Journal of Computer Science and Technology Studies
ISSN: 2709-104X
DOI: 10.32996/jcsts
Journal Homepage: www.al-kindipublisher.com/index.php/jcsts

JCSTS
AL-KINDI CENTER FOR RESEARCH AND DEVELOPMENT

| RESEARCH ARTICLE

Design and Build PMB System with Prediction of Prospective Students Accepted or Withdrawal Using Random Forest Algorithm

Puteri Sejati[1] and Munawar[2]✉
[12]Faculty of Computer, Science Master's Program in Computer Science, Esa Unggul University, Indonesia
Corresponding Author: Munawar, **E-mail**: moenawar@gmail.com

| ABSTRACT
New Student Admission is one of the essential activities carried out regularly every year or semester. As the operational system of student admissions progresses, student admission data increases yearly. ESA Unggul University (UEU) has not used this data to make strategic decisions, market potential, and consider invitations to enter the academic path. So it is necessary to conduct research whose results can be used by UEU in analyzing prospective students at the time of new student admissions. In this study, data analysis was carried out from 2014 to 2019. This study aims to produce a design using the classification method to predict whether prospective students are accepted or withdrawn. In this study, 19,603 training data and 4,901 test data were used. The results showed the best Random Forest algorithm with an accuracy of 73.61%. The results of this study can be used to support the marketing department in minimizing the number of prospective students who resign.

| KEYWORDS
Prospective Students, Predictions, Design and Construction, Random Forest

| ARTICLE INFORMATION
ACCEPTED: 10 September 2022 **PUBLISHED:** 16 September 2022 **DOI:** 10.32996/jcsts.2022.4.2.8

1. Introduction

The development of information technology has developed very quickly, following the needs of the times that require speed and accuracy in all aspects of life. The development follows in terms of hardware and software, as well as the human resources that operate it. At this time, almost all areas of life require information technology, and human behaviour is accustomed to applying information technology in everyday life. With computers, we can perform data processing and data storage. You can also input data, edit data, save, delete, and others so that the data is managed more effectively and efficiently.

New student admissions (PMB) is one of the annual routine activities as a medium for recruiting prospective new students (Alviana and Kurniawan, 2019). Student admission is one of the crucial things in lecture activities where many prospective students register. However, during the registration period, there were still prospective students who did not re-register. Therefore it is necessary to do something that can make it easier for the university to analyze early on how many students will enter at the time of new student admissions.

Several previous researchers have researched the analysis of new student admissions data or the comparison of classification methods. As in the research conducted by (Aribowo and Setiadi, 2018) regarding the comparative analysis of *Data Mining* for the classification of hereditary student candidates for STMIK Widya Pratama using the KNN, *Naive Bayes,* and *Decision Tree* C4.5 methods, the accuracy level of the *Tree* C.45 algorithm was the best at 80.72%, followed by the KNN algorithm with an accuracy rate of 80.46%. At the same time, the accuracy rate of *Naive Bayes* is the lowest, at 74.49%.

Similar research was conducted (Yahya and Jananto, 2019) regarding comparing the performance of the C.45 and *Naive Bayes* to predict new student admissions activities at STIKUBANG University Semarang. In this study, the C.45 algorithm has a slightly higher accuracy value of 88.74% than the Naive Bayes algorithm, which has an accuracy of 87.24%. Research comparing classification algorithms C4.5 (*decision tree*), *Naïve Bayes*, and *Random Forest* to determine course graduation at universities was carried out by (Frastian, Hendrian, and Valentino, 2018). In this study, it is known that the accuracy value of C4.5 (*decision tree*) is 98.89%, then the *Naive Bayes* is 96.67%, and the *Random Forest* is 95.56%.

The research was conducted (Fernández-García *et al.*, 2020) regarding Recommendation systems to assist students with a choice of subjects, predict dropout risk or student performance and maximize graduation rates. In that study, one used the *Random Forest* to get an accuracy value of 72.3%. KNN got an accuracy value of 72.2%, with the best top 6 rankings.

Based on research (Aribowo and Setiadi, 2018), (Yahya and Jananto, 2019), (Frastian, Hendrian, and Valentino, 2018), and (Fernández-García *et al.*, 2020), these studies use the KNN, *Naive Bayes* and *Random Forest* because the method is mainly used in several calcification studies with good accuracy results and is one of the best algorithms. The difference in this study with previous research, namely from the use of data on new student admissions at Esa Unggul University, the features used, such as school origin, study program, registration period, religion, province, age, base (type of regular, parallel or international class), and gender.

New student admissions is a significant activity for the university. As the operational system of student admissions goes on, the data on student admissions increases yearly. The acceptance data has not been utilized by the university in strategic decision-making, marketing potential, and consideration of invitations through academic admissions. So, to assist in processing new student admissions data that can provide insight to the university regarding the percentage analysis between prospective students who will potentially become new students and those who are not new students at Esa Unggul University, in this study, an analysis of new student admissions data was carried out. Methods *Naive Bayes*, *K-Nearest Neighbor,* and *Random Forest* for classifying prospective students who did not re-register.

The goal to be achieved in this study is to be able to make a prediction model for the resignation of prospective students during the registration period. In addition, it can also find out the characteristics of the opportunities for prospective students who resign during the registration period.

The benefit of this research is to help the *marketing* of Esa Unggul University to detect early the possibility of prospective student resignations at registration and know the best classification algorithm for predicting prospective students who do not re-register.

Data mining or data mining is software used to find hidden patterns, trends, or rules contained in a large base and generate rules that are used to predict future behaviour (Utari, 2018). *Data mining* is a process of producing beneficial information from information that was previously unknown or unknown, and the knowledge gained is valuable information. It can be easily understood and understood from data. According to (Rodiyansyah and Winarko, 2013), *Data Mining* is a process of *knowledge discovery* (discovery of knowledge) from extensive data.

In *Data Mining,* there is another term with a similar meaning, namely *Knowledge Discovery in Database* (KDD). *Data Mining* and KDD have the same goal, which is to use the data available in the database by processing data to produce valuable new information (Utomo and Mesran, 2020). In Figure 2.1, the following are the stages of the *Knowledge Discovery in Database* (KDD) process.

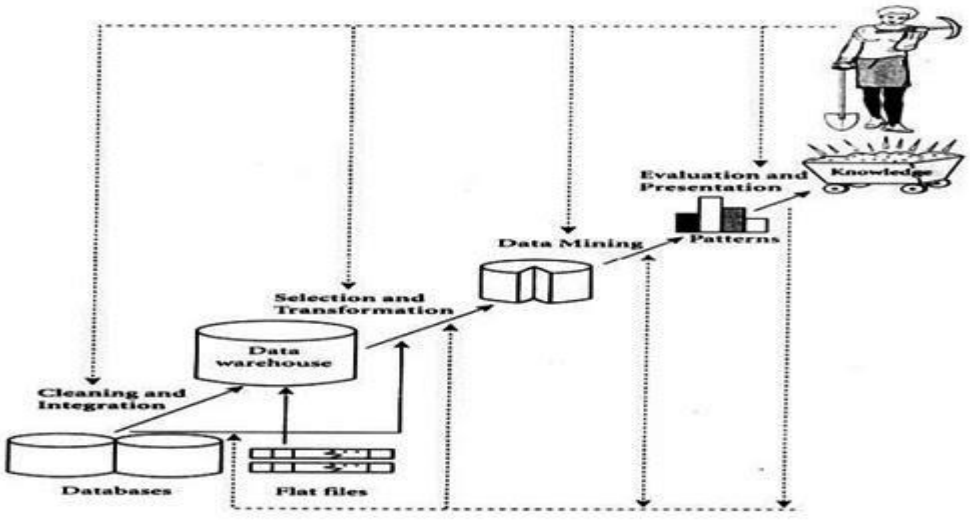


Figure 1 ***Knowledge Discovery in Database*** **(KDD)** (Utomo and Mesran, 2020)

Data Mining can be defined as the process of discovering practical and exciting knowledge in large data sets; the main objectives of *Data Mining* are prediction and description. *Data Mining* also has several main tasks, namely classification, regression, clustering, summarization, dependency modelling, change, and deviation detection (Ibrahim, Bu' logo, and Lubis, 2020). In *Data Mining*, several methods can be done, as shown in Figure 1 in this study, using the *Classification*.

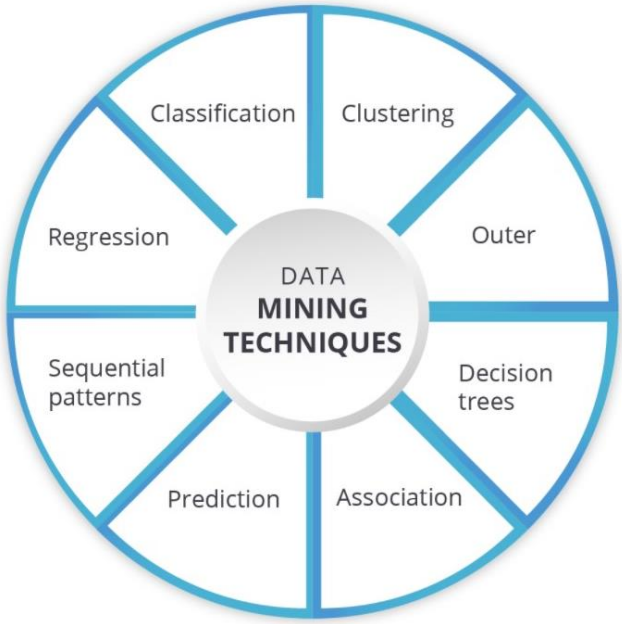


Figure 2 ***Data Mining Techniques (alternativespace.com)***

Data Understanding is the stage of understanding data starting with the initial data collection and activity results to familiarize oneself with the data to identify data problems, determine first insights into the data, or detect interesting subsets to form hypotheses for confidential information (Fadillah, 2015).

Preprocessing is a stage to improve data so that it is more structured to be processed on each *dataset*. According to (Astari, Dewa Gede Hendra Divayana, and Gede Indrawan, 2020), it is *preprocessing* processing *dataset* to prepare data before analyzing the *dataset* according to the implementation of the method used.

Process *preprocessing* is the stage where the data is processed for cleaning and preparing the data for classification (Taufik, 2018). The purpose of this process is to improve accuracy in classification modelling. In this study, the *preprocessing* used is as follows:

1. *Data Cleaning* is the process of removing *noise* and inconsistent or unnecessary data (Syarif, Anwar, and Dewiani, 2018).

2. Data transformation is a process of changing the form of data and, for example, changing numeric data into categorical data or changing from several existing variables to a new composite variable.
3. *Feature Creation* determines features or data types used in the data analysis process.
4. *Remove Outliers* are known as anomaly detection, searching for deviations from the data and identifying nonconforming data, which will then be removed (Liu *et al.*, 2021).
5. *Data Aggregation* is a process for *collapsing*, summarizing,

Classification is the process of searching for a collection of models or functions that describe and distinguish *data classes* with the aim that the model can be used to predict the *class* of an object whose class is not yet known (Putri, Ervita Kusuma & Setiadi, 2014). The model can be in the form of "if-then" rules, decision *trees*, and mathematical formulas such as *Naive Bayes* and *Support Vector Machines*.

The classification process is divided into two phases, namely, *learning* or *training* and *testing*. Phase *learning* or *training*, some of the data whose data class is known is likened to a predictive model. Because it uses previously labelled data as an example of correct data, the classification is often referred to as the *supervised method*. Then in the *testing*, the prediction model that has been formed is then tested using some other data to determine the accuracy of the model. If the accuracy is sufficient, this model can predict unknown class dates.

Naive Bayes is a simple probabilistic classifier that calculates a set of probabilities by adding up the frequency and combination of values from *the dataset* (Ibrahim, Bu'ulolo, and Lubis, 2020). An algorithm or process step that uses Bayes' theorem considers all non-interdependent attributes assigned by the value to the *class variable*. *Naive Bayes* is based on the simplifying assumption that attribute values are conditionally independent if given an output value. In other words, if given an output value, the probability of observing together is the result of the probability of each individual.

NBC (Naive Bayes Classifier) is a simple probability classification process that refers to *Bayes*. The theory states that the probability of an event occurring is equal to the intrinsic probability (calculated from currently available data) multiplied by the probability that the same thing will happen again in the future (based on previous theory or knowledge) (Mustofa and Mahfudh, 2019).

1.1 Random Forest

Random Forest is a classification and regression algorithm part of *ensemble learning* (Agustia Rahayuningsih, 2019). In the *Random Forest*, the development of the *Classification and Regression Tree* (CART) method, namely by applying the *bootstrap aggregating* (*bagging*) method and *random feature selection* (Breiman, 2001). *Random Forest* increases the diversity between classification trees by *sampling* the data with replacement and randomly changing the set of predictive variables for various tree induction methods (Dessy Kusumaningrum and Imah, 2020).

This algorithm randomly takes attributes and data *according* to the provisions in the decision tree. The decision tree consists of several nodes, namely the root node, internal node, and leaf node. The types of nodes are described as follows (Dessy Kusumaningrum and Imah, 2020):

a. Root Knot A root node has no *input* and has zero or more *output*.
b. Internal Nodes each internal node has exactly one *input* and at least two *outputs*.
c. Leaf Node (Terminal Node) Each leaf node has exactly one *input* and no *output*. The leaf nodes represent the class label.

In *Random Forest*, many trees are grown to form a forest (*forest*), and then the analysis is carried out on the group of trees. In the data cluster consisting of n observations and p explanatory variables, *Random Forest* is carried out in the following way (Breiman and Cutler, 2003):

1. Perform a random sampling of size n with recovery on the data cluster. This stage is the *bootstrap*.
2. Using the *bootstrap*, the tree is constructed until it reaches its maximum size (without pruning). At each node, the disaggregation is done by selecting m explanatory variables randomly, where m << p. The best disaggregation is selected from the m explanatory variables. This stage is the stage *of random feature selection*.
3. Repeat steps 1 and 2 k times so that a forest of k trees is formed.

Riski Annisa (2019) entitled, "Classification Algorithms *Data Mining* for Predicting Heart Disease Patients". Kaputama Journal of Informatics Engineering (JTIK). *Decision Tree, Naïve Bayes, K-Nearest Neighbor*, *Random Forest*, and *Decision Stump*. By using 10*Fold Cross Validation* and *t-test*. The results of the study obtained the highest accuracy value of 80.38%. The results show that the

Random Forest and *Decision Stump* perform best in classifiers in *the dataset*; C4.5 and *Naïve Bayes* also perform well. K-and is an algorithm that is not well implemented in *the dataset*.

2. Methodology

The conceptual framework or research stage is a form of thinking framework that can be used as an approach to solving problems. Usually, this research framework uses a scientific approach and shows the relationship between variables in the analysis process. The conceptual framework for this research can be seen in Figure 3.

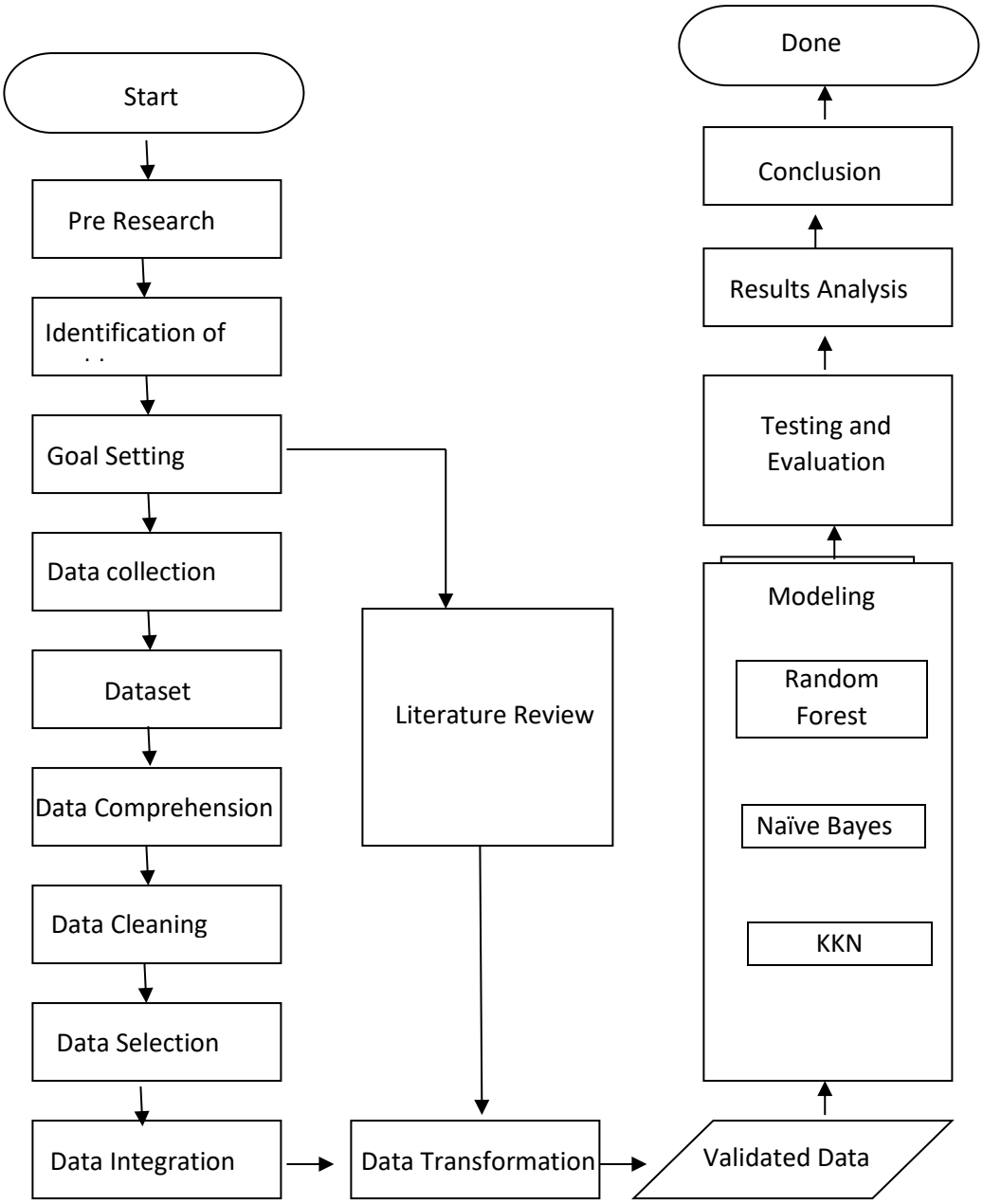


Figure 3 Research Stages

In Classifying New Student Admissions to Predict Students Who Re-register Case Study: Esa Unggul University, several stages become the leading design; this design is an overview of the process from the initial stages to the end of the system running, which is shown in Figure 4 below.

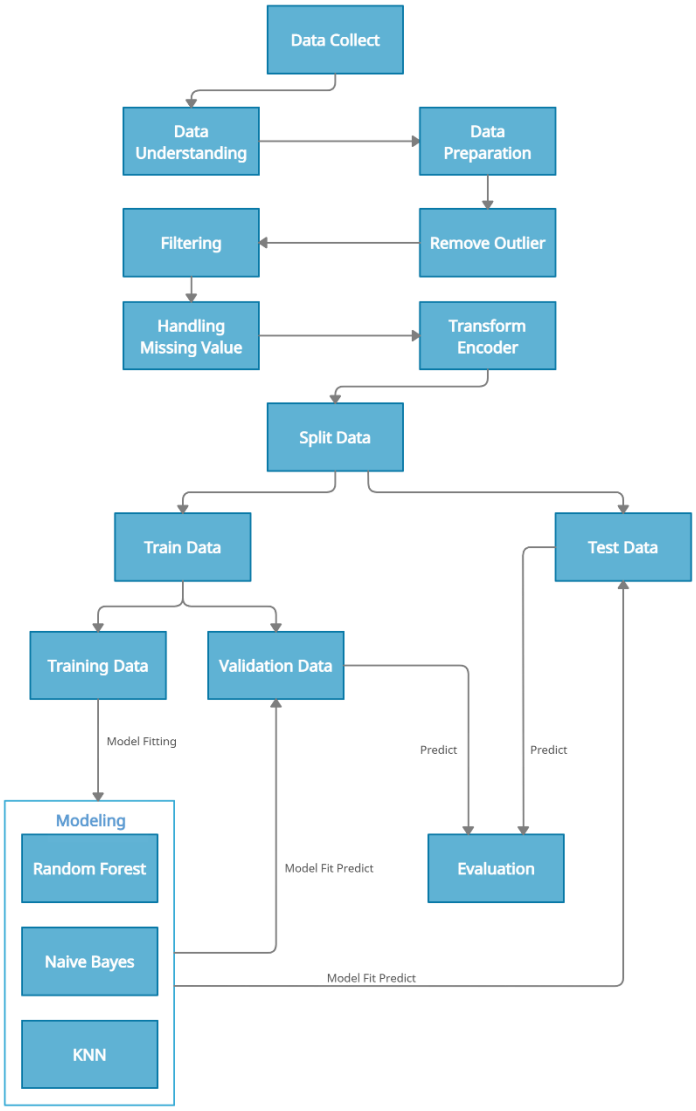


Figure 4 Application of the Method

The data used in this study is a *dataset* of new student admissions from 2015 to 2019 obtained from new student admissions at Esa Unggul University, which consists of two *datasets*, namely the *dataset* of new student registrants who entered as many as 24818 data and *the dataset* student registrar new data did not come in as many as 13315 data. Furthermore, after the *preprocessing*, the data of new student candidates who entered was 19720 and 11089 data of prospective students who did not. Sample *raw data* that has the potential to become a new student at Esa Unggul University is attached in appendix 1. This study uses several attributes, such as school origin (SMA or SMK), study program (chosen study program), and registration period (year of the registration period). , religion, province (province of domicile), age, base (regular, parallel, or international class), and gender.

3. Results and Discussion

In the Application of New Student Admissions Classification to Predict Students Who Re-register Case Study in; Esa Unggul University, there are *software* and *hardware* specifications used for implementation. This specification aims to support the system's performance designed to run correctly and get maximum results.

The software used to create this new admissions classification system is as follows:

a. Operating System Windows 10 Home (64-bit)
b. Programming Language Python 3.8.2
c. My SQL Front V6.0
d. Jupyter Notebook (Anaconda 3)

e. Google Chrome V91. 0.4472.106 (64-bit)
f. Microsoft Excel

Hardware or hardware used to create this new admissions classification system are as follows:

a. Processor : Intel®Core™ i7-10510U CPU @ 2.3GHz (64-bit)
b. RAM : 8.0 GB DDR4 2667 MHz
c. Disk : SSD 512 GB
d. VGA : Intel® UHD Graphics, NVIDIA GeForce MX 250 2.0 GB

This section discusses the stages of implementing the *Naive Bayes, K-Nearest Neighbor,* and *Random Forest* used in the admissions classification system. During the period 2015 to 2019, students who did not register. As shown in Figure 4.2, the following is a data analysis diagram of the distribution of prospective students who are not students of Esa Unggul University by province with the parameters of the number of students and province.

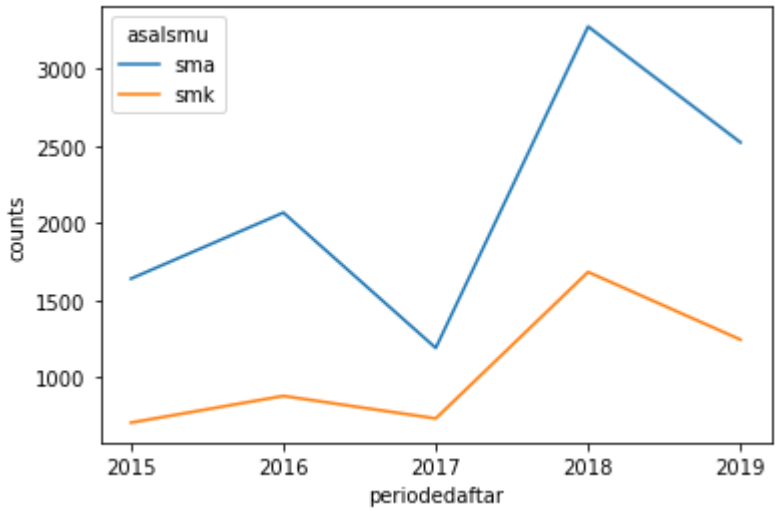


Table 5 Trends of Prospective Students Who

Re-enroll in Figure 5 is a graph of the number of prospective students who re-register with the attributes of the registration period and school level origin. Based on the graph, it can be seen that from the 2015 to 2019 period, there was an increase in 2015-2016 and 2017-2018, with a peak in 2018. Meanwhile, prospective students from the high school level were the majority who re-registered.

After going through the *Transform Encode* process, the modelling process is carried out with the data that has been done with the *data split*. In this sub-chapter, the modelling process will be carried out using the *K-Nearest Neighbor***.**

Algorithm 1 *K-Nearest Neighbor*

1. Initialize K value
2. Load *train* to model and *test* for prediction to model
3. Perform *Euclidean Distance*
4. Sort *Distance* from largest to smallest
5. If *Distance* <= K; calculate the frequency of occurrence of data on each label
6. The majority voted with the mode
7. Predictive class = class in mode

Algorithm 2 *Naive Bayes*

Load data train Initialization of features used in data
Calculation of the probability of each class against the amount of data
Calculation of the probability of each feature to the amount of data in a class
Load data to predict
Load the probability of each data feature to be predicted in each class
Multiply in each class in each feature probability and end by multiplying the probability of each class
The final result is by looking at the largest value of the likelihood value generated in the previous stage.

Algorithm 3 *Random Forest*

1. Initialize the number of trees you want to use
2. *Load* data *train* into the model
3. Perform random sampling using the *Bootstrap Aggregating*
4. training *Decision Tree* as many as initialization of the number of trees that have been determined using data that has been done *randomly*
5. Making predictions with *test* data on each model that has been trained
6. *Majority voting* with
7. class mode Prediction = class in

Test mode is one thing that needs to be done in every system development to evaluate, analyze and know the level of accuracy or similarity of results that have been achieved by the system that has been designed. Testing is done by calculating the accuracy, precision, and *recall* of predictions on data *validation* and *test*. In Figure 4.13, the following is the process of initializing the algorithm used for testing.

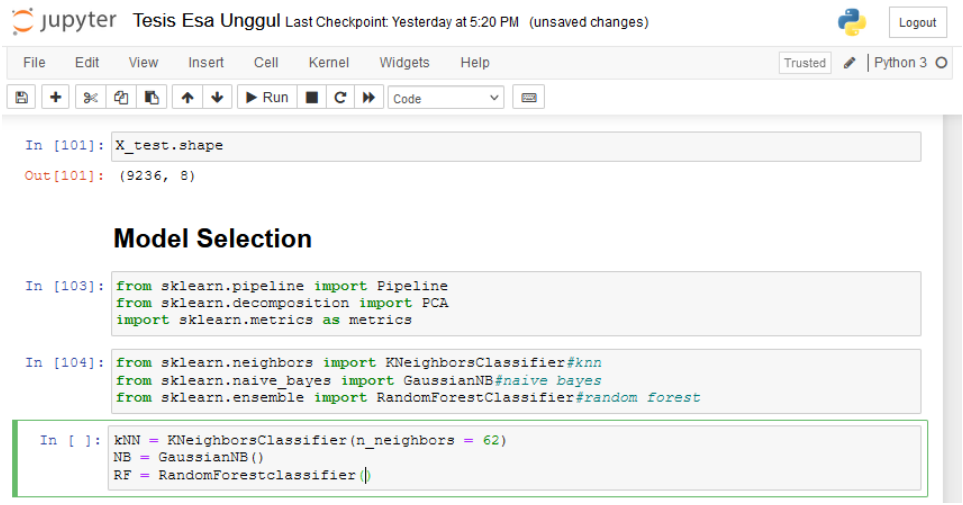


Figure 6 Algorithm Initialization Process

In Table 6, the following table contains the specifications of the method used and its parameters.

Table 3 Method Specification

No.	Method	Parameter
1	*K-Nearest Neighbor*	K = 5
2	*Random Forest*	Tree = 62
		min_sample_split = 2
		min_sample_leaf = 1
3	*Naive Bayes*	-

To calculate accuracy, precision, and *recall* using the *confusion matrix*. The following is an example of calculating accuracy, precision, and recall. In Table 4, the following is an example of a *confusion matrix* in the *Random Forest* with the amount of data *validation* used.

Table 4 Table of Testing Methods

Methods of	Data	Precision	*Recall*	Accuracy
K-Nearest Neighbor	Validation	75%	89%	75%
	Test	76%	90%	75%
Random Forest	Validation	83%	89%	82%
	Test	83%	89%	81%
Naive Bayes	Validation	73%	89 %	73%
	Test	74%	90%	73%

From the test results in Table 4, we can see that the test uses two data, namely data *validation,* and data *testing*. Data *validation* is used to determine whether the algorithm model is *overfitting* or *underfitting* compared to data *testing*. The result is that the *K-Nearest Neighbor*, *Random Forest,* and *Naïve Bayes* do not *overfit* because the comparison of accuracy between training using data *validation* and data *testing* is not too far away. The model also does not *underfit* because the comparison of accuracy between training using data *validation* and data *testing* is not too low.

Random Forest is an algorithm with the best test value in this study with accuracy using test data of 81% from the three evaluation results, namely precision, *recall*, and accuracy. Meanwhile, the other two algorithms produce a pretty good evaluation value with the second best order, the *K-Nearest Neighbor*. The algorithm with the lowest evaluation value was obtained by the *Naïve Bayes algorithm.*

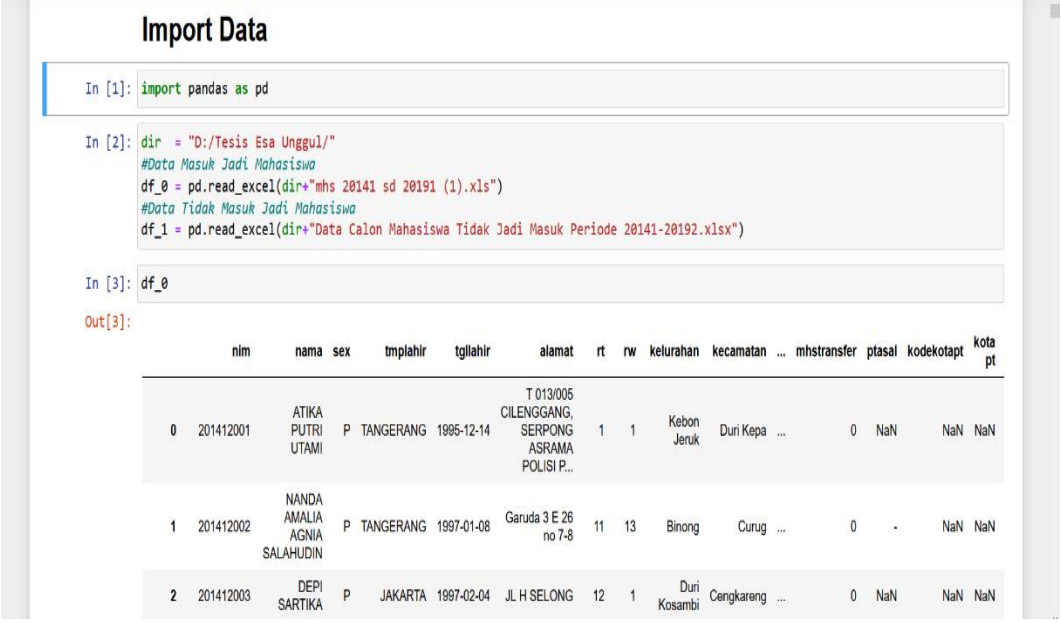
```
K-NEIGHBORS CLASSIFIER

PERFORMANCE WITH Validation DATA
Accuracy  : 75%
Precision : 75%
Recall    : 89%


RANDOM FOREST CLASSIFIER

PERFORMANCE WITH Validation DATA
Accuracy  : 82%
Precision : 83%
Recall    : 89%


GAUSSIAN NAIVE BAYES

PERFORMANCE WITH Validation DATA
Accuracy  : 73%
Precision : 73%
Recall    : 89%
```

Figure 7 Performance Results Using Data Validation

In Figure 7 below are the results of testing performance using validation data from the KNN, *Naive Bayes,* and *Random Forest* on the visualizations in Jupyter Notebook. A high accuracy value will get prediction results that are close to events in the field. Accuracy will answer the question, "What percentage of prospective students are correctly predicted to re-register or not?" A high precision value will produce a target from the training data used according to a predictable algorithm model. Precision will answer the question, "What percentage of prospective students who do not re-register from the total number of prospective students who are predicted not to re-register?" A *recall* value will produce a predictable algorithm model according to the target of the training data used. *The recall* will answer the question, "What percentage of prospective students are predicted not to re-register from all prospective students who do not actually re-register?" From the training results in Figure 7, the algorithm used is *Random Forest*.

Figure 8 is a screen display for *importing* data from prospective students of Esa Unggul University.

Figure 8 *Import* data



Import Data

```
In [1]: import pandas as pd

In [2]: dir  = "D:/Tesis Esa Unggul/"
        #Data Masuk Jadi Mahasiswa
        df_0 = pd.read_excel(dir+"mhs 20141 sd 20191 (1).xls")
        #Data Tidak Masuk Jadi Mahasiswa
        df_1 = pd.read_excel(dir+"Data Calon Mahasiswa Tidak Jadi Masuk Periode 20141-20192.xlsx")

In [3]: df_0

Out[3]:
```

	nim	nama	sex	tmplahir	tgllahir	alamat	rt	rw	kelurahan	kecamatan	...	mhstransfer	ptasal	kodekotapt	kota pt
0	201412001	ATIKA PUTRI UTAMI	P	TANGERANG	1995-12-14	T 013/005 CILENGGANG, SERPONG ASRAMA POLISI P...	1	1	Kebon Jeruk	Duri Kepa	...	0	NaN	NaN	NaN
1	201412002	NANDA AMALIA AGNIA SALAHUDIN	P	TANGERANG	1997-01-08	Garuda 3 E 26 no 7-8	11	13	Binong	Curug	...	0	-	NaN	NaN
2	201412003	DEPI SARTIKA	P	JAKARTA	1997-02-04	JL H SELONG	12	1	Duri Kosambi	Cengkareng	...	0	NaN	NaN	NaN

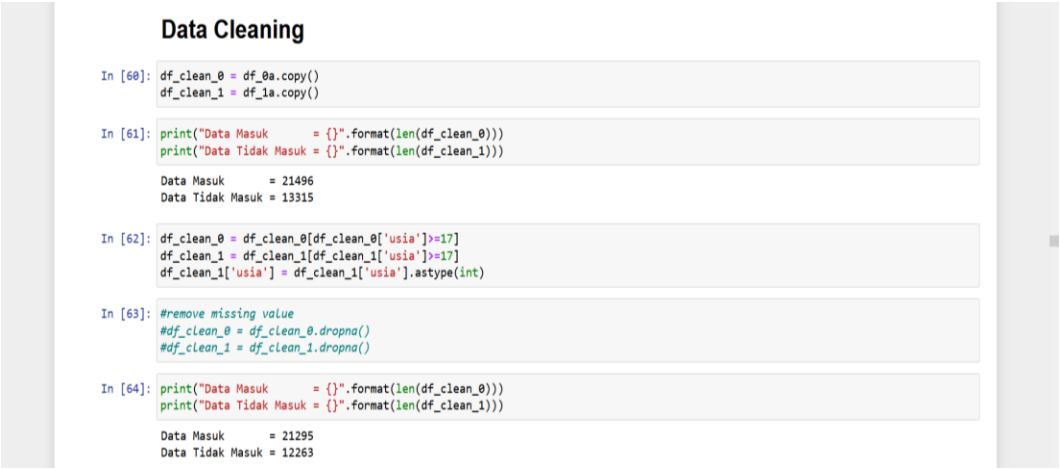
Figure 9 is a screenshot of the *preprocessing* for tidying up *the dataset*. The whole part of *preprocessing* is in Appendix 3.

Preprocessing

```
In [14]: #Copy dataframe
         df_0 = new_0.copy()
         df_1 = new_1.copy()

In [15]: #Merubah tipe data dari string ke datetime
         df_0['tgllahir'] = pd.to_datetime(new_0['tgllahir'], format='%Y-%m-%d %H:%M:%S')
         df_1['tgllahir'] = pd.to_datetime(new_0['tgllahir'], format='%Y-%m-%d %H:%M:%S')
         #Merubah tipe data dari integer ke string
         df_0['periodemasuk'] = df_0['periodemasuk'].astype(str)
         df_1['periodedaftar'] = df_1['periodedaftar'].astype(str)

In [16]: #Membuat datetime berdasarkan tahun periode daftar
         #Dasarnya adalah tanggal terakhir daftar ulang
         from datetime import datetime

         datetime_2015 = datetime.strptime('2015-09-15  00:00:00', '%Y-%m-%d %H:%M:%S')
         datetime_2016 = datetime.strptime('2016-09-15  00:00:00', '%Y-%m-%d %H:%M:%S')
         datetime_2017 = datetime.strptime('2017-09-15  00:00:00', '%Y-%m-%d %H:%M:%S')
         datetime_2018 = datetime.strptime('2018-09-15  00:00:00', '%Y-%m-%d %H:%M:%S')
         datetime_2019 = datetime.strptime('2019-09-15  00:00:00', '%Y-%m-%d %H:%M:%S')

In [17]: #Fuction merubah tanggal lahir ke usia
         from datetime import date

         def calculate_age(born,datetime):
             today = datetime
             return today.year - born.year - ((today.month, today.day) < (born.month, born.day))
```

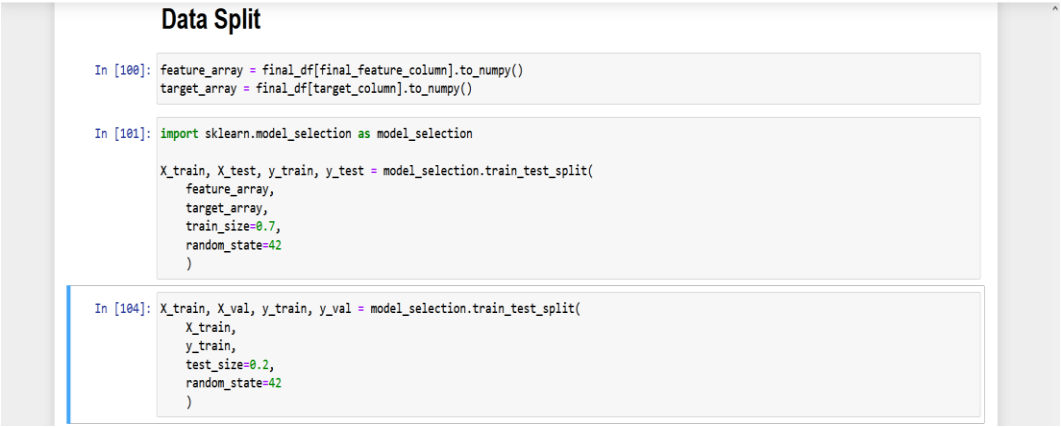
Screen Display *Preprocessing*

Screen *clean* to clean *datasets* from *noise*. The complete section of *clean* data is in Appendix 4.



Screen Display *Clean Data*

Screen *preparation* for preparing *datasets* to become *train* data and *test data*.



Screen Display *Preparation*

In Figure 12, the following is a *model selection* to test and select the best model. The whole part of *the model selection* is in Appendix 5.

Figure 12 *model selection*

In Figure 13, the following is a screenshot of the prediction sample using *testing data.* The whole part of the *testing* is in Appendix 6.

```
In [239]: result.head(10)
```

Out[239]:

	jenkel	basis	usia	provinsi	agama	periodedaftar	prodi	asalsmu	label	predicted_label
0	p	paralel	19	jawa barat	islam	20171	psikologi	sma	daftar ulang	daftar ulang
1	p	reguler	32	lampung	islam	20181	akuntansi	sma	tidak daftar ulang	tidak daftar ulang
2	p	paralel	24	dki jakarta	islam	20172	magister administrasi rumah sakit	sma	daftar ulang	daftar ulang
3	l	paralel	17	dki jakarta	islam	20172	akuntansi	smk	daftar ulang	daftar ulang
4	p	paralel	21	banten	islam	20182	teknik informatika	sma	daftar ulang	daftar ulang
5	l	paralel	20	jawa tengah	islam	20182	manajemen	sma	daftar ulang	daftar ulang
6	l	paralel	24	jawa timur	islam	20182	teknik informatika	sma	daftar ulang	daftar ulang
7	l	paralel	23	dki jakarta	islam	20191	ilmu hukum	sma	tidak daftar ulang	tidak daftar ulang
8	p	paralel	26	dki jakarta	protestan	20182	kesehatan masyarakat	sma	daftar ulang	daftar ulang
9	l	reguler	22	dki jakarta	islam	20191	teknik informatika	sma	tidak daftar ulang	tidak daftar ulang

Figure 13 Results of Sample Predictions

4. Conclusion

From the results of the discussion and process in Chapter IV using the K-Nearest Neighbor, Random Forest, and Naive Bayes to classify data on the admissions of new students at Esa Unggul University who will potentially enter and not become a student of Esa Unggul University, the conclusion is that the Random Forest gets the best performance, namely 82% accuracy, 83% precision and recall compared to the K-Nearest Neighbor and Naive Bayes. Then the K-Nearest Neighbor produces an accuracy value of 75%, a precision of 75%, and a recall of 89%. Naive Bayes produces an accuracy value of 73%, a precision of 75%, and a recall of 89%. Regarding student affairs, prospective students who come from DKI Jakarta and take a management study program have the potential not to re-register. Also, prospective students who register in the management study program have the potential to not re-register. Prospective students aged 19 to 22 years have the potential not to re-register. Moreover, prospective students from high school are more interested in studying at Esa Unggul University than prospective students from SMK.

Funding: This research received no external funding.
Conflicts of Interest: The authors declare no conflict of interest.
Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers.

References

[1] Agustia R., P. (2019) 'Comparison of Data Mining Classification Algorithms for Predicting Early Cancer Death Rates With Early Death Cancer Dataset', *Kaputama Informatics Engineering Journal (JTIK), 3*(1).
[2] Alviana, S. and Kurniawan, B. (2019) 'Analysis of New Student Admissions Data to Increase University Marketing Potential Using Business Intelligence (XYZ University Case Study)', Infotronics: *Journal of Information Technology and Electronics, 4*(1). 10–15. DOI: 10.32897/infotronik.2019.4.1.2.
[3] Annisa, R. (2019) 'Comparative Analysis of Data Mining Classification Algorithms', 3(1).
[4] Aribowo, D. and Setiadi, AEH (2018) 'Comparative Analysis of Data Mining Algorithms for the Classification of Hereditary Prospective Students of STMIK Widya Pratama', *IC-Tech, 13*(2). 1–6.
[5] Astari, NMAJ, Dewa G. H. D and Gede I (2020) 'Analysis of Twitter Document Sentiment Regarding the Impact of Corona Virus Using the Naive Bayes Classifier Method', *Journal of Systems and Informatics (JSI), 15*(1). 27–29. doi:10.30864/jsi.v15i1.332.
[6] Bode, A. (2017) 'K-Nearest Neighbor With Feature Selection Using Backward Elimination To Predict Arabica Coffee Commodity Prices', *ILKOM Scientific Journal, 9*(2). 188–195. DOI: 10.33096/Telkom.v9i2.139.188-195.
[7] Breiman, L. (2001) RFRSF: Employee Turnover Prediction Based on Random Forests and Survival Analysis, Lecture Notes in Computer Science (including the subseries Lecture Notes in Artificial Intelligence and Bioinformatics). doi:10.1007/978-3-030-62008-0_35.
[8] Breiman, L. and Cutler, A. (2003) Manual Setting Up Using and Understanding Random Forest V4.0.
[9] Dessy K and Imah, EM (2020) 'Comparative Study of Mental Workload Classification Algorithms Based on EEG Signals', *Journal of Intelligent Systems, 3*(2). 133–143. doi:10.37396/JSC.v3i2.69.
[10] Fadillah, AP (2015) 'Application of the CRISP-DM Method for Predicting Graduate Studies of Students Taking Courses (XYZ University Case Study)', *Journal of Informatics and Information Systems Engineering, 1*(3). 260–270. doi:10.28932/justice.v1i3.406.
[11] Fernández-García, A.J. (2020) 'Creating a recommender system to support higher education students in the subject enrollment decision', IEEE Access. 189069–189088. DOI: 10.1109/Access.2020.3031572.
[12] Fransiska A. K, D. (2011) 'Analysis and Implementation of Random Forests and Regression Tree (CART) for Classification in Misuse Intrusion Detection System', *Faculty of Informatics Engineering, (Data Mining*). 1–7.
[13] Frastian, N., Hendrian, S. and Valentino, V.H (2018) 'Comparison of Classification Algorithms *Determining Course Graduation at Universities*', *Exacta Factors, 11*(1), 66. doi:10.30998/factorexacta.v11i1.1826.
[14] Fu, Y. (1997) 'Data Mining, 16. 18–20.

[15] Han, J. and Kamber, M. (2000) Data Mining: Concepts and Techniques, Simon Fraser University. Morgan Kaufmann Publisher. Doi: 10.1016/0308-0161(89)90095-1.
[16] Ibrahim, M., Bu'ulolo, E. and Lubis, I. (2020) 'Application of the Naive Bayes Classifier Algorithm to Detect the Credibility Level of Hoax News/ Fake News on Android-Based Social Media in Indonesia (Case Study: Tribun Medan Office)', RESOLUTION: *Informatics and Information Engineering, 1*(1). 9–17.
[17] Liu, H. (2021) 'Clustering with Outlier Removal', *IEEE Transactions on Knowledge and Data Engineering, 33*(6). 2369–2379. doi:10.109/TKDE.2019.2954317.
[18] Mustofa, H. and Mahfudh, AA (2019) Classification of Hoax News Using the Naive Bayes Method', *Walisongo Journal of Information Technology, 1*(1). 1–12. DOI: 10.21580/wjit.2019.1.1.3915.
[19] Putri, E. K & Setiadi, T. (2014) 'Application of Text Mining in the Classification System of Spam Emails Using Naive Bayes', Application of Text Mining in the Classification System of Spam Emails Using Naive Bayes, 2(3). 73–83. doi:10.12928/justice.v2i3.2877.
[20] Rahutomo, F., Pratiwi, IYR and Ramadhani, DM (2019) 'The Naïve Bayes Experiment on Detecting Hoax News in Indonesian', *Journal of Communication Research and Public Opinion, 23*(1). DOI: 10.33299/jpkop.23.1.1805.
[21] Rodiyansyah, SF and Winarko, E. (2013) 'Twitter Post Classification of Traffic Congestion in Bandung City Using Naive Bayesian Classification', *IJCCS, 6*(1). 91–100.
[22] Rohman, A. (2012) 'K-Nearest Neighbor (KNN) Algorithm Model for Predicting Student Graduation', *Scientific Journal of Technology.*
[23] Rosandy, T. (2016) 'Comparison of the Naive Bayes Classifier Method with the Decision Tree Method (C4.5) for Analyzing Financing Smoothness (Case Study: KSPPS / BMT AL-FADHILA', *Journal of Information Technology Magister Darmajaya, 2(01)* 52–62.
[24] Saleh, A. (2015) 'Implementation of the Naïve Bayes Classification Method in Predicting the Amount of Household Electricity Use', *Creative Information Technology Journal, 2*(3). 207–217.
[25] Suprianto, S. (2020) 'Implementation of the Naive Bayes Algorithm to Determine Strategic Locations in Opening Medium and Lower Businesses in Medan City (Case Study: Disperindag Medan City)', *Journal of Computer Systems and Information Technology (JSON), 1(*2). 125. DOI: 10.30865/ JSON.v1i2.1939.
[26] Syarif, S., A and Dewiani (2018) 'Trending topic prediction by optimizing K-nearest neighbor algorithm', Proceedings of the 2017 *4th International Conference on Computer Applications and Information Processing Technology*, CAIPT 2017, 2018- Janua, pp. 1–4. DOI: 10.1109/CAIPT.2017.8320711.
[27] Tauf ik, A. (2018) 'Komparasi Algoritma Text Mining Untuk Klasifikasi Review Hotel', *Jurnal Teknik Komputer, IV*(2), 112–118. doi: 10.31294/jtk.v4i2.3461.
[28] Utami, P.D, and Sari, R. (2018) 'Filtering Hoax Menggunakan Naive Bayes Classifier', Multinetics, 4(1). 57. DOI: 10.32722/vol4.no1.2018.pp57-61.
[29] Utari, D.R (2018) 'Prediksi Bidang Kerja Bagi Lulusan Program Studi Vokasi Sekretaris Menggunakan Teknik Klasifikasi Data Mining', *Jurnal Sekretari & Administrasi (Serasi), 16*(2). 115–123.
[30] Utomo, D.P and Mesran, M. (2020) 'Analisis Komparasi Metode Klasifikasi Data Mining dan Reduksi Atribut Pada Data Set Penyakit Jantung', *Jurnal Media Informatika Budidarma, 4*(2). 437. doi: 10.30865/mib.v4i2.2080.
[31] Yahya, N. and Jananto, A. (2019) 'Komparasi Kinerja Algoritma C.45 Dan Naive Bayes Untuk Prediksi Kegiatan Penerimaanmahasiswa Baru (Studi Kasus : Universitas Stikubank Semarang)', Prosiding SENDI, (2014). 978–979.