 = N, A4 > 4.500, A1 = N, A6 = N, A2 = N, A4 > 6.500 *Then* A7 = Accepted.
11) *If* A5 = Y *And* A3 = N, A4 > 4.500, A1 = N, A6 = N, A2 = N, A4 ≤ 6.500 *Then* A7 = Rejected.
12) *If* A5 = Y *And* A3 = N, A4 > 4.500, A1 = N, A6 = N, A2 = Y *Then* A7 = Accepted.
13) *If* A5 = Y *And* A3 = N, A4 > 4.500, A1 = N, A6 = Y, A4 > 5.500 *Then* A7 = Accepted.
14) *If* A5 = Y *And* A3 = N, A4 > 4.500, A1 = N, A6 = Y, A4 ≤ 5.500 *Then* A7 = Rejected.
15) *If* A5 = Y *And* A3 = N, A4 > 4.500, A1 = Y *Then* A7 = Rejected.
16) *If* A5 = Y *And* A3 = N, A4 ≤ 4.500, A1 = N *Then* A7 = Rejected.
17) *If* A5 = Y *And* A3 = N, A4 <= 4.500, A1 = Y, A4 > 3.500, A2 = N *Then* A7 = Accepted.
18) *If* A5 = Y *And* A3 = N, A4 ≤ 4.500, A1 = Y, A4 > 3.500, A2 = Y *Then* A7 = Rejected.
19) *If* A5 = Y *And* A3 = N, A4 ≤ 4.500, A1 = Y, A4 ≤ 3.500, A2 = N *Then* A7 = Accepted.
20) *If* A5 = Y *And* A3 = N, A4 ≤ 4.500, A1 = Y, A4 ≤ 3.500, A2 = Y, A4 > 2.500 *Then* A7 = Rejected.
21) *If* A5 = Y *And* A3 = N, A4 ≤ 4.500, A1 = Y, A4 ≤ 3.500, A2 = Y, A4 ≤ 2.500 *Then* A7 = Accepted.
22) *If* A5 = Y *And* A3 = Y *Then* A7 = Rejected.

The description of the decision tree is shown in Fig. 6.

Referring to Fig. 6, it can be known that if A5 = N and A4 > 3.500, A1 = Y, then A7 = Accepted. Similarly, if A5 = N, A4 ≤ 3,500 and A1 = Y, then A7 = Accepted. Based on these rules, it can be concluded that if the value of A5 = N and the value of A1 = Y will produce A7 with an Accepted value, i.e., if the value of A5 = N and the value of A1 = Y, then the value of the other attributes will have no effect with the results of the recommendations.

Model validation is done, using the k-fold cross validation technique with a value of $k = 10$. The model test results are shown in Fig. 7.

Tree

```
A5 = N
|   A4 > 3.500
|   |   A1 = N
|   |   |   A3 = N: Accepted {Rejected=125, Accepted=445}
|   |   |   A3 = Y
|   |   |   |   A6 = N
|   |   |   |   |   A4 > 4.500
|   |   |   |   |   |   A4 > 5.500: Accepted {Rejected=1, Accepted=2}
|   |   |   |   |   |   A4 ≤ 5.500: Rejected {Rejected=3, Accepted=0}
|   |   |   |   |   A4 ≤ 4.500: Accepted {Rejected=2, Accepted=4}
|   |   |   |   A6 = Y: Rejected {Rejected=4, Accepted=1}
|   |   A1 = Y: Accepted {Rejected=2, Accepted=80}
|   A4 ≤ 3.500
|   |   A1 = N
|   |   |   A2 = N: Accepted {Rejected=12, Accepted=20}
|   |   |   A2 = Y: Rejected {Rejected=53, Accepted=11}
|   |   A1 = Y: Accepted {Rejected=0, Accepted=37}
A5 = Y
|   A3 = N
|   |   A4 > 4.500
|   |   |   A1 = N
|   |   |   |   A6 = N
|   |   |   |   |   A2 = N
|   |   |   |   |   |   A4 > 6.500: Accepted {Rejected=1, Accepted=3}
|   |   |   |   |   |   A4 ≤ 6.500: Rejected {Rejected=6, Accepted=4}
|   |   |   |   |   A2 = Y: Accepted {Rejected=15, Accepted=33}
|   |   |   |   A6 = Y
|   |   |   |   |   A4 > 5.500: Accepted {Rejected=1, Accepted=2}
|   |   |   |   |   A4 ≤ 5.500: Rejected {Rejected=3, Accepted=0}
|   |   |   A1 = Y: Rejected {Rejected=9, Accepted=5}
|   |   A4 ≤ 4.500
|   |   |   A1 = N: Rejected {Rejected=63, Accepted=14}
|   |   |   A1 = Y
|   |   |   |   A4 > 3.500
|   |   |   |   |   A2 = N: Accepted {Rejected=2, Accepted=4}
|   |   |   |   |   A2 = Y: Rejected {Rejected=3, Accepted=2}
|   |   |   |   A4 ≤ 3.500
|   |   |   |   |   A2 = N: Accepted {Rejected=0, Accepted=9}
|   |   |   |   |   A2 = Y
|   |   |   |   |   |   A4 > 2.500: Rejected {Rejected=2, Accepted=2}
|   |   |   |   |   |   A4 ≤ 2.500: Accepted {Rejected=0, Accepted=4}
|   A3 = Y: Rejected {Rejected=22, Accepted=0}
```

Fig. 6. Decision tree description.

ParameterSet

```
Parameter set:

Performance:
PerformanceVector [
-----accuracy: 78.54% +/- 3.83% (micro average: 78.54%)
ConfusionMatrix:
True:   Rejected        Accepted
Rejected:       162     50
Accepted:       167     632
-----precision: 79.18% +/- 3.13% (micro average: 79.10%) (positive class: Accepted)
ConfusionMatrix:
True:   Rejected        Accepted
Rejected:       162     50
Accepted:       167     632
-----recall: 92.67% +/- 3.34% (micro average: 92.67%) (positive class: Accepted)
ConfusionMatrix:
True:   Rejected        Accepted
Rejected:       162     50
Accepted:       167     632
-----AUC (optimistic): 0.889 +/- 0.024 (micro average: 0.889) (positive class: Accepted)
-----AUC: 0.767 +/- 0.059 (micro average: 0.767) (positive class: Accepted)
-----AUC (pessimistic): 0.646 +/- 0.107 (micro average: 0.646) (positive class: Accepted)
]
Decision Tree.criterion = gini_index
Decision Tree.minimal_gain      = 0.01
```

Fig. 7. Performance vector.

Evaluation of test results is done manually by calculating the confusion matrix. The results of the confusion matrix calculation in the C4.5 algorithm can be seen as follows.

TABLE V: CONFUSION MATRIX

	true Rejected	true Accepted
pred. Rejected	162	50
pred. Accepted	167	632

$$\text{Accuracy} = \left(\left(\frac{162+632}{1011}\right)*100\%\right) = 78,54\%$$

$$\text{Precision} = \left(\left(\frac{632}{167+632}\right)*100\%\right) = 79,10\%$$

$$\text{Recall} = \left(\left(\frac{632}{50+632}\right)*100\%\right) = 92,67\%$$

Based on the calculations, it can be concluded that the calculation of the *accuracy*, *precision,* and *recall* rate is the same as the calculation results shown in Fig. 7. 78,54% accuracy, 79,10% precision, and 92,67% recall rates indicate that the decision tree method with the C4.5 algorithm can be used to provide recommendations of KJP awardees in Vocational High School XYZ.

V. CONCLUSION

Based on the experiments, it can be concluded that the C4.5 algorithm can be used to classify the KJP scholarship awardees. The data that used in this study are the KJP scholarship registrant data in Vocational High School XYZ. The initial data obtained cannot be directly used since they require adjustment by pre-processing. In this study, the author uses two types of pre-processing, which are data cleaning to clean the duplicate data and data transformation to simplify the value of each attribute by giving it a code.

There are two classifications produced in this study, which are "accepted" and "rejected". The evaluation result, using confusion matrix, and the validation result, using cross validation, indicate an unfavorable accuracy rate of 78.54% with 79.18% precision and 92.67% recall rates. As is well known, the C4.5 algorithm is the best algorithm that can accept all types of data. In this case, the not satisfactory value of accuracy can be due to several reasons, such as the suitability of the data and the method of pre-processing the data. Based on the result, it can be concluded that the C4.5 algorithm can be implemented on KJP scholarship registrants in Vocational High School XYZ because the method is able to predict and recommend KJP awardees.

CONFLICT OF INTEREST

The authors declare no conflict of interest.

AUTHOR CONTRIBUTIONS

All authors conducted the research. Rina Damiaza did the experiment; prepared the dataset. Devi Fitrianah supervised the implementation of algorithm using the dataset; ensure the method applied was running well. All authors analyzed the data and wrote the paper; had approved the final version.

REFERENCES

[1] D. Maurina and A. Z. Fanani. (2016). Penerapan data mining untuk rekomendasi beasiswa pada sma muhammadiyah gubug menggunakan. [Online]. pp. 1–5. Available: http://eprints.dinus.ac.id/16497/1/jurnal_15440.pdf

[2] N. Hijriana and M. Rasyidan. (2017). Penerapan metode decision tree algoritma C4.5 Untuk Seleksi. [Online]. pp. 9–13. Available: https://ojs.uniska-bjm.ac.id/index.php/JST/article/download/983/832

[3] J. Han, M. Kamber, and J. Pei, *Data Mining: Concepts and Techniques*, 2012.

[4] D. Fitrianah, H. Fahmi, A. N. Hidayanto, and A. M. Arymurthy, "A data mining based approach for determining the potential fishing zones," *Int. J. Inf. Educ. Technol.*, vol. 6, no. 3, pp. 187–191, 2016.

[5] E. R. Paramita Mayadewi, "Prediksi nilai proyek akhir mahasiswa menggunakan algoritma klasifikasi data mining," *Sist. Inf.*, vol. 11, pp. 1–7, 2015.

[6] M. Sadikin and F. Alfiandi, "Comparative study of classification

method on customer candidate data to predict its potential risk," *International Journal of Electrical and Computer Engineering*, vol. 8, no. 6, pp. 4763–4771, 2018.
[7] H. Jantan, "Human Talent Prediction in HRM using C4.5 Classification Algorithm," *International Journal on Computer Science and Engineering*, vol. 2, no. 8, pp. 2526–2534, 2010.
[8] M. H. Rifqo and T. Arzi, "Implementasi algoritma C4.5 untuk menentukan calon debitur dengan mengukur tingkat risiko kredit pada bank bri cabang curup," *J. Pseudocode*, vol. III, no. 2, pp. 83–90, 2016.
[9] K. Adhatrao, A. Gaykar, A. Dhawan, R. Jha, and V. Honrao, "Predicting Students' Performance using ID3 and C4.5 Classification Algorithms," *International Journal of Data Mining & Knowledge Management Process*, vol. 3, no. 5, pp. 39–52, 2013.
[10] G. L. Agrawal and P. H. Gupta, "Optimization of C4.5 decision tree algorithm for data mining application," *International Journal of Emerging Technology and Advanced Engineering*, vol. 3, no. 3, pp. 341–345, 2013.
[11] J. Mesarić and D. Šebalj, "Decision trees for predicting the academic success of students," *Croatian Operational Research Review*, vol. 7, pp. 367–388, 2016.
[12] R. Jothikumar and R. V. Siva Balan, "C4.5 classification algorithm with back-track pruning for accurate prediction of heart disease," *Biomed. Res.*, special issue 2, pp. S107–S111, 2016.
[13] J. R. Quinlan, "Generating production rules from decision trees," in *Proc. Tenth Int. Jt. Conf. Artif. Intell.*, vol. 30107, pp. 304–307, 1987.
[14] T. M. Mitchell, *Machine Learning*, 1997.
[15] I. H. Witten, E. Frank, and M. A. Hall, *Data Mining: Practical Machine Learning Tools and Techniques (Google eBook)*, 2011.
[16] L. Rokach and O. Maimon, *Data Mining with Decision Trees: Theory and Applications*, 2014.
[17] J. K. Alhassan and S. A. Lawal, "Using data mining technique for scholarship disbursement," *International Journal of Information and Communication Engineering*, vol. 9, no. 7, pp. 1748–1751, 2015.
[18] Sumarlin, "Implementasi algoritma k-nearest neighbor sebagai pendukung keputusan klasifikasi penerima beasiswa PPA dan BBM," *J. Sist. Inf. Bisnis*, vol. 1, pp. 52–62, 2015.
[19] M. S. Juliardi and D. E. Cahyani, "Universitas Sebelas Maret Bidikmisi Applicant's Classification using C4.5 Algorithm," *ITSMART: Jurnal Ilmiah Teknologi dan Informasi*, vol. 6, no. 1, pp. 16–23, 2017.
[20] K. Rajesh and S. Anand, "Analysis of SEER Dataset for Breast Cancer Diagnosis using C4.5 Classification Algorithm," *International Journal of Advanced Research in Computer and Communication Engineering*, vol. 1, no. 2, pp. 72–77, 2012.
[21] M. Hofmann and R. Klinkenberg, *RapidMiner: Data Mining Use Cases and Business Analytics Applications*, 2014.
[22] S. Wahyuni, K. S. S, and M. I. Perangin-angin. (2017). The implementation of decision tree algorithm C4.5 using RapidMiner in analyzing dropout students. [Online]. Available: https://www.researchgate.net/publication/324939019_THE_IMPLEMENTATION_OF_DECISION_TREE_ALGORITHM_C45_USING_RAPIDMINER_IN_ANALYZING_DROPOUT_STUDENTS

Rina Damiaza was born in Jakarta, Indonesia in 1997. She is currently pursuing the bachelor degree in informatics engineering at the Faculty of Computer Science, Universitas Mercu Buana, Indonesia.

Her research interests are data mining, machine learning and deep learning.

Devi Fitrianah was born in Jakarta, Indonesia in 1978. She received her bachelor degree in computer science from Bina Nusantara University, Indonesia in 2000 and Master degree in Information Technology from Universitas Indonesia in 2008. She received her Ph.D degree in computer science at the Faculty of Computer Science Universitas Indonesia in 2015.

At present, she is a research assistant in the Image Processing, Pattern Recognition and GIS Laboratory. Her research interests are image processing, data mining, applied remote sensing and Geographic Information System.