Features Selection based on Enhanced KNN to Predict Raw Material Needs on PT. SANM

Siti Aisyah Naili Mutia[1*)], Tjong Wan Sen[2]
[1]Program Studi Teknik Informatika, Fakultas Komputer, President University
[2]Program Studi Teknik Informatika, Fakultas Komputer, President University
Email: [1]sitiaisyah.mutia@student.president.ac.id, [2]wansen@president.ac.id

Abstract – Raw material inventory must be able to meet production needs. So it is necessary to plan / predict raw material needs in the following month to determine the raw material inventory. Currently PT. SANM uses a manual counting method, the expenditure of raw materials for six months, then deducts the current raw material inventory. As a result, there are raw materials that are over order or lacking, which causes production to be constrained. The manual calculation method is not effective enough to meet the raw material inventory. In this research, the researcher proposes an algorithm which is contained in Data Mining, that is Enhanced KNN using GWO to predict raw material needs. Because GWO and Enhanced KNN algorithms give the results are easy to understand, have good accuracy compared to other machine learning methods, can cover the trapped problem from KNN traditional and capable of improving the accuracy using feature selection method. The method used in this study is to compare Enhanced KNN with and without GWO that gives a significant increase in the accuracy value by 16.5%, from 44.6% to 61.1%.

Keywords – Data mining, Classification, Enhanced K-Nearest Neighbor, Feature Selection

I. INTRODUCTION

PT. SANM is a company that operates in the field of eyeglass lenses, which includes medical devices. There are many types of raw materials, ranging from different segments, different indexes, white or color and diameter. According to the type of segment, the lens can be divided into three, namely a circular segment or called a kryptok lens, a semicircular segment or called a flattop and does not have a segment called single vision. According to the index there are index lenses 1.50, 1.53, 1.56, 1.57, 1.60, 1.67, 1.74. According to the color there are white, photochromic gray, photochromic brown, nupolar green, nupolar gray and also anti-radiation. Meanwhile, according to the diameter there are lenses with a diameter of 60, 65, 70, 72, 74, 76.

PT. SANM only carries out production according to orders or what is usually called a business process driven by order. When a customer places an order with various types of lenses, spherical size, cylinder or add, then the production process will be carried out. So that there is no production planning. Currently, to order raw materials, the manual calculation method is still used. The raw material for this lens does not have an expiration date, but if it is stored in an inappropriate manner and time, the raw material can no longer be used because it is out of standard or damaged.

To avoid the problem of shortage or excess of raw materials, it is necessary to classify the prediction of raw material needs. One of the classification algorithms that can be used is KNN[1].

Estimates for predicting disturbances from past data/events in the industrial sector using KNN. This study also explains predictions according to supervised data about changes in the characteristics of disorders that have occurred[2]. The classification results using KNN are easy to understand and have good accuracy compared to other

machine learning methods[3] or deep learning with newer approach using speech[4] or image[5] as input. The advantage of the KNN method is that it is effectively applied to large amounts of data and is resilient to noise training data, which is data that has the farthest range of values compared to other data but can disrupt the existing data structure. In addition to advantages, KNN also has weaknesses, namely less than optimal in determining the value of k which is the number of nearest neighbors and must determine the attribute to be selected or feature selection in order to get the best results, the presence of outliers in the training data and sensitive to the value of k (there are no clear rules). to determine the optimal value of k). Despite its simplicity and high efficiency, traditional KNNs can easily get stuck[6].

To overcome these shortcomings, an improvement solution is needed to optimize the KNN classification. To overcome these shortcomings, an improvement solution is needed to optimize the KNN classification. One solution that can be applied is to perform optimization by applying EKNN[6] and adding a feature selection process using the GWO algorithm[7][8][9]. Based on the above studies, the researcher chose the EKNN method to classify and predict the raw material needs of the coming month and used GWO as a feature selection method to improve EKNN performance.

II. RESEARCH METHODOLOGY

The research stages are outlined in Figure 1. which describes the research process that will be taken.

In this research, there are several stages carried out, including: Preparation, includes background creation, problem identification, problem formulation, objectives, problem boundaries, research benefits and novelty. Study literature by conducting a literature review on predictions using EKNN. Data collection, namely data collection by

collecting sales data for 2018-2020. Details of the dataset used are shown in Table 1.
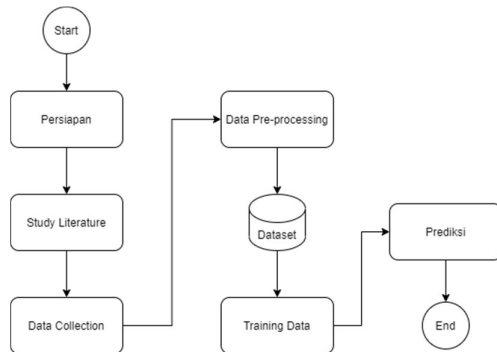

Figure 1 Research Stages

Table 1 Dataset Detail

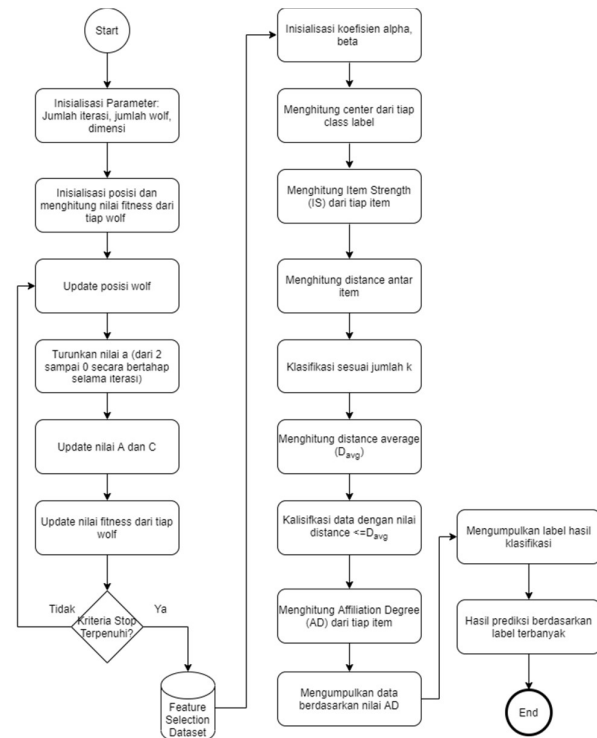| Dataset		
Data	**Attribute**	**Target**
60	34	3

Figure 2 Training Model

Data pre-processing, namely removing data noise such as duplicate data, empty values, transforming data from time series into per material code and data normalization. Data training will be carried out using the EKNN and GWO algorithms. The researcher proposes the EKNN algorithm because it is easy to understand and provides better accuracy results than other machine learning methods[3] and is able to overcome the problem of being trapped in traditional KNN[6]. Then the use of GWO as a feature selection method to increase the accuracy value on

EKNN. The experiment in the first step will be trained using the EKNN algorithm and then using EKNN-GWO to get a comparison of the results of accuracy, precision and recall values, as shown in Figure 2.

A. *Enhanced KNN*

The Enhanced KNN algorithm is a development algorithm from K-Nearest Neighbor. Traditional KNN is easily trapped in several situations, such as there are two classes that have the same number of neighbor classifications/series. Examples of traps that may occur in traditional KNN can be seen in Figure 3.
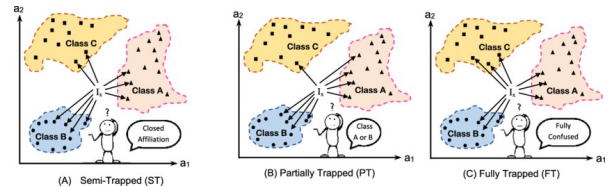

Figure 3 Different-types-of-KNN-trapping [6]

To overcome this weakness, another parameter is needed to determine the closest neighbor. Like the traditional KNN, the EKNN deals with its closest neighbors with the item being tested in the feature room.

Input: Training dataset, center, α, β
Output: Item Strength (IS)
Steps:
For each class label (cx € CL) do:
     Project the training examples in the n dimensional feature space
Next
For each class (cx € CL) do:
For each item (Ij € cx) do:
     Calculate the strength of item Ij as:

$$IS(I_j) = \frac{\left[\alpha * \sum_{j \neq k} \frac{1}{Dis(I_j, I_k)} + \frac{\beta}{Dis(I_j, C)} \right]}{2}$$

   Next
Next

Figure 4 EKNN Phase Training Algorithm

However, these neighbors will have different strengths according to the level of closeness to the item being tested and the neighbor's membership level with their class, which is called Item Strength (IS). The EKNN is implemented in two stages, namely; the training and testing phase[6].

In the training phase, all strength is counted. Where item strength is a measure for the relationship level of the item with its hosting class. Assuming *n* features, *m* as the target class, to calculate the strength of each item. EKNN training algorithm can be seen in Figure 4 and EKNN training phase flow can be seen in Figure 5.

First, all items are projected into the dimension space feature. Then, the center of each class containing examples in n dimensionless feature spaces can be solved using equation (1).

$$C = \left\{ \frac{\sum_{q=1}^t V_q^1}{t}, \frac{\sum_{q=1}^t V_q^2}{t}, \ldots\ldots, \frac{\sum_{q=1}^t V_q^n}{t} \right\} \quad (1)$$
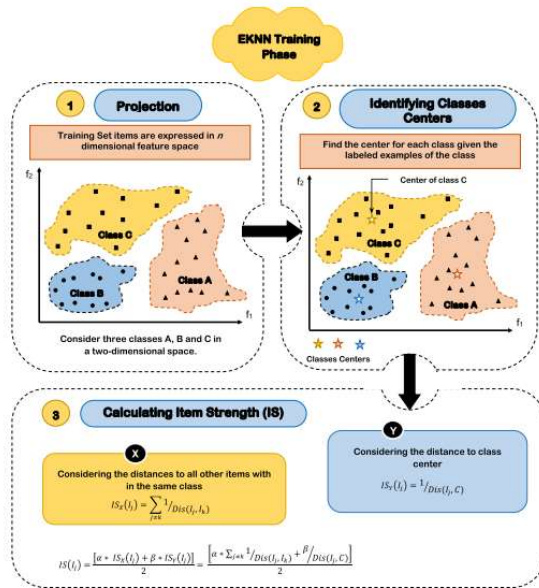
Figure 5 EKNN Training Phase [6]

Where C is the class center in the n-dimensional feature space, t is the number of instances in the class, and V_q^i is the i-th dimension value of the q-th example. Then, the strength of item I_j can be calculated using equation (2).

$$IS(I_j) = \frac{[\alpha * IS_X(I_j) + \beta * IS_Y(I_j)]}{2} \qquad (2)$$

Where $IS(I_j)$ is the power of the item I_j, which is the weighted average of the two values. The first measures strength of an item based on its proximity to another item (example) in its class and is denoted as; $IS_X(I_j)$. On the other hand, the value further considers the proximity of the item to the center of the class as an indication of its degree of affiliation with the class and according to its strength, which is denoted as; $IS_Y(I_j)$. As in the previous equation, α and β are weighting factors that express the relative impact of $IS_X(I_j)$ and $IS_Y(I_j)$, where $0<\alpha\leq 1$ and $0<\beta\leq 1$. In general, $IS_X(I_j)$ and $IS_Y(I_j)$ can be calculated using equations (3) and (4), respectively.

$$IS_X(I_j) = \sum_{j\neq k} \frac{1}{Dis(I_j,I_k)} \qquad (3)$$

$$IS_Y(I_j) = \frac{1}{Dis(I_j,C)} \qquad (4)$$

Where $IS_X(I_j)$ and $IS_Y(I_j)$ are the strength of the item considering the closeness of each class item and class center.

$Dis (I_j, I_k)$ is the Euclidean distance between item I_j and I_k, and $Dis(I_j, C)$ is the Euclidean distance between item I_j and the class center. Calculating the distance between two points p_x and p_y in a feature space of dimension n can be calculated using equation (5).

$$Dis(p_x,p_y) = \sqrt{\sum_{i=1}^{n}\left(p_x^i - p_y^i\right)^2} \qquad (5)$$

Where $p_x i$ and p_y^i are the i-th dimensional values of the p_x and p_y points in the n-dimensional feature space, respectively. Then, in place of (2), the power of item I_j can be calculated using equation (6).

$$IS(I_j) = \frac{\left[\alpha * \sum_{j\neq k}\frac{1}{Dis(I_j,I_k)} + \frac{\beta}{Dis(I_j,C)}\right]}{2} \qquad (6)$$

An important issue is how to estimate the optimal values of α and β. As tunable parameters, the optimal values of α and β can be calculated empirically by assigning different values in a predetermined scenario, then calculating the resultant accuracy of the EKNN classifier by considering a series of test items. The optimal α and β values are those that provide maximum classification accuracy. The suggested scenario is to start with initial values of α and β, for example $\alpha = 0$ and $\beta = 0$, the values of α and β will be increased using a constant positive step ξ keeping the values of α and β greater than 0 and less than or equal to 1.

In the testing phase, the proposed EKNN, consider ring a target class set m to CL = {c 1, c 2, ... cm}. EKNN testing algorithm can be seen in Figure 6 and EKNN testing phase flow can be seen in Figure 7.

Initially, the item under test is projected in a feature of n dimensional space. Then, the closest K-neighbor was identified. The distance from the tested item Ij to each k nearest neighbor is calculated, then the average $distance(D_{avg})$ is then calculated. Circular spheres are identified whose radius is equal to D_{avg}. Only samples in the identified environment will be considered for classifying new items. Affiliation Degree (AD) items tested for each class are calculated using the equation (7).

$$AD_x(I_j) = \sum_{\forall I_k \in S_x} D_k * IS(I_k) \qquad (7)$$

Input: Training dataset, IS
Output: Classifying the tested item Ij to one of the available target class
Steps:
1. The project the tested item into the n dimensional feature space.
2. Pick the nearest K items near the test point Ij.
3. Computer the average distance Davg from Ij and each of the K nearest examples
4. Pick items located in the circle with diameter Davg
5. Calculate the Affiliation Degree (AD) of the tested items to each classes;
$$AD_x(I_j) = \sum_{\forall I_k \in S_x} D_k * IS(I_k)$$
6. Identify the target class of the tested item;
$$Target\_Class(I_j) = \underset{\forall C_x \in CL}{argmax} AD_x(I_j)$$
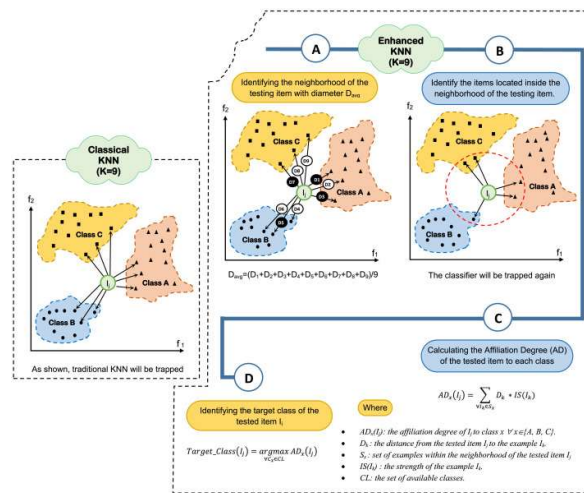
Figure 6 EKNN Algorithm Testing Phase

Figure 7 EKNN Testing Phase [6]

Where AD_xI_j is the level of affiliation of item I_j to tested class x∀ x ∈{A B C}. D_k is the distance from the tested problem I_j to the example I_k. S_x is the set of examples in the environment of the items tested I_j. And $IS(I_k)$ is the strength of the I_k example. Finally, the item under test is targeted to the class that has the maximum level of affiliation as illustrated in equation (8).

$$Target\_Class(Ij) = \frac{argmaxADx(Ij)}{\forall CxCL} \qquad (8)$$

Where CL is the set of the target class under consideration.

B. Feature Selection

Feature selection is an optimization problem that plays an important role in dealing with classification problems. It is the process of selecting an optimal subset of features from a data set so that the classifier can obtain better accuracy and/or reduced computational load. However, removing irrelevant features is challenging and time consuming due to the large search space and the relationship between features[10]. The traditional FS method has the disadvantage of nesting effects and computational costs. To solve this problem, population-based optimization algorithms, such as gray wolf optimization (GWO)[8], particle swarm optimization (PSO)[11], genetic algorithm (GA), genetic programming (GP), ant colony optimization (ACO), brain storm optimization (BSO) are used. and harmony search (HS)[12].

C. Grey Wolf Optimization (GWO)

Gray Wolf Optimization (GWO) is a population-based metaheuristic algorithm that simulates leadership hierarchies and the mechanism of hunting grey wolves in nature discovered by Mirjalili in 2014[8]. The gray wolf (Gray Wolf) is a carnivorous animal that is at the highest level in the food chain, otherwise known as the Apex Predator. Gray wolves live in groups, where each group consists of 5-12 individuals[7]. The interesting thing about gray wolves is their social level. The gray wolf is divided into four social levels, as in Figure 8.

The first level is called Alpha (α), which is the leader of a group where they are female and male wolves. The first level wolf has the most right to make all decisions, regarding hunting activities, hunting time, resting place and so on.
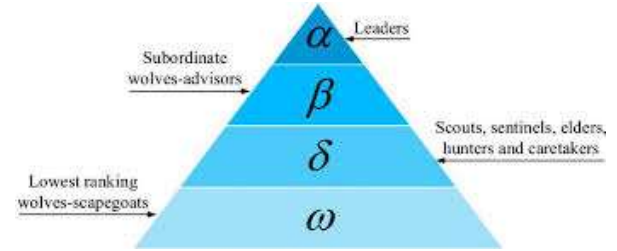

Figure 8 Gray Wolf Level [8]

The wolf's orders of the first level will be obeyed by all wolves that are in the lower levels. The second level wolf is also called Beta (β), which is in charge of assisting the first level wolf in making decisions, as an advisor and as a substitute for the first level wolf when it dies. The second level wolf can be male or female. The third level wolf or Delta (δ) is usually called subordinate. The third level wolf has several categories; scouts are in charge of guarding the territory and giving warnings when in danger, sentinels as group protectors, elders who are experienced and have the potential to replace first and second level wolves, hunters who help first and second level wolves in hunting and provide food, and caretakers as carers for a wolf that is sick or injured.ω

Apart from the social level, another interesting thing about the gray wolf is group hunting. At the first level GWO is considered the best solution.

The steps of hunting and mathematical models on GWO are as follows:

Step 1: Encircling prey

$$\vec{D} = \left| \vec{C} \times \overrightarrow{X_p}(t) - \vec{X}(t) \right| \qquad (9)$$

$$\vec{X}(t+1) = \overrightarrow{X_p}(t) - \vec{A} \times \vec{D} \qquad (10)$$

Where t denotes the current iteration, $\vec{A}$ and $\vec{C}$ is the vector coefficient, $\overrightarrow{X_p}$ is the vector of the prey position, and $\vec{X}$ denotes the vector of the position of the gray wolf. Where $\vec{A}$ and $\vec{C}$ can be calculated by the equation below.

$$\vec{A} = 2\vec{a} \times \vec{r}_1 - \vec{a} \qquad (11)$$

$$\vec{C} = 2 \times \vec{r}_2 \qquad (12)$$

Where the component is derived linearly from 2 to 0 during iteration and r1, r2 are random vectors in [0,1].$\vec{a}$

Step 2: Hunting: hunting activities are usually guided by Alpha level wolves. Beta and Delta might participate in the hunt every now and then. In the mathematical model of the gray wolf hunting behavior, it is assumed that Alpha, Beta, and Delta have better knowledge of the potential location of the prey. The first three best solutions are saved and the other wolves must update

their position according to the position of the best seeker wolf as shown in the equation below.

$$\vec{D}_\alpha = \left| \vec{C}_1 \times \vec{X}_\alpha - \vec{X} \right|$$
$$\vec{D}_\beta = \left| \vec{C}_2 \times \vec{X}_\beta - \vec{X} \right|$$
$$\vec{D}_\delta = \left| \vec{C}_3 \times \vec{X}_\delta - \vec{X} \right| \qquad (13)$$

$$\vec{X}_1 = \vec{X}_\alpha - \vec{A}_1 \times \left(\vec{D}_\alpha \right),$$
$$\vec{X}_2 = \vec{X}_\beta - \vec{A}_2 \times \left(\vec{D}_\beta \right)$$
$$\vec{X}_3 = \vec{X}_\delta - \vec{A}_3 \times \left(\vec{D}_\delta \right) \qquad (14)$$

$$\vec{X}(t+1) = \frac{\vec{X}_1 + \vec{X}_2 + \vec{X}_3}{3} \qquad (15)$$

Step 3: Attacking the prey (exploitation): The gray wolf completes the hunt by attacking its prey when it stops moving. The vector is a random value in the interval [-2a, 2a], where a is lowered from 2 to 0 during iteration. When the wolf attacks the prey, that represents a process of exploitation.$\vec{A} \left| \vec{A} \right| < 1$

Step 4: Looking for prey (exploration): The roaming process in GWO is applied according to different positions and positions to find prey and gather to attack the prey. The exploration process is modeled mathematically by utilizing A with a random value greater than 1 or less than -1 to oblige the seeker group to deviate from its prey. When | A |> 1, the wolf is forced to deviate from its prey to become a fitter prey.

D. Validation

One alternative approach to "train and test" that is often adopted in some cases (and some regardless of size) is called K-Fold Cross Validation.[13] by testing the amount of error in the test data[14]. We use the k-1 sample for training and the remaining 1 sample for testing. For example, there are 10 data subsets, we use 9 subsets for training and the remaining 1 subset for testing. There are 10 training times where in each training there are 9 data subsets for training and 1 subset is used for testing. After that, the average error and standard deviation of error are calculated[14]. Each k part is in turn used as the test set and the other k -1 parts are used as the training set[15].

E. Measurement Stages

Confusion matrix is a dataset having only two classes, one class as positive and the other class as negative[15]. This method uses a configuration matrix model table.

Table 2 Confusion Matrix Model[16]

		Prediction		Total
		-1 (Negative)	+1 (Positive)	
Exa mple	-1 (Negative)	p	q	p + q
	+1 (Positive)	u	v	u + v

s	Total	p + u	q + v	m

Where p is the number of correct predictions that the instance will be negative. q is the number of incorrect predictions that the instance will be positive. u is the number of incorrect predictions that the instance will be negative. v is the number of correct predictions that the instance will be positive.

Here is the confusion matrix model equation:

a. The accuracy value (acc) is the proportion of the number of correct predictions. Can be calculated using the equation:

$$acc = \frac{p+v}{(p+q+u+v)} = \frac{p+v}{m} \qquad [9]$$

b. The recall or true positive rate (tp) is the proportion of correctly classified positive cases calculated using the equation:

$$tp = \frac{u}{u+v} \qquad [10]$$

c. Precision (p) is the proportion of correct positive case predictions, which are calculated using the equation:

$$p = \frac{v}{q+v} \qquad [11]$$

III. RESULTS AND DISCUSSION

Data collection was obtained from PT SANM. The dataset used in this study is sales data for the years 2018-2020. The data contains 60 rows and 34 attributes among other BL01, BL01BB, BL01P, BL01PUV, BL02, BL02BB, BL02PUV, BL05, BL05BB, BL05BB1.6, BL05BL, BL05D, BL05DW, BL05L, BL05LB, BL05M, BL05M1.6, BL05M1.67, BL05NU, BL05NUG, BL05O, BL05OB, BL05PG, BL05PGUV, BL05PGUV1.6, BL05PGUV1.67, BL05SSS, BL05YVII, BL05YVII1.67, BL06PB, IMPACT1.57, SV1.67, SV1.74 and TRIVE1.53. The data pre-processing stage used in this study is to set aside unused attributes and data transformation. Data transformation is used to change the number class data type to nominal. This research classification method uses Enhanced KNN based[6].

One alternative approach to "train and test" that is often adopted in some cases (and some regardless of size) is called K-Fold Cross Validation.[17]. The results of the accuracy of each fold are shown in Table 3.

Table 3 Test Result Accuracy Value

Fold	EKNN	GWO-EKNN
1	33%	83%
2	50%	33%
3	83%	33%
4	50%	83%
5	50%	0%
6	50%	33%
7	33%	83%

8	17%	83%
9	40%	100%
10	40%	80%
Average	44.6%	61.1%

From 10 tests, the average accuracy of the EKNN method is 44.6%, with the highest accuracy value of 83% and the lowest accuracy value of 17%. Then the test was carried out using the GWO-EKNN method with an average accuracy value of 61.1%, with the highest accuracy value of 100% and the lowest accuracy value of 0%.

Based on the accuracy value data between the EKNN and GWO-EKNN methods, the results show that the GWO-EKNN accuracy value is better than the EKNN with an average accuracy value of 61.1% with a difference of 16.5%. The following is a graphical comparison result to see a clearer difference in accuracy values.

Figure 9 Comparison of the accuracy value of EKNN with GWO-EKNN

In Figure 9, the highest average accuracy value is found in the GWO-EKNN algorithm. In the experiments that have been carried out, it can be concluded that the experimental results are in the table below:

Table 4 Comparison of accuracy, precision and recall values

Method	Average Accuracy	Average Precision	Average recall
EKNN	44.6%	23.3%	44.4%
GWO-EKNN	61.1%	52.5%	62.8%

Based on Table 4, it shows that the best experimental results are in the GWO-EKNN algorithm with an accuracy value of 61.1%, a precision value of 52.5% and a recall value of 62.8%. The second method is the EKNN algorithm with an accuracy value of 44.6%, a precision value of 23.3% and a recall value of 44.4%.

IV. CONCLUSION

Based on the research that has been done to improve the performance of EKNN in predicting raw material needs by selecting features using GWO, it can be concluded that GWO is able to increase EKNN performance by 16.5%, from 44.6% to 61.1%. Based on the GWO feature selection method, it was found that the features that can increase the accuracy of the prediction of

raw material requirements include BL01, BL01BB, BL01P, BL01PUV, BL02, BL02PUV, BL05BB, BL05BB1.6, BL05BL, BL05DW and Target. So that the GWO-EKNN method can predict the raw material needs more precisely on the sales dataset of PT. SANM in 2018-2020.

Based on the above conclusions, for future research it is recommended to use other metaheuristic feature selection methods, such as: PSO, GA, ACO, BCO, and others as well as using optimization of metaheuristic methods.

REFERENCES

[1] R. Arian, A. Hariri, A. Mehridehnavi, A. Fassihi, and F. Ghasemi, "Protein kinase inhibitors' classification using K-Nearest neighbor algorithm," *Comput. Biol. Chem.*, vol. 86, no. April, p. 107269, 2020, doi: 10.1016/j.compbiolchem.2020.107269.

[2] F. Borghesan, M. Chioua, and N. F. Thornhill, "Forecasting of process disturbances using k-nearest neighbours, with an application in process control," *Comput. Chem. Eng.*, vol. 128, no. 675215, pp. 188–200, 2019, doi: 10.1016/j.compchemeng.2019.05.009.

[3] D. A. Adeniyi, Z. Wei, and Y. Yongquan, "Automated web usage data mining and recommendation system using K-Nearest Neighbor (KNN) classification method," *Appl. Comput. Informatics*, vol. 12, no. 1, pp. 90–108, 2016, doi: 10.1016/j.aci.2014.10.001.

[4] T. W. Sen, "Voice Activity Detector for Device with Small Processor and Memory," *ICSECC 2019 - Int. Conf. Sustain. Eng. Creat. Comput. New Idea, New Innov. Proc.*, pp. 212–217, 2019, doi: 10.1109/ICSECC.2019.8907081.

[5] T. W. Sen, S. Suakanto, A. M. Siregar, and A. L. P. Localization, "License Plate Localization for Low Computation Resources Systems Using Raw Image Input and Artificial Neural Network," vol. 15, no. 1, pp. 39–44, 1858.

[6] W. M. Shaban, A. H. Rabie, A. I. Saleh, and M. A. Abo-Elsoud, "A new COVID-19 Patients Detection Strategy (CPDS) based on hybrid feature selection and enhanced KNN classifier," *Knowledge-Based Syst.*, vol. 205, p. 106270, 2020, doi: 10.1016/j.knosys.2020.106270.

[7] M. A. Oktaviani, R. S. Wibowo, and N. K. Aryani, "Aliran Daya Optimal dengan Efek Katup Menggunakan Grey Wolf Optimization," *J. Tek. ITS*, vol. 7, no. 2, 2019, doi: 10.12962/j23373539.v7i2.30906.

[8] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey Wolf Optimizer," *Adv. Eng. Softw.*, vol. 69, pp. 46–61, 2014, doi: 10.1016/j.advengsoft.2013.12.007.

[9] M. H. Nadimi-Shahraki, S. Taghian, and S. Mirjalili, "An improved grey wolf optimizer for solving engineering problems," *Expert Syst. Appl.*, vol. 166, p. 113917, 2021, doi: 10.1016/j.eswa.2020.113917.

[10]   J. Gholami, F. Pourpanah, and X. Wang, "Feature selection based on improved binary global harmony search for data classification," *Appl. Soft Comput. J.*, vol. 93, p. 106402, 2020, doi: 10.1016/j.asoc.2020.106402.

[11]   R. D. Liklikwatil, E. Noersasongko, and C. Supriyanto, "Optimasi K-Nearest Neighbor Dengan Particle Swarm Optimization Untuk Memprediksi Harga Komoditi Karet," *e-Jurnal JUSITI (Jurnal Sist. Inf. dan Teknol. Informasi)*, vol. 7–2, no. 2, pp. 172–182, 2018, doi: 10.36774/jusiti.v7i2.252.

[12]   N. Bharanidharan and H. Rajaguru, "Classification of dementia using harmony search optimization technique," *IEEE Reg. 10 Humanit. Technol. Conf. R10-HTC*, vol. 2018-Decem, pp. 1–5, 2019, doi: 10.1109/R10-HTC.2018.8629846.

[13]   L. S. Han and J. A. N. Nordin, "Predicting the stock price trends using a K-nearest neighbors-probabilistic model," *J. Theor. Appl. Inf. Technol.*, vol. 96, no. 18, pp. 6245–6255, 2018.

[14]   M. E. Lasulika, "Prediksi Harga Komoditi Jagung Menggunakan K-Nn Dan Particle Swarm Optimazation Sebagai Fitur Seleksi," *Ilk. J. Ilm.*, vol. 9, no. 3, pp. 233–238, 2017, doi: 10.33096/ilkom.v9i3.148.233-238.

[15]   B. Song, S. Tan, H. Shi, and B. Zhao, "Fault detection and diagnosis via standardized k nearest neighbor for multimode process," *J. Taiwan Inst. Chem. Eng.*, vol. 106, no. xxxx, pp. 1–8, 2020, doi: 10.1016/j.jtice.2019.09.017.

[16]   A. Sharma, P. Madhushri, V. Kushvaha, and A. Kumar, "Prediction of the Fracture Toughness of Silicafilled Epoxy Composites using K-Nearest Neighbor (KNN) Method," *2020 Int. Conf. Comput. Perform. Eval. ComPE 2020*, pp. 194–198, 2020, doi: 10.1109/ComPE49325.2020.9200093.

[17]   M. Bramer, *Principles of Data Mining*. Springer London, 2007.