I-STATEMENT, Vol. 9 No. 1, September 2023, pp. 30-45
ISSN: 2442-8345

Prediction of Total Transaction Using Extreme Gradient Boosting (XGBoost)

Mohamad Akbar Wisnu Nadyanto*[1], Desy Komalasari[2]
[1,2]Sekolah Tinggi Ilmu Manajemen dan Ilmu Komputer ESQ
e-mail: *[1]m.akbar.w@students.esqbs.ac.id, [2]desy.komalasari@esqbs.ac.id

Abstract: *In the 4.0 era, several sectors related to business have started to involve technology. One of the technologies applied is to analyze the business processes that occur, such as predicting the total transactions of a company. The technology involved is the application of machine learning or what is commonly referred to as machine learning. Machine learning is very suitable for use in the context of regression prediction, where data will be trained in such a way that a pattern is obtained that will be used for regression prediction. The machine learning method used is Extreme Gradient Boosting (XGBoost). The performance of XGBoost can be classified as very good, this is proven in the model testing that has been carried out. The results of the model evaluation show that the MAE values in the training data and test data yield values of 0.1353 and 0.1308, respectively. While the evaluation using R^2 on the training data and test data each shows the same value, 0.99.*

Keywords: *Predictions, Sales Forecasting, Machine Learning, XGBoost.*

1. INTRODUCTION

At this time, the amount of data is growing exponentially. Along with this, data can be processed into useful information using statistics, especially in the business field. With the growth of data in the digital era like now, data processing is no longer only done with a statistical approach but can be combined with a machine learning technology approach. This is because with the increase in data volume, data processing using statistics takes a long time and is prone to input or calculation errors so that it has an impact on the results. For this reason, machine learning is present as automation in processing data which also uses statistical theory in building mathematical models [1]. According to Rajula et al., [2] The main difference between statistical approaches and machine learning is that statistics aim to infer relationships between variables, while machine learning focuses on making predictive models. The advantage of machine learning is that it has great flexibility and is free from assumptions, while statistics relies on strong assumptions such as the type of error distribution, parameters in linear predictors, and proportional. Machine learning also has the ability to analyze different types of data and integrate them into predictions. One technique that is widely used in machine learning is prediction.

Prediction is an estimate of what will happen in the future. Predictions are obtained based on past knowledge and experience [3]. In its application, prediction is divided into two, namely classification and regression. Classification is a grouping of data used for discrete target variables. Only a few types of possible target values are obtained and no time series values must be obtained to obtain the final value. While regression is similar to classification, the difference is that regression cannot look for patterns described as classes and regression is used for continuous target variables. In regression, the time series value must be calculated to get the final

value and there is a continuous value that must be calculated to get the desired final target value [4].

As the amount of data and technology grows rapidly, machine learning approaches are widely used in business, especially sales transaction prediction [5]. By predicting regression on sales transactions, it can help companies make better sales plans so they can determine the right steps in increasing these sales transactions. Therefore, it is necessary to use a machine learning approach to automate data so that processing large amounts of data can be faster and produce more accurate prediction results.

Machine Learning is computer programming to optimize performance criteria using sample data or past experience [1]. Machine learning modeling is defined by multiple parameters and learning is the execution of a computer program to optimize model parameters using training data or past experiences. Models in machine learning can be predictive to make future predictions or descriptive to derive knowledge from data or both. In this case, the machine learning that is carried out is regression prediction using machine learning methods. Prediction is done is regression by predicting numerical data that is continuous and evaluating the model that has been made. One of the most popular machine learning methods used in regression prediction research is the XGBoost method.

eXtreme Gradient Boosting or more commonly known by XGBoost is a machine learning algorithm model which is an implementation of a regression tree and has the same decision rules as the decision tree. In a regression tree, each node represents a value for attribute testing and each leaf with a score represents a decision [6]. XGBoost also includes an efficient linear model solver and tree learning algorithm, this algorithm can also be used for various objective functions, including regression, classification, and ranking. The advantages of XGBoost include being able to perform parallel computing automatically so that in general it is 10 times faster than gradient boosting machines (GBM), XGBoost also supports customization of objective functions, evaluation functions, and has better performance on several different datasets [7]. For this reason, XGBoost is one of the most popular machine learning algorithms and is widely used in classification or regression prediction research.

Many researches on transaction or sales regression predictions have been carried out, including a study entitled "Machine Learning Model for Sales Forecasting by Using XGBoost" [6]. This study uses the XGBoost method and the purpose of this study is to determine the performance of the XGBoost algorithm model. The results of this study show that the calculation of RMSSE (Root Mean Squared Scaled Error) obtained the lowest XGBoost model of 0.655, linear regression obtained a result of 0.783, 19.5% higher than the XGBoost model, and the ridge regression model obtained a result of 0.774, also 13, 6% higher than the XGBoost model.

The next research is entitled "Walmart Sales Forecasting using XGBoost Algorithm and Feature Engineering" The next research is entitled "Walmart Sales Forecasting using XGBoost Algorithm and Feature Engineering" [5]. This study uses the XGBoost method and the purpose of this study is to determine the performance of the XGBoost algorithm model. The results of this study show that the calculation of RMSSE (Root Mean Squared Scaled Error) obtained the lowest XGBoost model of 0.652, logistic regression yielded a value of 0.793, 0.141 greater than the XGBoost model, and the ridge regression model yielded a value of 0.765, 0.113 greater than the XGBoost model.

Based on the explanation and some previous research references that have been described, XGBoost has the lowest error value compared to other machine learning methods. Therefore, in this study,

we will apply the concept of machine learning using the XGBoost method to analyze the total transaction regression prediction model. The data used is XYZ company transaction data on the Pasar Digital Usaha Mikro, Kecil, Menengah (PADI UMKM) platform. The application of predictions made in this study is regression prediction that focuses on evaluating model testing so that it is expected to be able to get the best model for predicting total transaction regression.

2. THEORETICAL BASIS

2.1 PASAR DIGITAL USAHA MIKRO, KECIL, MENENGAH (PADI UMKM)

Pasar Digital Usaha Mikro, Kecil dan Menengah (PADI UMKM) is a digital platform initiated by the ministry of BUMN and in synergy with BUMN, ministry of UMKM and also the Government Service Procurement Policy Institute. PADI UMKM was also created with the aim of opening wider market access for UMKM for the procurement of government goods and services, especially BUMN [8]. PADI UMKM is also initiated to monitor the contribution of BUMN to UMKM throughout Indonesia. Monitoring includes collecting data on UMKM sellers, transaction data for BUMN with UMKM, e-commerce enablers with marketplace integration, BUMN e-procurement data, and UMKM financing data that can be monitored at the PADI UMKM Control Tower. The service features that exist in PADI UMKM other than the B2B and B2C marketplaces are tax automation used by BUMN buyers to make it easier to determine the tax value before making transactions, then PADI e-procurement, which is a digital transformation solution for BUMN procurement using a procurement platform 4.0, and Control Tower Dashboard to find out information related to UMKM and BUMN spending on UMKM.

Control Tower Dashboard PADI UMKM is one of the features of PADI UMKM that provides statistical and comprehensive information related to UMKM and BUMN spending on UMKM. Control Tower has the goal of being able to get accurate and real time data-based insights so that it will make it easier for policy making, providing incentives, and new rules in the UMKM ecosystem [9]. Control Tower Dashboard users who use this system are the government and the Ministry of BUMN as the basis for data analysis for policy determination, for UMKM to see the total transaction or transaction value, and also for BUMN to view transactions and other activities in the PADI UMKM ecosystem. An example of the visualization of the Control Tower Dashboard PADI UMKM can be seen in Figure 1.

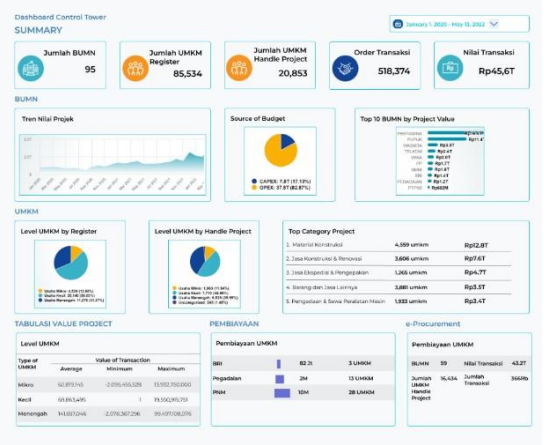

Figure 1. Data Visualization of Control Tower Dashboard PADI UMKM

2.2 MACHINE LEARNING

Machine learning is programming a computer to optimize performance criteria using sample data or past experience. Machine learning uses statistical theory in building mathematical models, because its core task is to make inferences or conclusions from samples. The role of computer science is twofold: firstly, in terms of training, it needs efficient algorithms to solve optimization problems, as well as to store and process large amounts of data that it usually has. Second, once the model has learned it, the representation and algorithmic solutions for the inference requirements become efficient as well. In certain applications, the efficiency of a learning or inference algorithm can be measured by

the complexity of space and time, this is as important as the accuracy of its prediction [1].

According to Ibnu Daqiqil [10], Machine learning is a field of study that focuses on the design and analysis of algorithms that enable computers to learn. Machine learning can also be defined as something that has the ability to learn without being explicitly programmed. The program utilizes past data to build models and make decisions on the models that have been made previously. Machine learning applications have 3 components, namely task T, performance measure P, and experience E. To build a machine learning application, these 3 components must be identified. Some examples of T tasks in machine learning applications are as follows:

- Classification, grouping data that has a label or target class. In this type of task, the computer is instructed to "guess" or select a group or class from the data. An example is the classification of emotions in photos which have 3 classes, namely happy, angry, and neutral.
- Regression, one of the machine learning applications for computers is ordered to "guess" numbers on a continuous data type based on a number of data. For example, in stock price predictions, the output of the prediction is the stock price prediction value.
- Transcription, a type of machine learning task that tries to understand an unstructured data and represent other data. An example is a speech to text application where data is in the form of sound waves containing text and then converted into text.
- Machine Translation, the process of converting an ordered set of inputs containing certain symbols into certain symbols in other languages. Examples are Indonesian and English translator applications.
- Anomaly Detection, a type of task that aims to examine some data and then mark it as something unusual.
- Syntesis and Sampling, a type of task that aims to create something based on an existing example. An example is an application that can generate a landscape image based on a particular sketch example.

Machine learning can be classified based on how the machine learns so that it can do its job. The classification of machine learning based on how it is learned is as follows:

- Supervised Learning, the computer will learn the labeled training data. An example is the detection of cats and dogs, the computer receives input data from cats and dogs and then studies it with a certain algorithm, then creates a model based on training data containing examples of cats and dogs. As a result, the computer was able to classify cats and dogs that had never been seen before. Examples of algorithms included in supervised learning are Linear Regression, Logistics Regression, k-Nearest Neighbor, Support Vector Machine (SVM), Random Forests, Neural Networks, and so on.
- Unsupervised Learning, different from supervised learning. Unsupervised learning is the opposite where the learning process is carried out without instructions on unlabeled data. For example, in data clustering of two different items, the two items will be separated into several categories depending on the data. The computer only knows the features that will be used to determine the difference between the two items is the color of the shape. By applying the clustering algorithm, the computer will be able to divide the items into two groups without having to be previously labeled and automatically grouped based on the color and shape of the object. An

example of an algorithm that is included in simple unsupervised learning is K-Means.

- Reinforcement Learning, In machine learning, a computer will interact with a very dynamic environment in which the computer must perform a specific task. Machines will learn how to make specific decisions based on a changing environment. Examples of cases are chess games and self-driving cars.

2.3 PREDICTION

Prediction is the art and science of predicting future events by involving taking historical data and projecting it into the future with a systematic approach model [11]. Understanding Prediction is the same as forecast or forecast. According to the Big Indonesian Dictionary, prediction is the result of predicting or forecasting or estimating future values using past data. Prediction shows what will happen in a given situation and is an input for planning and decision-making processes.

2.4 FEATURE ENGINEERING

Feature engineering is the process of using domains to extract features, test trial and error, and evaluate models. Feature engineering is one step that is quite complex and quite time-consuming because it is the main task in the data preparation phase, finding suitable features is an important element in learning a good predictive model [12].

According to Sitorus, Rizal, and Jajuli [13] feature engineering is an activity of extracting features from raw data and converting them into a format that is suitable for the machine learning model created. Feature engineering also involves applying transformation functions such as aggregates and arithmetic operators to a given feature to generate new features. This transformation helps to scale features convert non-linear relations between features and target classes into linear relationships that are easier to learn.

2.5 EXTREME GRADIENT BOOSTING (XGBOOST)

eXtreme Gradient Boosting (XGBoost) is a boosting algorithm implementation of a gradient decision tree designed for performance and speed. XGBoost is used for supervised learning problems that use training data wherex_i is the independent variable to predict the dependent variable y_i [14]. XGBoost is a highly scalable, flexible and versatile tool. This method is designed to exploit resources properly and to overcome the limitations of previous gradient boosting. The main difference between XGBoost and other gradient boosting is that it uses a new regularization technique to control overfitting. Overfitting is a condition where the data used for training is the "best" data, but if the test is carried out using different data, it will reduce its accuracy. The key to XGBoost's computing speed lies in optimizing the computer's cache and memory usage so that it can work efficiently even with large amounts of data. Therefore, XGBoost is faster and more powerful in tuning and modeling [15].

Based on the explanation above, XGBoost can be mathematical formula as follows:

$$\hat{y}_i^{(t)} = \sum_{k=1}^{t} F_k(x_i), \qquad f_k \in F$$

Where $\hat{y}_i$ is predicted value, F is the set that includes all the regression trees, f_k is one of the regression trees, and K is the sum of the regression trees. The Predicted value $\hat{y}_i$ is expected to be as close as possible to the true value of y_i. In XGBoost there are objective functions, training loss functions, and regularization. The objective function in XGBoost is as follows:

$$obj(\theta) = L(\theta) + \Omega(f_k)$$

Where L is loss function and Ω is regularization. Loss function is useful for representing the difference between the predicted value of $\hat{y}_i$ and the actual value y_i. *Loss function* is formulated as follows:

$$L(\theta) = \sum_i l\,(\hat{y}, y_i)$$

Then $\Omega(f_k)$ is regularization that defines the complexity of the model. The smaller the value of $\Omega(f_k)$, the lower the complexity and the stronger the generalization ability. The formula for calculating $\Omega(f_k)$ is as follows:

$$\Omega(f) = \gamma \mathrm{T} + \frac{1}{2}\lambda \|w\|^2$$

Where γ and λ is constant coefficient, T is the number of leaf nodes, and w is a representation of the score of the leaf nodes. In using XGBoost, there are several parameters that can be set and adjusted in value to get better results.

2.6 MEAN ABSOLUTE ERROR (MAE)

Mean Absolute Error (MAE) is one of the evaluation indicators of the regression model that measures the average value of the absolute difference between the actual value and the predicted value [16]. The MAE value can be said to be good if the value is getting lower or close to zero, MAE is formulated as:

$$MAE = \frac{1}{n}\sum_{i=1}^{n} |A_i - f_i|$$

Where n is the number of data, A_i is the actual data value, and f_i is the predicted data value.

2.7 COEFFICIENT OF DETERMINATION

Coefficient of determination or more often called *R Squared* (R^2) is is a measurement of the proportion of variation in one variable (objective or response variable) that is explained by another variable (explanatory variable) in the regression. This measurement is mostly done in regression [17]. The test is carried out with R^2 or coefficient of determination which is to see how much influence the X variable simultaneously has on the Y variable. R^2 is formulated as:

$$R^2 = 1 - \frac{RSS}{TSS}$$

Where RSS is *Residuals Sum of Squares* or the square of the difference between the predicted Y value and the average Y value and TSS is the Total Sum of Squares or the square of the difference between the actual Y value and the average Y value. The R^2 value ranges from zero to one, the closer to one, the greater the effect. This is because RSS has a value between 0 and the sum of the squares of Y, so the coefficient of determination can be from zero to one, or 0% to 100%.

3. RESEARCH METHOD

3.1 DATA COLLECTION

In this research, the data used came from the control tower database which was obtained from the company where the researcher interned. This data is transaction data and UMKM registered data from one agency in 2019-2021 as many as 48,259 transaction data and 3,076 UMKM data. UMKM data is only used as additional information to determine the number of UMKM registered in 2019-2021 and does not affect the research process. The transaction data in 2019 was 93 data, in 2020 there was 10,288 data, and in 2021 there was 37,878 data. As for the UMKM data in 2019 as much as 0 data, in 2020 as much as 2,165 data, and in 2021 as much as 911 data. The details and frequency of transaction data and MSME data can be seen in Table 1.

Table 1. Frequency of Total Transaction and UMKM Data 2019-2021

Year	Data (Frequency)	
	Transaction	UMKM
2019	93	0
2020	10.288	2.165
2021	37.878	911

In addition, the transaction data contains 28 attributes as shown in Table 2.

Table 2. Transaction Data Attributes 2019-2021

No	Attributes	Description

1.	trx_date	Date of transaction
2.	trx_id	ID transaction
3.	bumnid	BUMN that transact
4.	project_name	Project name purchased by BUMN
5.	project_category	Project category
6.	total_project_value	Total value project
7.	type_project_value	Type project value
8.	uid	ID UMKM
9.	umkm_name	UMKM name
10.	umkm_category	UMKM category
11.	umkm_classification	UMKM classification
12.	pengampu	UMKM support
13.	type_pengampu	Support type
14.	provinsi	Seller province
15.	kota	Seller city
16.	buyer_id	ID buyer
17.	status	Transaction status
18.	users	UMKM user
19.	post_timestamp	Post transaction date
20.	update_timestamp	Update transaction date
21.	cluster	BUMN cluster
22.	batch	Batch transaction
23.	bumn_code	BUMN code
24.	buyer_group_provinsi	Buyer group province
25.	purchase_order_date	Purchase order date
26.	payment_group	Payment group
27.	payment_term	Payment term
28.	payment_method	Payment method

3.2 PREPROCESSING

At this stage, data cleaning is carried out so that the data is free from missing values, outliers, and also the removal of irrelevant data. This data cleaning is done manually by checking the presence or absence of missing values, then deleting columns that are not relevant to the research. From the preprocessing that has been done, the total column or attribute that previously amounted to 28 becomes 7 columns. The 7 columns are trx_date, bumnid, project_category, total_project_value, type_project_value, umkm_category, and umkm_classification. The preprocessing stage is also carried out by presenting data that takes references from Figure 1 and then some are reprocessed by displaying data specifically for XYZ company. This presentation is only as additional information on the analysis of the data taken and does not enter into the prediction model development process. This process is explained about the total transactions, the total registered UMKM, the total transaction value, the total transaction plot, the total transaction plot based on the type of project value, the total transaction plot based on the UMKM category, the 10 most transactions by category, and tabulation. The presentation of the data is as follows:

- Total transaction, displaying the number of transactions of XYZ company on the PADI UMKM platform, total 48,259 transactions as shown in Table 3.

Table 3. Total Transaction XYZ Company

BUMNID	Total Transaction
XYZ	48.259

- Total registered UMKM, showing the number of UMKM registered by XYZ company on the PADI UMKM platform, total 3,076 UMKM as shown in Table 4.

Table 4. Total UMKM Registered XYZ Company

BUMNID	Sum UMKM Registered
XYZ	3.076

- Total transaction value, displaying the total transaction value of XYZ company on the PADI UMKM platform, amount Rp.1,878,558,906,502 as shown in Table 5.

Table 5. Total Transaction Value XYZ Company

BUMNID	Total Transaction Value
XYZ	Rp.1.878.558.906.502

- The trend plot for total transactions, showing a graph in the form of a line chart of the total transactions of XYZ company showing a pattern that continues to increase from month to month as shown in Figure 2. Details of total transactions can be seen in Table 6.

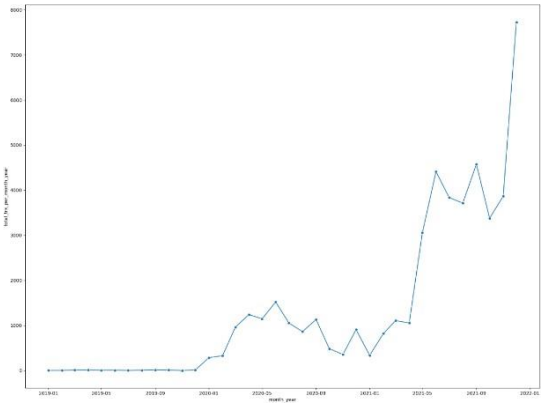

Figure 2. Total Transaction Plot XYZ Company

Table 6. Total Transaction XYZ Company per Month Detail

Year	Month	Total Transaction per Month
2019	January	3
	February	4
	March	10
	April	11
	May	7
	June	8
	July	3
	August	8
	September	15
	October	10
	November	2
	December	12
2020	January	287
	February	328
	March	960
	April	1242
	May	1147
	June	1522
	July	1056
	August	863
	September	1133
	October	486
	November	356
	December	908
2021	January	338
	February	819
	March	1108
	April	1055
	May	3057
	June	4411
	July	3835
	August	3713
	September	4575
	October	3378
	November	3864
	December	7725

- Plot of total transactions based on the type of project value, showing the percentage in the form of a pie chart of transactions for two types of projects, Capital Expenditure (CAPEX), namely expenses made by the company aimed at increasing asset value. Operating Expenditure (OPEX) is expenditure made by the company to meet operational needs. The total transactions for each type of project are 20,819 and 27,440 and the percentages are shown in Figure 3.

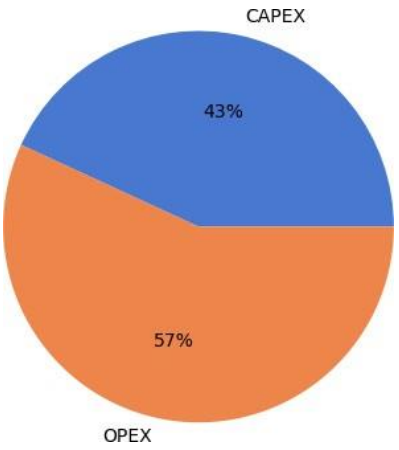

Figure 3. Percentage of Total Transactions by Project Type

- Plot of total transactions by UMKM category, showing the percentage in the form of a pie chart for each UMKM category *mikro, kecil,* and *menengah*. Total transactions from each category amounted to 4,716, 9,093 and 34,446 as shown in Figure 4.

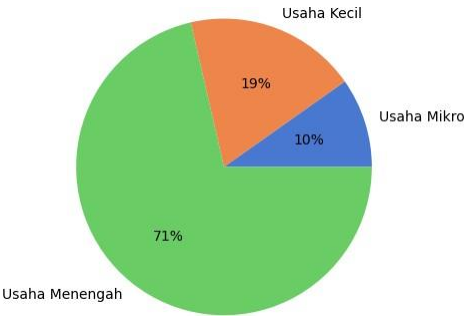

Figure 4. Percentage of Total Transactions by UMKM Category

- Top 10 transaction categories, displaying the top 10 categories of transactions based on the highest transaction value. The top category is

construction and renovation services with a total transaction of 15,488 transactions and a total transaction value of Rp.907,235,781,343. The details of the 10 categories with the most transactions are shown in Table 7.

Table 7. Top 10 Category Most Transaction Value XYZ Company

project_category	total_trx_umkm_category	total_value_category
Jasa Konstruksi & Renovasi	15.488	Rp.907.235.781.343
Material Konstruksi	7.057	Rp.455.516.274.389
Jasa *Event Organizer*	1.495	Rp.98.793.586.357
Barang Elektronik, Komputer, & Periferal	3.794	Rp.87.552.461.861
Souvenir & *Merchandise*	3.386	Rp.59.914.567.377
Alat & Jasa Kesehatan-Keselamatan	2.700	Rp.38.125.971.924
Alat Tulis Kantor	6.503	Rp.29.596.169.546
Jasa Perawatan Elektronik & IT	518	Rp.29.321.858.455
Jasa *Advertising*	711	Rp.27.854.583.637
Barang dan Jasa Lainnya	129	Rp.23.722.115.843

- Tabulation, presentation of data for three UMKM category showing the average, min, and max values of the total transaction value. Tabulation can be seen in Table 8.

Table 8. Tabulation UMKM Category

umkm_category	total_project_value		
	Mean	Min	Max
Kecil	1,344164e+07	3,300000e+03	8,714960e+08
Menengah	4,830605e+07	-2,078367e+09	4,788000e+09
Mikro	1,958465e+07	-2,095456e+09	1,790000e+09

3.3 FEATURE ENGINEERING

Basically, all machine learning algorithms use features extracted from input data to create output data. To prepare input data that is precise, compatible, and improves the performance of machine learning models, it is necessary to apply engineering feature techniques to get the expected output. The feature engineering techniques carried out are as follows:

- Changed the data type in the trx_date attribute which was previously an object data type to a datetime data type using the pandas library.
- Added the month_year attribute, which is data per month from 2019-2021 which is obtained from the trx_date attribute using the period in the pandas library.
- Added year and month attributes by retrieving data from the trx_date attribute and then taking the year and month format separately as new attributes using the datetime library.
- Changed the value of some category data in the umkm_category attribute because it causes redundancy using the pandas library.
- Added the sum_total_value_per_month_year attribute, which is an attribute to find out the total transaction value sorted by month and year. These attributes are taken from the month_year and total_project_value attributes using the group by function from the pandas library.
- Added the total_trx_per_month_year attribute, which is an attribute to find out the total transactions sorted by month and year. These attributes are taken from the month_year and total_project_value attributes using the group by function from the pandas library.
- Added the total_umkm_category attribute, which is an attribute to find out the total number of UMKM per category. These attributes are retrieved from the project_category attribute using the group by function from the pandas library.
- Added the total_value_category attribute, which is an attribute to find out the total value of UMKM

transactions per category. These attributes are taken from the project_category and total_project_value attributes using the group by function from the pandas library.

- Changed the attributes of type_project_value, project_category, umkm_category, and umkm_classification by encoding them to numeric categorization. From the encoding, four additional attributes are obtained, type_project_value_code, project_category_code, umkm_category_code, and umkm_classification_code.

From the feature engineering that has been carried out, there are 10 additional attributes so that the total attributes are 17.

3.4 DATA PROPORTION

The main step in implementing machine learning and data mining is to enter some data into an algorithm which then trains it to understand data patterns. After the algorithm learns the pattern, one has to enter other data to check the level of understanding performed by the algorithm. It therefore divides the data into two subsets for training and testing purposes.

The ratio of training data and test data can be adjusted to get better performance. In this study, the distribution is done by choosing the proportion that is often used in similar studies, namely 80:20. The proportion is done by testing 80% for the training data which amounted to 38,607 and 20% for the test data which amounted to 9,652.

3.5 DATA ANALYSIS

Sales data on the PADI UMKM platform has many attributes such as trx_date, trx_id, bumnid, project_name, project_category, total_project_value, type_project, uid, umkm_name, umkm_classification, umkm_category, province, city, and status. The data from each of these attributes will be preprocessed so that it can be processed. Then the data will be tested by dividing the data into training data and test data to test which model will be used to predict the total transaction. This process is also carried out to determine the parameters X and Y as needed in this study. Next, divide the training data and test data, namely the variables x_train, x_test, y_train, and y_test with a proportion of 80:20 for each training data and test data. After dividing the data, the next process is tuning the XGBoost model with several hyperparameters to get the best model.

From the results of preprocessing, 17 attributes were obtained. These attributes will be selected as predictor or independent variables (X) and target or dependent variables (Y). For the variable that is used as the target attribute is total_trx_per_month_year or total transactions per month, this variable will be used to predict the total transactions. The selection of attributes to be used as predictor variables using the Pearson correlation test. Pearson correlation test is a correlation test that is used to measure the strength of the relationship between two variables X and Y as a reference for selecting features. Pearson correlation coefficient values between -1 to 1. -1 indicates a negative correlation, 0 means there is no linear correlation between the two variables, and +1 means there is a positive correlation [18]. Correlation testing is done by calculating several X variables against Y variables to see the coefficient values that have a positive correlation and selecting attributes based on the highest Pearson value. The results of the Pearson correlation test can be seen in Table 9.

Table 9. Pearson Correlation Result

Atribut	Total_trx_per_month_year
Month	0,586097
Year	0,694469
total_project_value	-0,058998
total_trx_per_month_year	1.000000
total_trx_umkm_category	0,112172
total_value_category	0,106778

type_project_value_code	0,026136
project_category_code	-0,107692
umkm_category_code	0,141242
umkm_classification_code	-0,108786
Month	0,586097

In Table 9, the results show that the highest correlation is in the month and year attributes, each of which produces a correlation coefficient of 0.58 and 0.69. This value indicates that there is a positive correlation to the Y variable or the total_trx_per_month_year attribute. Therefore, the attributes chosen as X variables to be predictors are month and year attributes. Next, a model testing scenario is carried out in order to obtain the best model and see the effectiveness of the XGBoost method for predicting total transaction regression.

3.6 DATA TESTING

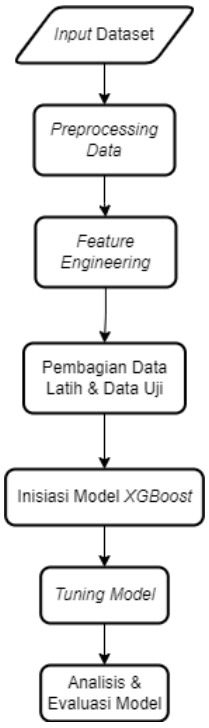

Figure 5. Data Testing Experiment Flow

Figure 5 shows the experimental flow of data testing that will determine the model to be used in the prediction of total transaction regression. At this stage, several test scenarios were carried out in order to get the best model. The details of the processing stages until testing are described as follows:

- Dataset Input
  The sales dataset to be processed has many attributes that are united in one comma-separated values (CSV) format file.
- Data Preprocessing
  Before the process of sharing training data and test data is carried out, it is necessary to preprocess data so that data processing gets more accurate predictions. The preprocessing stage consists of four stages, namely data cleaning by filling in data if something is missing and resolving inconsistencies found. Next is data integration, which brings together data with different representations to make data smoother. The next process is data transformation where the data will be normalized and generalized. The last process is data reduction, ie large amounts of data will be reduced in order to increase efficiency and analysis results.
- Feature Engineering
  This process is carried out by selecting and adding features from the dataset so that the model created can work more accurately in predicting the total transaction. The strategy carried out consists of feature selection which is the process of selecting features that are relevant to the problem to be solved. Feature extraction or the process of creating derivative features from existing features if these features are difficult to interpret. Then feature combination is the process of combining several features into one feature so that the model can be made as simple as possible.
- Sharing of Training Data and Test Data
  For the evaluation needs of the model to be trained, it is necessary to divide the training data and test data, the

proportion is done by dividing the training data as much as 80% and the test data 20%. Furthermore, random state initiation to randomize training data and test data.

- XGBoost Model Initiation and Tuning Model
  Designing the XGBoost model using the XGBoost library from python. The design of the model is carried out in several iterations in order to find out which model is suitable for the transaction dataset. Model tuning is carried out in two scenarios, namely by setting parameters by default and setting parameters using GridSearchCV from the sklearn library. Testing with GridSearchCV was carried out to get some of the best parameters for the data you have. Then check again using cross validation which is the default function of the XGBoost library to see if the model is good, overfitting, or underfitting.
- Model Analysis and Evaluation
  This process will display a line chart plot of the magnitude of the error value in each model that has been made. The model with the lowest error value will be used to predict the total transaction regression.

3.7 DATA INTERPRETATION

At this stage, the researcher first analyzed the results obtained from the data test scenario. These results are related to the analysis and theory from previous literature studies. Furthermore, these results are also associated with previous research to compare the results obtained with the results of previous studies. Then the last researcher will write conclusions and suggestions from the interpretation of the results of data testing carried out on the total transaction data.

4. RESULT AND ANALYSIS

4.1 ANALYSIS

At this stage, an analysis of the model testing that has been carried out is carried out. There are two scenarios tested, namely testing using all the default parameters from the XGBoost library and using GridSearchCV to generate the appropriate parameters automatically. The scenario table along with the parameters used for each test can be seen in Table 10.

Table 10. Parameter Testing of Each Scenario Model

Parameter	Test Model Scenario 1	Test Model Scenario 2
	Default Parameter Values	GridSearchCV Parameters Values
n_estimators	100	1000
max_depth	6	3
learning_rate	0,3	0,05
reg_lambda	1	1
reg_alpha	0	1
gamma	0	0
min_child_weight	1	0,5
early_stopping_round	*None*	10
n_estimators	100	1000
max_depth	6	3
learning_rate	0,3	0,05

In the model one test scenario, a model has been created using XGBoost default parameters. The results of the scenario are then evaluated to check how well the model is. The evaluation of the two model test scenarios was carried out using Mean Absolute Error (MAE) dan R Square (R^2).

The results of the evaluation of the MAE value in the model one test scenario for training data and test data are respectively 0.000295 and 0.000293. While the evaluation using R^2 on the training data and test data each shows the same value, namely 0.99. It can be seen that if the modeling is carried out without setting the parameters, the error value is very low. To check whether the model is overfitting, underfitting, or good, cross validation is carried out with a total of 10 folds and 100 iterations. 99 as shown in Figure 6.

```
   train-mae-mean  train-mae-std  test-mae-mean  test-mae-std
0     2582.129370       3.627125    2582.143555     32.729577
1     1807.645789       2.545255    1807.652783     23.328245
2     1265.933325       1.765065    1265.938098     16.732931
3      886.404785       1.236629     886.415234     11.645547
4      620.703894       0.872440     620.722522      8.452391
99    0.000283
Name: train-mae-mean, dtype: float64
99    0.000282
Name: test-mae-mean, dtype: float64
```

Figure 6. Results of Cross Validation Test Scenario Model 1

Based on Figure 6, the error value of the training data and test data generated by cross validation has a difference of 0.000012 and 0.000011. The meaning of the value using cross validation shows that the model can be said to be quite good.

In the model two test scenario, the results of the evaluation of MAE values are 0.1353 and 0.1308 for training data and test data, respectively. While the evaluation using R^2 on the training data and test data each also shows the same value, namely 0.99. The resulting error value is slightly higher than the first model test scenario, in the training data the difference is 0.135005 and the test data is 0.130507. Validation was also carried out to see whether or not the model with 10 folds and in the 999th iteration as shown in Figure 7.

```
   train-mae-mean  train-mae-std  test-mae-mean  test-mae-std
0     3503.468188       4.959934    3503.407397     44.656194
1     3328.495459       4.708876    3328.452808     42.465363
2     3162.267651       4.474240    3162.188159     40.395972
3     3004.383545       4.255414    3004.284277     38.423306
4     2854.209546       4.024687    2854.222754     36.546430
999    0.16117
Name: train-mae-mean, dtype: float64
999    0.162374
Name: test-mae-mean, dtype: float64
```

Figure 7. Results of Cross Validation Test Scenario Model 2

Based on Figure 7, the error value of the training data and test data generated by cross validation has a difference of 0.02587 and 0.031574. The meaning of the value using cross validation shows that the model can be said to be very good. This is because the error validation values for training data and test data are slightly larger than the error values for training data and test data.

4.2 DISCUSSION

In the previous process, we have carried out two model test scenarios by trying the default parameters and also the best parameters that have been generated by GridSearchCV. Based on the model test scenario that has been made, the best model is obtained, namely the test scenario of model two. The modeling in this scenario, although it produces a small error value compared to the model one test scenario, the model can be said to be better than the model two test scenario. The MAE values in the training data and the test data each produce a value of 0.1353 and 0.1308, meaning this value indicates that the actual value and the predicted value have a very low difference. While the evaluation using R^2 on the training data and test data each shows the same value, namely 0.99, this value means that the month (X1) and year (X2) attributes simultaneously or jointly affect the total_trx_per_month_year attribute (Y) so that it can be said that the resulting effect is very significant. Based on the model test, the difference between the actual value and the predicted value is very small. The interpretation of the comparison of the actual value and the predicted value is presented in the form of tables and graphical visualizations as in Table 11 and Figure 8.

Table 11. Results of Comparison of Actual and Predicted Values of Model 2

Tahun	Bulan	Nilai Aktual	Nilai Prediksi
2019	Januari	3	5.22
	Februari	4	7.68
	Maret	10	2.93
	April	11	14.47
	Mei	7	-5.75
	Juni	8	41.34
	Juli	3	3.81
	Agustus	8	-32.17
	September	15	13.13
	Oktober	10	16.56
	November	2	-11.44
	Desember	12	12.28
2020	Januari	287	286.86
	Februari	328	327.84

	Maret	960	960.43
	April	1242	1241.75
	Mei	1147	1147.41
	Juni	1522	1521.88
	Juli	1056	1055.65
	Agustus	863	863.05
	September	1133	1133.24
	Oktober	486	484.98
	November	356	356.85
	Desember	908	907.96
2021	Januari	338	338.06
	Februari	819	819.07
	Maret	1108	1107.68
	April	1055	1055.21
	Mei	3057	3056.89
	Juni	4411	4410.97
	Juli	3835	3835.12
	Agustus	3713	3713.01
	September	4575	4574.92
	Oktober	3378	3378.15
	November	3864	3863.93
	Desember	7725	7724.99

In Table 11 the predicted values generated from the XGBoost method show various values. Some of the predicted values are higher than the actual values, some are lower than the actual values, and there are even values that are almost the same as the actual values. This is because the regression model in the XGBoost method produces different residual values in each iteration. With these different residual or error values, after the calculations are carried out, the prediction values are obtained which are likely to be higher, lower, or even almost the same as the actual value. Based on this value, it does not mean that the resulting model is not good, but this is done by XGBoost to avoid overfitting or underfitting models. By selecting parameters using GridSearchCV, the XGBoost method performs regularization and constraints to prevent the model from overfitting or underfitting so that a good model is obtained.

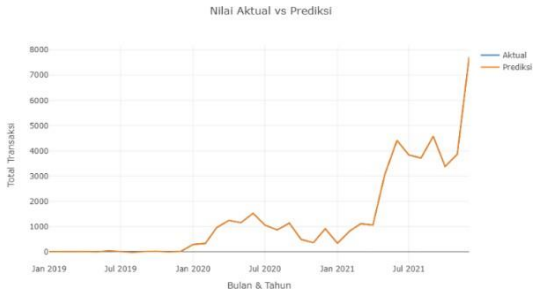

Figure 8. Visualization Plot of Comparison of Actuation Value and Prediction of Model 2

In Figure 8 it can be seen that the blue line that defines the actual value can be said to be close to the predicted value. However, if you look closely, the difference in values is still visible even though the distance difference is very small and can be seen in Figure 9. It is also indicated in Table 11 that all the predicted values produced have differences with the actual values.

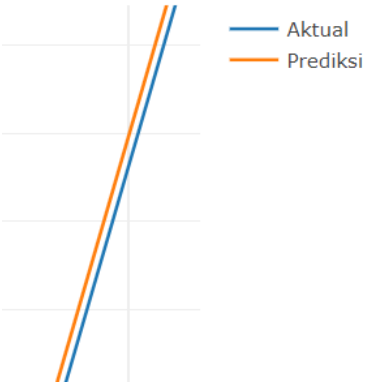

Figure 9. Difference between Actual Value Line and Predicted Test Model 2

Based on these results, this is evidenced by several previous studies using similar methods. In the research conducted by Dairu and Shilong [6], the resulting RMSSE value was 0.655. Meanwhile, in the research conducted by Ranjitha and Spandana [19], the resulting MAE value was 0.029. Research conducted by Zhang et al [20], the MAE generated from two different datasets is 0.178 and 0.046, respectively. The XGBoost method is proven to produce the lowest error value which is supported by several previous

studies that have been described previously and it can be said that the model from the XGBoost method can be implemented to predict total transaction regression.

5. CLOSING STATEMENT

5.1 CONCLUSION

Based on the results of research and analysis that has been done, the following conclusions can be drawn:

1. Based on the tests carried out, modeling using XGBoost provides good performance in this study. This is indicated by the results of MAE and R^2 given respectively 0.1308 and 0.99.
2. The evaluation of the best model performance is in the model two test scenario with MAE values in the training data and test data yielding values of 0.1353 and 0.1308, respectively. While the evaluation using R^2 on the training data and test data each shows the same value, 0.99.

5.2 SUGGESTIONS

Based on the results obtained during this research, the researcher provides suggestions to further researchers in the field and object of the same research as follows:

1. Further research can add to the analysis of transaction predictions with time series viewable per day, per week, per month, and per year so that subsequent predictions can be more informative.
2. Model testing can add more varied parameters if using a similar method.
3. The analysis and visualization carried out are still very simple, so they need to be redeveloped. For further researchers, they can develop the resulting output in the form of a system or dashboard so that the information displayed makes it easier for readers to understand it.

REFERENCES

[1] Alpaydin Ethem, “Introduction to Machine Learning,” 2020.

[2] H. S. R. Rajula, G. Verlato, M. Manchia, N. Antonucci, and V. Fanos, “Comparison of conventional statistical methods with machine learning in medicine: Diagnosis, drug development, and treatment,” *Medicina (Lithuania)*, vol. 56, no. 9, pp. 1–10, Sep. 2020, doi: 10.3390/medicina56090455.

[3] D. Kurniawan, *Pengenalan Machine Learning dengan Python*. PT Elex Media Komputindo, 2020.

[4] N. Wayan Wardani, “Penerapan Data Mining Dalam Analytic CRM,” 2021. [Online]. Available: https://www.researchgate.net/publication/351776255

[5] Y. Niu, “Walmart Sales Forecasting using XGBoost algorithm and Feature engineering,” in *Proceedings - 2020 International Conference on Big Data and Artificial Intelligence and Software Engineering, ICBASE 2020*, Oct. 2020, pp. 458–461. doi: 10.1109/ICBASE51474.2020.00103.

[6] X. Dairu and Z. Shilong, “Machine Learning Model for Sales Forecasting by Using XGBoost,” in *2021 IEEE International Conference on Consumer Electronics and Computer Engineering, ICCECE 2021*, Jan. 2021, pp. 480–483. doi: 10.1109/ICCECE51280.2021.9342304.

[7] T. Chen and T. He, “xgboost: eXtreme Gradient Boosting,” 2017.

[8] “Pengadaan Barang dan Jasa Pemerintah di Marketplace PaDi UMKM,” May 12, 2022. https://padiumkm.id/ (accessed May 12, 2022).

[9] "Frequently Asked Question," May 12, 2022. https://control.padiumkm.id/faq (accessed May 12, 2022).

[10] Daqiqil Ibnu, "Machine Learning: Teori, Studi Kasus, dan Implementasi Menggunakan Python," 2021.

[11] T. Indarwati, T. Irawati, and E. Rimawati, "PENGGUNAAN METODE LINEAR REGRESSION UNTUK PREDIKSI PENJUALAN SMARTPHONE," *Jurnal Teknologi Informasi dan Komunikasi (TIKomSiN)*, vol. 6, no. 2, Jan. 2019, doi: 10.30646/tikomsin.v6i2.369.

[12] F. Nargesian, H. Samulowitz, U. Khurana, E. B. Khalil, and D. Turaga, "Learning feature engineering for classification," in *IJCAI International Joint Conference on Artificial Intelligence*, 2017, vol. 0, pp. 2529–2535. doi: 10.24963/ijcai.2017/352.

[13] C. M. Sitorus, A. Rizal, and M. Jajuli, "Prediksi Risiko Perjalanan Transportasi Online Dari Data Telematik Menggunakan Algoritma Support Vector Machine," *Jurnal Teknik Informatika dan Sistem Informasi*, vol. 6, no. 2, 2020, doi: 10.28932/jutisi.v6i2.2672.

[14] A. A. Firdaus, A. Komarudin, M. P. Statistika, F. Matematika, D. Ilmu, and P. Alam, "Klasifikasi Pemegang Polis Menggunakan Metode XGBoost," 2021, doi: 10.29313/.v0i0.30320.

[15] Daoud Al Essam, "Comparison between XGBoost, LightGBM and CatBoost Using a Home Credit Dataset," 2019.

[16] J. Qi, J. Du, S. M. Siniscalchi, X. Ma, and C. H. Lee, "On Mean Absolute Error for Deep Neural Network Based Vector-to-Vector Regression," *IEEE Signal Processing Letters*, vol. 27, pp. 1485–1489, 2020, doi: 10.1109/LSP.2020.3016837.

[17] E. Kasuya, "On the use of r and r squared in correlation and regression," *Ecological Research*, vol. 34, no. 1, pp. 235–236, Jan. 2019, doi: 10.1111/1440-1703.1011.

[18] Rozy Fachrul, Fauzi Ali, Arum Yuita, and Dewi Eka, *Sentiment Analysis on Movie Reviews Using Ensemble Features and Pearson Correlation Based Feature Selection*. IEEE, 2018. doi: https://doi.org/10.1109/SIET.2018.8693211.

[19] P. Ranjitha and M. Spandana, "Predictive Analysis for Big Mart Sales Using Machine Learning Algorithms," in *Proceedings - 5th International Conference on Intelligent Computing and Control Systems, ICICCS 2021*, May 2021, pp. 1416–1421. doi: 10.1109/ICICCS51141.2021.9432109.

[20] L. Zhang, W. Bian, W. Qu, L. Tuo, and Y. Wang, "Time series forecast of sales volume based on XGBoost," in *Journal of Physics: Conference Series*, Apr. 2021, vol. 1873, no. 1. doi: 10.1088/1742-6596/1873/1/012067.