Accredited Ranking SINTA 2
Decree of the Director General of Higher Education, Research, and Technology, No. 158/E/KPT/2021
Validity period from Volume 5 Number 2 of 2021 to Volume 10 Number 1 of 2026

Published online on: **http://jurnal.iaii.or.id**

JURNAL RESTI
(Rekayasa Sistem dan Teknologi Informasi)
Vol. 7 No. 4 (2023) 809 - 816 ISSN Media Electronic: 2580-0760

Comparison of Naive Bayes and PSO-Based Naive Bayes Algorithms for Prediction of Covid-19 Patient Recovery Data in Indonesia

Alvina Felicia Watratan[1], Ema Utami[2], Anggit Dwi Hartanto[3]
[1,2,3] Magister of Informatics Engineering, Universitas Amikom Yogyakarta
[1]vinawatratan@students.amikom.ac.id, [2]ema.u@amikom.ac.id, [3]anggit@amikom.ac.id

Abstract

A brand-new illness known as COVID 19 was identified in 2019 but has yet to infect humans (World Health Organization, 2019). This group of viruses can infect mammals including humans as well as birds and cause sickness. People commonly contract coronaviruses from the flu and other minor respiratory ailments, but they can also spread serious diseases like SARS, MERS, and the deadly COVID-19. So that there are no more casualties, this number must be decreased. It is crucial to understand the variables that can truly reduce the danger of death and gauge the propensity for recovery in Covid-19 patients. Several techniques in data mining can be used to forecast patient recovery rates depending on various characteristics. This study's criteria included gender, age, province, and status. The Naive Bayes (NB) and Pso-based Naive Bayes algorithms are compared in this study using patient datasets to determine whether strategy is more accurate. The findings of this study reveal that using the NB method has a 94.07% accuracy rate, a precision value of 14%, a recall value of 1%, and an AUC value of 0.613, according to the study's data. The accuracy rate of PSO-based Naive Bayes is 95.56%, the precision is 25%, the recall is 1%, and the AUC is 0.540.

Keywords: *naive bayes algorithm; particle swarm optimization; covid-19 patients; data mining; classification*

1. Introduction

The coronavirus (CoV) also resembles the influenza virus and can cause the Middle East respiratory syndrome and severe acute respiratory syndrome. 2019 is the year of the discovery of this virus, commonly known as COVID-19. A novel disease COVID-19, has never been observed in humans [1]. The coronavirus first impacted humans in Wuhan, China, where the first case first surfaced. Although flu-like symptoms initially manifested, pneumonia was previously thought to be the culprit. Among these symptoms are appetite loss, weariness, fever, fatigue, and shortness of breath. In contrast to the flu, the coronavirus spreads quickly, has a greater potential for serious infections, and can even result in organ failure. Patients, especially those who are ill, go through this crisis [2].

On March 11, 2020, the coronavirus was declared a pandemic whose spread is growing very fast [1]. Therefore, we cannot ensure that the vast majority of nations are immune to the coronavirus [3]. In 196 countries, there were 18,440 deaths and 414,179 confirmed cases of COVID-19. 790 confirmed COVID-19 cases had been reported as of March 2, 2020, with 58 fatalities, 31 recoveries, and 701 patients still undergoing treatment.

The K-Nearest Neighbor (KNN) algorithm method has been used to predict the chances of healing for COVID-19 patients. To determine if a COVID-19 patient has a probability of survival or not, sixteen signs are found. Studies demonstrate that KNN can accurately forecast a patient's chance of recovering from COVID-19. A web-based prediction system uses the best KNN model as well. Future research can gather more representative data in order to obtain predictions that are more accurate because this study lacks the necessary data. Additionally, it might make use of Deep Learning techniques, which are used in machine learning applications, in addition to KNN. Probabilities of recovery or death are the two prediction classes that KNN creates. The probability of recovery for confirmed COVID-19 patients is predicted by KNN with an average accuracy of 88.16% based on experimental results on 496 data of confirmed positive COVID-19 patients. By choosing the top model out of five test scenarios based on the provided k value, the COVID-19 patient recovery prediction system was created. With an accuracy rating of 88.8%, the best model is found with a k value of 4 [4].

Accepted: 25-01-2023 | Received in revised: 01-06-2023 | Published: 12-08-2023

The least squares method and web-based techniques are expected to be used by researchers to estimate the overall number of Covid-19 patients. The rising patient population, a lack of facilities, money, and medical personnel, according to this study, limited Indonesia's ability to control COVID-19. Researchers created a COVID-19 patient prediction model based on this conflict using the trend least square technique. Based on this discussion, researchers created a forecasting method employing the least square trend method for the quantity of COVID-19 patients. RStudio tools are used in the prediction method. For Indonesian patients with COVID-19, the homogeneous prediction value of the mean absolute%age error (MAPE), which is employed in this study's prediction, is 59.2%. This demonstrates that it is impossible to forecast Indonesia's COVID-19 patient population using the least square trend technique. Researchers recommend conducting additional study on this forecasting system by applying alternative forecasting approaches or integrating new data sources in order to expand the diversity of patient growth variance and the accuracy of the conclusions reported by the system [5].

Based on the discussion of prior research, this study will use Particle Swarm Optimization (PSO), Naive Bayes, and Support Vector Machine (SVM) to predict data on Covid-19 patients in Indonesia with three categories of patient status, namely in treatment, recovered, and died. The motivation for this research was raised because this PSO method has not been widely used in predicting the status of Covid-19 patients in Indonesia, so that one form of stopping the virus' spread can be used. By practicing social seclusion, it is possible to stop the COVID-19 transmission chain. In order to prevent the transmission of the disease from spreading further, this social isolation keeps each group from becoming infected through interaction with others.

PSO is one of the metaheuristic algorithms that can resolve optimization issues. It might turn out that PSO is more competitive in some circumstances. When used in conjunction with neural networks and algorithmic classification approaches, this optimization strategy has been successful and useful in resolving multidimensional and multiparametric optimization problems in machine learning education [6]. In order to conduct this research utilizing the PSO-based Naive Bayes Algorithm, data must be collected in the context of the Covid-19 epidemic.

2. Research Methods

2.1. Research Flow

In order for this research to be well organized, the researcher made a research flow, and can be seen in Figure 1.

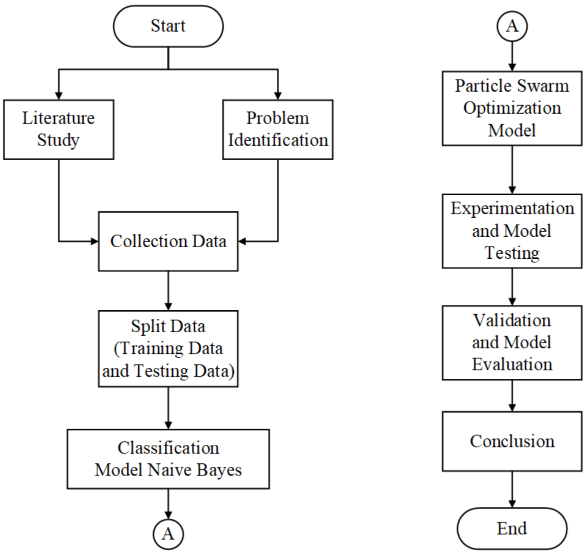
Start
Literature Study
Problem Identification
Collection Data
Split Data (Training Data and Testing Data)
Classification Model Naive Bayes
A
A
Particle Swarm Optimization Model
Experimentation and Model Testing
Validation and Model Evaluation
Conclusion
End

Figure 1. Research flowchart

First Stage: The first stage of this research is the literature review and problem identification phase, which entails searching the literature from various online journal references and then identifying problems from the references found in the prior literature search to highlight the issues raised here to study.

Second Stage: This phase will involve the gathering and sharing of data so that information about Indonesian patients with Covid-19 may be obtained through kawalcovid19.com. Age, gender, province, and patient condition are the three attributes that make up the 450 rows of data that make up the patient record. The patient's status is the necessary attribute if the patient's status may be divided into three categories: under treatment, recovered, and deceased. Training and testing data are divided into two groups after data collection, with 80% going to training and 20% going to testing.

Third Stage: At this level of classification, the Naive Bayes technique is utilized. With the Naive Bayes approach, the objective is to obtain accurate findings using Confusion Matrix and AUC and combine them with PSO to ascertain which method is the most accurate. Google Colab is the utilized tool.

Fourth Stage: Experiments and model testing are conducted during this phase. The steps in this fourth phase are: Create a test database with Covid-19 patient information; Perform data cleansing on the record; Use data separation to separate the Covid-19 patient data from the training and test data; and The results of the Confusion Matrix and the AUC produced from evaluating the data using the PSO-based Naive Bayes method are then compared to determine the accuracy of each methodology.

DOI: https://doi.org/10.29207/resti.v7i4.4893
Creative Commons Attribution 4.0 International License (CC BY 4.0)

Fifth Stage: Model validation and evaluation are carried out during this step. 10k fold cross-validation is a technique used in the validation process to reduce noise and word mistakes and increase accuracy. The AUC and confusion matrix analysis of accuracy measures yield scores for precision, accuracy, and recall.

Sixth Stage: The final stage includes a presentation of the study's findings, a discussion, and a conclusion.

2.2. Data Mining

Data mining uses machine learning, artificial intelligence, math, statistics, and other approaches to locate and extract pertinent information from large databases [7]. Processing extraordinarily large amounts of data automatically in order to uncover profitable links or patterns is another aspect of data mining.

A decision-support technique called data mining can search data for illuminating patterns. As a group of procedures, data mining may also be separated into other process phases. The user can participate in the interactive phase directly or through the database.

Based on the tasks used, data mining is categorized into different groups [8], namely: Description: the procedure of identifying significant traits in database data. You can describe a pattern or trend by using pattern and trend descriptions. Understanding repeating patterns in the data and being able to translate those patterns into simple rules or criteria is the goal of description.

Classification: target variable that is categorical. Finding patterns or traits that best explain the data and putting them into groups is the process of classification. Examining an object's qualities and placing it in one of a number of predetermined categories constitutes the act of classification.

Estimation: classification and evaluation are similar, with the exception that numerical evaluation uses the target variable instead of categorical evaluation. The model, which is built using the full dataset, provides the value of the target variable as a prediction value. Following that, the target variable value is approached during verification using the projected variable value.

Prediction: with the exception of the expected value being in the future, categorization, estimation, and prediction are comparable. Prediction is similar to classification and estimation except that the predicted value is in the future.

Clustering: classes of objects with comparable records come together to form clusters of interest. A cluster is a group of records that are distinct from records in other clusters but comparable to each other. The objective is to cluster together related things into groupings. The higher the similarity and lower the difference between objects in a cluster, the higher the quality of the cluster analysis.

Association: Look for characteristics that pop up right away. Finding guidelines to gauge the relationship between two or more traits is the work of association.

2.3. Classification

In data mining or machine learning, classification is one of the most important subfields. Identification of items into classes, classes, or groups is the process of classification. It is based on the specified definitions, properties, and procedures. To categorize items that only fit into one group is the goal of classification [9].

2.4 Naïve Bayes Algorithm

Thomas Bayes' theorem, which was discovered in the 18th century, is used in this technique [10]. An intriguing technique to introduce the use of probability theory in machine learning is using Bayes' theorem. Because many algorithms frequently perform a crucial function, if "only" in the background [11]. A statistical categorization known as Naive Bayes (NB) can be used to forecast the likelihood of falling into a particular category. Using the training data's frequency of each classification, NB uses probability theory, a subfield of mathematics, to determine the likelihood of a probable categorization [12].

Object attributes are taken into account to be independent in the Naive Bayes method. The "Final Decision Table's" frequencies are added to determine the probability that will be used for the final decision. In contrast to other classifiers, the Naive Bayes classifier outperforms. According to the paper “NB , Decision Trees and Neural Networks in the Classification of Training Web Pages” [13], the NBC has higher accuracy when compared to other classification models. The Bayes theorem can be seen in Equation 1 [14].

$$P(H|X) = \frac{P(X|H).P(H)}{P(X)} \quad (1)$$

Where X is the No class information available; H is the Category for hypothesized data; $P(H|X)$ is the Probability of hypothesis H in condition X; $P(H)$ is the Hypothesis H-probability; $P(X|H)$ is the X likelihood given the circumstances described in hypothesis H; and $P(X)$ is the Probability of X.

This Naive Bayesian method serves as an illustration of the need to comprehend the categorization process, which calls for a number of statements in order to decide which class is appropriate for the sample under study.

The naïve bayes technique can therefore be modified as Equation 2.

$$P(C|F1 \dots Fn) = \frac{P(C)P(F1 \dots Fn|C)}{P(F1 \dots Fn)} \quad (2)$$

The variables $F1$ through Fn reflect the remaining expression attributes required to complete the

DOI: https://doi.org/10.29207/resti.v7i4.4893
Creative Commons Attribution 4.0 International License (CC BY 4.0)

classification assuming variable *C* represents the class. According to this formula, the likelihood that a sample with particular characteristics will fall into class *C* (downstream) is equal to the probability that the sample will fall into that class multiplied by the likelihood that class *C* will actually occur (before the sample is collected, also known as the prior). In the case of the Class *C* characteristic, divided by the likelihood that the sample feature exists in all populations worldwide (also called the rate). The prior Equation can therefore be expressed as Equation 3.

$$Posterior = \frac{prior\ x\ likelihood}{evidence} \quad (3)$$

For every class in the sample, the proof value is constant. To establish which class the sample belongs to, the latter value is then contrasted with the loss values of the other classes. By declaring *(C|F1,..., Fn)* and carrying out the subsequent multiplication, the Bayes formula is employed in Equation 4.

$$\begin{aligned} &(C|F_1, \ldots, F_n = P(C)P(F_1, \ldots, F_n|C) \\ &= (C)P(F_1|C)P(F_2, \ldots, F_n|C, F_1) \\ &= P(C)P(F_1|C)P(F_2|C, F_1)P(F_3, \ldots, F_n|C, F_1,F_2 \\ &= (C)P(F_1|C)P(F_2|C, F_1)P(F_3|C, F_1,F_2)P(F_4, \ldots, F_n|C, F_1, F_2, F_3) \\ &= P(C)P(F_1|C)P(F_2|C, F_1)P(F_3|C, F_1,F_2) \ldots P(F_n|C, F_1, F_2, F_3, \ldots, F_{n-1}) \quad (4) \end{aligned}$$

What determines the outcome of this therapy is the intricacy of the elements influencing the probability value that is close to the probability that was individually evaluated. The calculation is challenging because of this. Each specification *(F1, F2,..., Fn)* is supposed to be independent of the others, with the naive assumption being that the level of independence is very high. As shown in Equation 5 and 6.

$$P(F_i|F_j) = \frac{P(F_i \cap F_j)}{P(F_j)} = \frac{P(F_i)P(F_j)}{P(F_j)} = P(F_i) \quad (5)$$

For i≠j , so

$$P(F_i|C, F_j) = P(F_i|C) \quad (6)$$

The Naive Bayes theorem is modeled in the equation above and used in the classification procedure. Using the Gaussian density formula, Equation 7 will categorize continuous data.

$$P(X_i = x_i \mid Y = y_j) = \frac{1}{\sqrt{2\pi\sigma_{ij}}} e^{-\frac{(x_i - \mu_{ij})^2}{2\sigma^2 ij}} \quad (7)$$

Where *P* is the Chances; *Xi* is the Feature I; *xi* the the *x* value to *I*; *Y* is the class to search; *yi* is the search term for the *Y* subclass; *μ* is the word "mean" expresses the average of all traits; and *σ* is the term "standard deviation" expresses how variable all attributes are of all qualities.

Figure 2 illustrates the flow chart for the Naive Bayes algorithm [14].

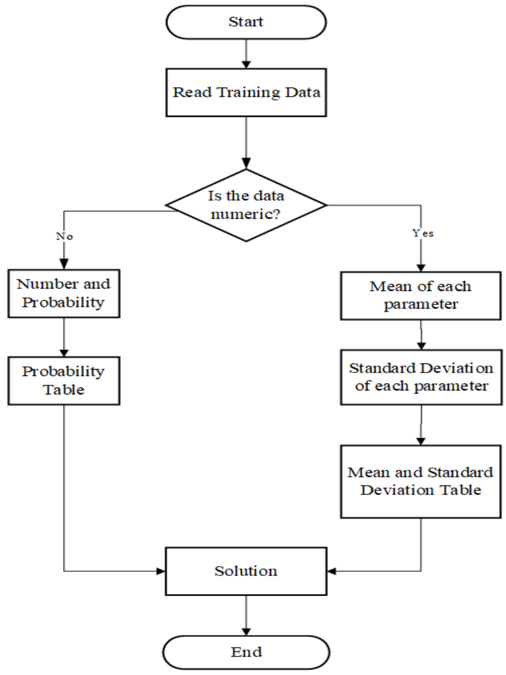

Figure 2. Flowchart of Naïve Bayes Algorithm

The explanation of Figure 2 are: Read the training information; Do the sum and probability calculations, but only if the data is a number : Calculate each parameter's mean and standard deviation if it is made up of numerical data. Equation 8 is used to calculate the mean.

$$\mu = \frac{\sum_{i=1}^{n} x_i}{n} \quad (8)$$

or

$$\mu = \frac{x_1 + x_2 + x_3 + \ldots + x_n}{n} \quad (9)$$

where *μ* is the word "mean" expresses the average of all traits; x_i is the *x* value to *I*; and *n* is the total samples.
Equation 10 is for calculating the standard deviation value.

$$\sigma = \sqrt{\frac{\Sigma_{i=1}^{n} (x_i - \mu)^2}{n - 1}} \quad (10)$$

Where *σ* is the term "standard deviation" expresses how variable all attributes are of all qualities; x_i is the *x* value to *I*; *μ* is the word "mean" expresses the average of all traits; and *n* is the total samples.

The probability will be calculated by dividing the total number of data in a class by the number of identical data in that class. It is advisable to consult tables of probability, mean, and standard deviation statistics, after that, the solution is complete.

2.5 Particle Swarm Optimization (PSO)

Using social simulation models as the foundation for a stochastic optimization technique, James Kennedy and Russell Eberhart created the PSO evolutionary algorithm in 1995. PSO has two primary component

DOI: https://doi.org/10.29207/resti.v7i4.4893
Creative Commons Attribution 4.0 International License (CC BY 4.0)

methods as its foundation, but it also has connections to evolutionary programming, genetic algorithms, and evolutionary computing [15]. PSO, or population search optimization, is a technique that emerged from studies of the movement of creatures in communities of birds or fish, like the genetic B algorithm. Through iterative repetition, PSO looks for a population (collection) of individuals (particles) [16].

Some research claim that PSO is a swarm information system technique that can handle optimization issues in the search space [17]. Based on biological examples, such as the occurrence of animal groups, where each group has individual behaviors when executing coordinated actions to attain the same goal, swarm intelligence systems make use of inventive intelligence to solve optimization challenges. Swarms are frequently referred to as agents. Because there is no central authority to regulate each agent's response, global intelligence develops without any one agent in the swarm using it to accomplish the desired result. Each agent in the swarm functions according to local norms. Swarm information systems must take into account a number of features, including: Normative behavior, fault tolerance, speed, modularity, and parallelism.

The PSO method correlates each particle with its position and velocity to dynamically conduct a fresh search based on its behavior, searching the population for a random solution. Based on personal best (pbest) and global best (gbest), or each particle's experience in finding the best solution, each particle's fitness value must be assessed for each generation. The impact of the prior speed on the new speed can be calculated using this experience as an inertia weight parameter [18].

If the global and local search procedures are balanced, the value of inertia weight can boost particle speed [17]. The variables used in this process are: One element that is thought to be crucial to the solution of the issue is the quantity of particles; The inertia weight has a considerable impact on the PSO algorithm's particle speed; The learning aspect: The identifying element is parameter c1, and the social component is parameter c2. For the PSO algorithm's convergence behavior, it is not very significant; Particle Dimensions and Area: Based on the solved problem, the particle size and surface area are chosen; The maximum velocity is defined as the largest change any particle can undergo in a single repetition; If the necessary conditions are satisfied, the stop condition is an option.

The explanation of Figure 3 are: The process starts with initializing the parameter values in the PSO algorithm; Then initialize the generation of the initial position and velocity of the particle which is done with one random process by taking patient data to be processed together with data taken randomly; The next process is evaluating the fitness function of each particle; The next stage is carried out by determining the better local best and global best value updates; And the next step is to check whether the fitness value obtained has met the criteria?; If not, the process continues by updating the velocity value and updating the position value of the next particle; and If yes, then the final result is to display the best gbest solution.

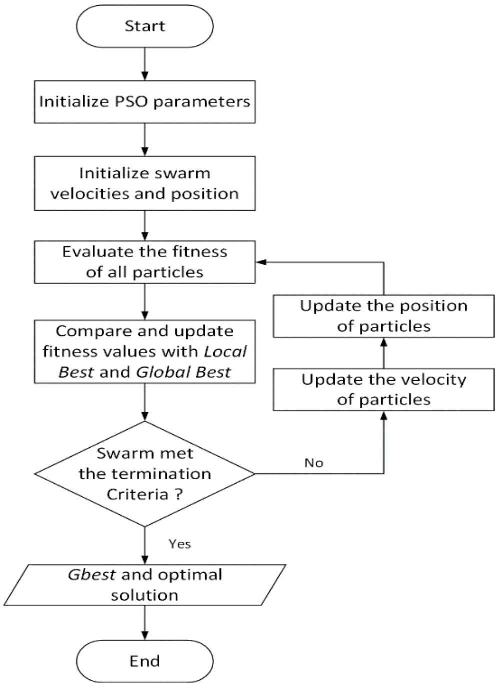
Start
Initialize PSO parameters
Initialize swarm velocities and position
Evaluate the fitness of all particles
Compare and update fitness values with *Local Best* and *Global Best*
Update the position of particles
Update the velocity of particles
Swarm met the termination Criteria ?
No
Yes
Gbest and optimal solution
End

Figure 3. Particle Swarm Optimization (PSO)

2.6 Confusion Matrix

A table called a confusion matrix lists the distinction between appropriate test sets and inappropriate test sets. The classification system's actual and anticipated classifications are detailed in the confusion matrix [19]. Based on how many test items are given in the table that predict true and false, the classification model is evaluated using the confusion matrix [20].

Table 1's confusion matrix covers two classification classes. It is possible to determine precision using a confusion matrix for categorization components, specificity, positive and negative predictive values, and other metrics [21].

Table 1. Confusion Matrix

		Predicted Values	
		Positive	**Negative**
Actual Values	**Positive**	TP (True Positive)	FN (False Negative)
	Negative	FP (False Positive)	TN (True Negative)

Equation 11, 12 and 13 can be used to determine precision, accuracy, and recall.

$$Accuracy \ = \frac{TP+TN}{Total\ all\ data} \tag{11}$$

DOI: https://doi.org/10.29207/resti.v7i4.4893
Creative Commons Attribution 4.0 International License (CC BY 4.0)

$$Precision \quad = \quad \frac{TP}{TP+FP} \tag{12}$$

$$Recall \quad = \quad \frac{TP}{TP+FN} \tag{13}$$

3. Results and Discussions

Data mining is used to forecast Covid-19 patient recovery in Indonesia. The highest Covid-19 patient death rate was seen in Indonesia on Saturday, April 4, 2020, according to researchers at about 9.11% [22]. Because of this, it's critical to understand the actual variables that influence Covid-19 patients' chances of survival and prognosticate their likelihood of recovery. Naive Bayes, K-Nearest Neighbors, and Logistic Regression are the three classification techniques used in this study. Numerous research have contrasted these three methodologies, including one that examined KRL transportation sentiment using Naive Bayes, KNN, and Decision Tree [23], studies on the efficiency of Naive Bayes for web page classification training by comparing logistic regression, NB, and random forest methods to predict animal survival [24]. The evaluations of the performance of NB, KNN, or logistic regression techniques in any of these investigations did not reveal a high incidence of Covid-19 in patients. This study assesses the efficacy of three methods for predicting the rate of recovery for Covid-19 patients in Indonesia. The data mining tool used by researchers is Orange Software version 3.25.0. The goal of this study is to forecast the patient's rate of recovery by taking into account a variety of factors. Age and gender are the independent factors in this study. The most accurate performance analysis techniques used in this study include NB, logistic regression, and KNN methods. The findings also indicate that KNN, with an accuracy of 0.750, outperforms both NB and Logistic Regression, which both have accuracy values of 0.703. KNN also has the greatest value of 0.750 when compared to NB and Logistic Regression, which both have an accuracy level of 0.700. KNN is still the best model with a third recall value of 0.750 when compared to the two reference models, which have equal third recall values of 0.708 [25].

The next study used the Naive Bayes approach to forecast the prognosis of Indonesian COVID-19 patients. The primary objective of the researchers in this study was to determine the most efficient course of treatment for COVID-19 patients based on the information currently available in order to be able to provide the most acceptable course of treatment for COVID-19 patients. As a result, there are numerous COVID-19 therapeutic choices, many of which are supported by scientific data. According to the study's findings, the accuracy of the Naive Bayes method's prediction of COVID-19 patients' recovery was 96.51%, the accuracy of success (Yes) 100 and failure (No) 95.71%, and the sensitivity of success (Yes). 84.21% and Fail (No) 100%. The study's findings show that the Naive Bayes approach computation has an accuracy of 96.51% for recovering from COVID-19, which means that the results predict the success and failure of patient treatment and become a reference for further research [26].

In this study, using experimental research to determine accuracy optimization between the Naïve Bayes algorithm method and PSO-based Naive Bayes. In research on predicting patient recovery data by entering test data derived from training data. Where the final data is obtained after preprocessing. The results with preprocessing turned out to be 450 lines of data taken from the kawalcovid19.com website. The processed dataset is divided into 30% for testing and 70% for training, so the cross validation method is used to test the accuracy level. AUC and Confusion Matrix analysis to measure accuracy produce scores for precision, accuracy, and recall.

After obtaining the data to be processed, testing is carried out using Google colab software to test the Naive Bayes and PSO-based Naive Bayes algorithms. The results of this study will determine which strategy is more accurate between the Naive Bayes Algorithm (NB) and PSO-based Naive Bayes.

This study uses a patient dataset with roughly 450 rows of data in Table 2. This patient dataset consists of four attributes: Status with type category, Province with type category, and Gender with type numeric. The Status attribute is the goal as well (In Treatment, Recovered, Died).

Table 2. Patient Dataset Attributes

Attribute	Type
Gender	Category
Age	Numeric
Province	Category
Status	Category

For easy reading of gender, province, and state, the dataset for this study still comprises strings that need to be encoded into numbers using encoder tags.

Table 3 shows the encoder label 0 for female and 1 for male are used as the signs for the Gender attribute.

Table 3. Gender encoder labels

Gender	Code
Female	0
Male	1

The Province attribute in Table 4 uses encoder labels 12 for North Sumatra, 14 for Riau, 18 for Lampung, 21 for Riau Islands, 31 for DKI Jakarta, 32 for West Java, 33 for Central Java, 34 for DI Yogyakarta, 35 for East Java, 36 for Banten, 51 for Bali, 61 for West Kalimantan, 64 for East Kalimantan, 71 for North Sulawesi, 73 for South Sulawesi, and 74 for South East Sulawesi.

DOI: https://doi.org/10.29207/resti.v7i4.4893
Creative Commons Attribution 4.0 International License (CC BY 4.0)

Table 4. Province encoder labels

Province	Code
North Sumatra	12
Riau	14
Lampung	18
Riau Islands	21
DKI Jakarta	31
West Java	32
Central Java	33
DI Yogyakarta	34
East Java	35
Banten	36
Bali	51
West Kalimantan	61
East Kalimantan	64
North Sulawesi	71
South Sulawesi	73
Southeast Sulawesi	74

The Status attribute in Table 5 uses Encoder label 1 for Recovered, 2 for Under Treatment and 3 for Died.

Table 5. Encoder label Status

Status	Code
Recovered	1
In Treatment	2
Died	3

3.1 Algoritma Naive Bayes

Table 6 illustrates the Naive Bayes algorithm is known to have an accuracy rate of 0.94% when used to calculate the accuracy of training data. This is based on the test results of patient recovery data that has been preprocessed.

Table 6. Naive Bayes Algorithm Accuracy

	Precision	Recall	F1-Score
1	1.00	0.14	0.25
2	0.94	1.00	0.97
3	0.00	0.00	0.00
Accuracy			0.94
Macro avg	0.65	0.38	0.41
Weighted avg	0.93	0.94	0.92

3.2 PSO-Based Naive Bayes

Table 7 illustrates PSO-based When used to determine the correctness of training data, Naive Bayes is known to have an accuracy rate of 0.96%. This is based on test results from preprocessed patient recovery data.

Table 7. PSO-Based Naive Bayes Accuracy

	Precision	Recall	F1-Score
1	1.00	0.25	0.40
2	0.96	1.00	0.98
3	0.00	0.00	0.00
Accuracy			0.96
Macro avg	0.65	0.42	0.46
Weighted avg	0.95	0.96	0.94

3.3 Experiment Results

Stages in the research are: The processed dataset is split into 30% for testing and 70% for training. The naive bayes algorithm and PSO-based naive bayes approach are then used to perform the computations; A parametric test employing the Naive Bayes approach was used to determine the accuracy in the first trial, which was 94.07%, precision of 14%, Recall 1%, and AUC 0.613; Additional research into the particle swarm-optimized naïve Bayesian approach. The test had a 95.56% accuracy rate, precision 25%, Recall 1% and AUC of 0.540.

Table 8 displays the experimental findings from Google Colab.

Table 8. Experiment Results

Algorithm	Accuracy	Precision	Recall	AUC
Naive bayes	94.07%	14%	1%	0.613
PSO based on Naive Bayes	95.56%	25%	1%	0.540

4. Conclusion

The 450 patient dataset used in this study, which was obtained from the website kawalcovid19.com, was processed using the Naive Bayes algorithm approach and Naive Bayes based on particle swarm optimization. After the second experiment using PSO-based Nave Bayes produced an accuracy of 95.56% or 96% and an AUC value of 0.540, the Patient Dataset Calculation utilizing the Nave Bayes Algorithm may produce results with an accuracy of 94.07% and the Area under cover (AUC) value of 0.613. The accuracy of PSO on the Patient dataset can therefore be improved, it can be said.

This leads to the conclusion that PSO can assist in computing patient data. This leads to the conclusion that PSO can increase the precision of patient datasets. The findings of this study have led experts to suggest doing tests with different kinds of data mining.

References

[1] World Health Organization, “Coronavirus”, 2019. [Online]. Available: https://www.who.int/healthtopics/coronavirus.

[2] V. No and N. Mona, “Konsep Isolasi Dalam Jaringan Sosial Untuk Meminimalisasi Efek Contagious (Kasus Penyebaran Virus Corona Di Indonesia),” *J. Sos. Hum. Terap.*, vol. 2, no. 2, pp. 117–125, 2020, doi: 10.7454/jsht.v2i2.86.

[3] Widiyani, R., "Latar Belakang Virus Corona,Perkembangan hingga Isu Terkini", 2020. Retrieved from detik News: https://news.detik.com/berita/d4943950/latar-belakangviruscoronaperkembanganhingga-isu-terkini Nuha Medika

[4] N. Salma, T. Ichsan, M. D. Sa’adillah, Z. W. Budiawan,and D. Popon, “Implementation of K-Nearest Neighbor to Predict the Chances of COVID-19 Patients' Recovery,” *International Conference on Wireless and Telematics (ICWT)*, 2022, doi: 10.1109/ICWT55831.2022.9935435

[5] J. S. Widjaya, D. Agushinta R, and S. R. Puspita Sari, “Sistem Prediksi Jumlah Pasien Covid-19 Menggunakan Metode Trend Least Square Berbasis Web,” *Sistemasi*, vol. 10, no. 1, p. 39, 2021, doi: 10.32520/stmsi.v10i1.1036.

[6] Fei, S. W., Miao, Y. B., & Liu, C. L., "Chinese Grain Production Forecasting Method Based on Particle Swarm Optimization-based Support Vector Machine," Recent Patents on

DOI: https://doi.org/10.29207/resti.v7i4.4893
Creative Commons Attribution 4.0 International License (CC BY 4.0)

Engineering., vol 3, no. 1, pp. 8-12, 2009.

[7] J. Ipmawati, Kusrini, and E. Taufiq Luthfi, "Komparasi Teknik Klasifikasi Teks Mining Pada Analisis Sentimen," *Indones. J. Netw. Secur.*, vol. 6, no. 1, pp. 28–36, 2017.

[8] Bramer, M., "Principles of Data Mining", 2007. London: Springer.

[9] H. Muhamad, C. A. Prasojo, N. A. Sugianto, L. Surtiningsih, and I. Cholissodin, "Optimasi Naïve Bayes Classifier Dengan Menggunakan Particle Swarm Optimization Pada Data Iris," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 4, no. 3, p. 180, 2017, doi: 10.25126/jtiik.201743251.

[10] Suyanto, "Data Mining untuk Klasifikasi dan Klasterisasi Data",2017. Bandung : Informatika Bandung.

[11] J. Žižka, F. Dařena, and A. Svoboda, *Introduction to Text Mining with Machine Learning*. 2019. doi: 10.1201/9780429469275-1.

[12] A. Mukminin and D. Riana, "Komparasi Algoritma C4 . 5 , Naïve Bayes Dan Neural Network Untuk Klasifikasi Tanah," *J. Inform. Univ. Bina Sarana Inform.*, vol. 4, no. 1, pp. 21–31, 2017, [Online]. Available: https://ejournal.bsi.ac.id/ejurnal/index.php/ji/article/view/1002

[13] D. Xhemali, C. J. Hinde, and R. G. Stone, "Naive Bayes vs. Decision Trees vs. Neural Networks in the Classification of Training Web Pages," *Int. J. Comput. Sci.*, vol. 4, no. 1, pp. 16–23, 2009, [Online]. Available: http://cogprints.org/6708/

[14] J. J. Aripin, "Penerapan Algoritma Naive Bayes Untuk Mengklasifikasi Data Nasabah Asuransi pada BPR Pantura," 2019, [Online]. Available: https://repository.nusamandiri.ac.id/index.php/repo/viewitem/13890

[15] M. O. Okwu and L. K. Tartibu, "Particle Swarm Optimisation," *Stud. Comput. Intell.*, vol. 927, pp. 5–13, 2021, doi: 10.1007/978-3-030-61111-8_2.

[16] S.-W. Fei, Y.-B. Miao, and C.-L. Liu, "Chinese Grain Production Forecasting Method Based on Particle Swarm Optimization-based Support Vector Machine," *Recent Patents Eng.*, vol. 3, no. 1, pp. 8–12, 2009, doi: 10.2174/187221209787259947.

[17] S. Sumathi and S. Paneerselvam, *Computational Intelligence Paradigms Theory and Applications*. 2010.

[18] D. F. Shiau, "A hybrid particle swarm optimization for a university course scheduling problem with flexible preferences," *Expert Syst. Appl.*, vol. 38, no. 1, pp. 235–248, 2011, doi: 10.1016/j.eswa.2010.06.051.

[19] R. Kohavi, P. Langley, and Y. Yun, "The utility of feature weighting in nearest-neighbor algorithms," *Proc. Ninth Eur. Conf. Mach. Learn.*, no. September 1997, pp. 85–92, 1997, [Online]. Available: http://www.isle.org/~langley/papers/diet.ecml97.pdf

[20] Gorunescu, F., "Data Mining: Concepts and Techniques", 2011. Verlag berlin Heidelberg: Springer.

[21] M. F. Akay, "Support vector machines combined with feature selection for breast cancer diagnosis," *Expert Syst. Appl.*, vol. 36, no. 2 PART 2, pp. 3240–3247, 2009, doi: 10.1016/j.eswa.2008.01.009.

[22] M. D. H. Rahiem, "Technological barriers and challenges in the use of ICT during the COVID-19 emergency remote learning," *Univers. J. Educ. Res.*, vol. 8, no. 11B, pp. 6124–6133, 2020, doi: 10.13189/ujer.2020.082248.

[23] N. Tri Romadloni, I. Santoso, and S. Budilaksono, "Perbandingan Metode Naive Bayes, Knn Dan Decision Tree Terhadap Analisis Sentimen Transportasi Krl Commuter Line," *J. IKRA-ITH Inform.*, vol. 3, no. 2, pp. 1–9, 2019.

[24] E. M. M. van der Heide, R. F. Veerkamp, M. L. van Pelt, C. Kamphuis, I. Athanasiadis, and B. J. Ducro, "Comparing regression, naive Bayes, and random forest methods in the prediction of individual survival to second lactation in Holstein cattle," *J. Dairy Sci.*, vol. 102, no. 10, pp. 9409–9421, Oct. 2019, doi: 10.3168/JDS.2019-16295.

[25] M. R. Romadhon and F. Kurniawan, "A Comparison of Naive Bayes Methods, Logistic Regression and KNN for Predicting Healing of Covid-19 Patients in Indonesia," *3rd 2021 East Indones. Conf. Comput. Inf. Technol. EIConCIT 2021*, pp. 41–44, 2021, doi: 10.1109/EIConCIT50028.2021.9431845.

[26] P. T. A. Barus Okky, "Prediksi kesembuhan pasien COVID-19 di Indonesia melalui terapi menggunakan metode Naïve Bayes," *J. Inf. Syst. Dev.*, vol. 6, no. 2, pp. 59–66, 2021, [Online]. Available: https://ejournal.medan.uph.edu/index.php/isd/article/view/460

DOI: https://doi.org/10.29207/resti.v7i4.4893
Creative Commons Attribution 4.0 International License (CC BY 4.0)