Efficient Multi-Class Wind Speed Prediction Through Genetic Algorithm and Multi-Layer Perceptron Integration

Siti Aisyah
Department of Information Technology
Politeknik Negeri Malang
Malang, Indonesia
sitiaisyah4110@gmail.com

Vivi Nur Wijayaningrum
Department of Information Technology
Politeknik Negeri Malang
Malang, Indonesia
vivinurw@polinema.ac.id

Mamluatul Hani'ah
Department of Information Technology
Politeknik Negeri Malang
Malang, Indonesia
mamluatulhaniah@polinema.ac.id

Noprianto
Department of Information Technology
Politeknik Negeri Malang
Malang, Indonesia
noprianto@polinema.ac.id

Vipkas Al Hadid Firdaus
Department of Information Technology
Politeknik Negeri Malang
Malang, Indonesia
vipkas@polinema.ac.id

2024 International Conference on Electrical and Information Technology (IEIT) | 979-8-3315-1688-8/24/$31.00 ©2024 IEEE | DOI: 10.1109/IEIT64341.2024.10763047

***Abstract*—Wind energy is a top-rated and environmentally friendly renewable electricity source, particularly in tropical regions and coastal areas where wind speed is critical for wind power plant design. Wind turbines are significantly influenced by the stochastic, fluctuating, and uncertain nature of wind, which affects their performance. Current wind speed predictions often rely solely on meteorological parameters, leading to prediction errors, increased costs, and computational complexity. This study aimed to identify the most influential meteorological parameters in wind speed prediction. Parameter optimization was conducted using the Genetic Algorithm, and multi-class classification was applied using Multi-Layer Perceptron. Feature selection with the Genetic Algorithm employed binary representation to choose optimal features, with fitness calculations applied to multi-class classification using Multi-Layer Perceptron. The results indicated that the four optimal features affecting wind speed are air temperature, air pressure, wind direction, and dew point temperature. Model evaluation using these four optimal features yielded an accuracy of 56%, precision of 51%, recall of 56%, F1-Score of 46%, and ROC-AUC Score of 77%, with a computation time of only 10.6 seconds. The low F1-Score and accuracy, as indicated by sensitivity and specificity evaluations, were attributed to class imbalance, impacting the overall results. Further testing with adjusted data showed an average accuracy of 85%. Despite data imbalance challenges, using selected feature subsets significantly improved accuracy and reduced computation time in predicting wind speed compared to using all original features.**

Keywords*—*feature selection; meteorological; multi-class; wind speed

I. Introduction

Renewable energy is increasingly popular as an environmentally friendly solution for electricity generation, especially due to the limitations of fossil and nuclear fuel resources. One significant source of renewable energy is wind energy, which does not produce harmful emissions and is considered a clean energy source. The implementation of wind power technology has been carried out in several tropical or coastal areas, including Surabaya [1][2].

The utilization of renewable energy, such as wind energy, has become a global trend as a replacement for fossil fuel power plants [3]. However, the random, fluctuating, intermittent, and unpredictable characteristics of wind energy affect the stability of wind turbines and wind power grids. Factors such as wind speed, direction, temperature, altitude, and pressure impact the performance of wind power generation systems, resulting in significant uncertainty in wind power output [4].

Accurate wind speed prediction is crucial to protect integration systems, reduce operational costs, and enhance the stability of wind power generation systems [5]. Previous studies have explored various approaches to predicting wind speed using machine learning techniques. In 2017, Filik utilized multivariable Artificial Neural Networks (ANN) with local meteorological data, demonstrating significant improvement in prediction accuracy with the inclusion of additional meteorological parameters [6]. Additionally, multi-step prediction models have been designed using SSA, EMD, and CNNSVM, achieving better generalization and accuracy [7]. Short-term wind speed predictions have also been implemented using a combination of Local Mean Decomposition (LMD) and Least Squares Support Vector Machine (LSSVM), demonstrating high prediction accuracy [8]. In 2022, Arvanitidis explored the impact of different temperature reduction functions in optimization techniques, using SVM, MLP, and LSTM models with weather data from Skyros Island. The optimized MLP model outperformed others in terms of prediction accuracy [9].

However, these studies often did not consider multicollinearity among input variables, which can affect prediction reliability [10]. Moreover, using too few or unrepresentative parameters can lead to prediction errors, while too many features can increase data acquisition costs and computational complexity, resulting in the "curse of dimensionality" [11]. To address these issues, feature selection plays a crucial role. Gupta et al. demonstrated the effectiveness of feature selection using the Genetic Algorithm (GA), which reduced the number of input variables and enhanced prediction accuracy by eliminating irrelevant information, thereby reducing computation time and complexity [12].

This study aimed to analyze the selection of meteorological parameters that significantly influence wind speed prediction. By utilizing 15 parameters derived from

979-8-3315-1688-8/24/$31.00 ©2024 IEEE

weather monitoring data [13], the study employed the GA for optimal feature selection. The GA was chosen for its interactive nature, ability to handle high-dimensional data without information loss, and reduction in processing time and memory requirements [11][12]. The use of GA helped identify the most relevant features, reducing the dimensionality of the data, and addressing multicollinearity. The selected features were then used as input for wind speed prediction using the Multi-Layer Perceptron (MLP) method, which is well-suited for modeling non-linear data and addressing prediction challenges with its structure based on the human brain's neural system [14]. This research provided recommendations on the most significant parameters affecting wind speed prediction, ultimately improving prediction accuracy and efficiency.

II. Methodology

This section describes the research methodology employed in this study, including data collection and processing techniques, feature selection process, prediction method, system development approach, and testing methods.

A. Data Collection

The data used in this research were sourced from meteorological weather data provided by Russia on the website https://rp5.ru/. The data are updated in real-time every three hours, and a total of 8,264 data points were used, encompassing 15 variables as detailed in Table I.

TABLE I. WEATHER DATA

No	Variable	Description
1	T	Air temperature (°C) at 2 meters above the ground
2	Po	Atmospheric pressure at the weather station (mm Hg)
3	P	Atmospheric pressure reduced to mean sea level (mm Hg)
4	Pa	Pressure tendency over the last three hours (mm Hg)
5	U	Relative humidity (%) at 2 meters above the ground
6	DD	Average wind direction (compass points) at 10-12 meters
7	N	Total cloud cover
8	Cl	Low clouds: Stratocumulus, Stratus, Cumulus, Cumulonimbus
9	Nh	Amount of low or medium clouds
10	H	Height of the lowest cloud base (m)
11	Cm	Medium clouds: Altocumulus, Altostratus, Nimbostratus
12	Ch	High clouds: Cirrus, Cirrocumulus, Cirrostratus
13	VV	Horizontal visibility
14	Td	Dew point temperature (°C) at 2 meters above the ground
15	Ff	Average wind speed at 10-12 meters above the ground (m/s)

B. Data Preprocessing Techniques

The data processing in this research involved three main stages, which are preprocessing, feature selection using the GA, and wind speed prediction using MLP method. The steps of the data processing technique were as follows:

1. *Data Preprocessing:* Data preprocessing involved several steps. First, the dataset was filtered for the period from January 2021 to October 2023. Inconsistent parameters were removed, resulting in a consistent set of 15 parameters. The data were then identified as 14 input parameters and 1 output parameter. Missing values were handled using the KNNImputer method, and categorical data were encoded to numerical values. Following this, data normalization was performed to ensure that all features were on the same scale, which is crucial for the effective performance of the GA and MLP. Normalization was done using the Min-Max Scaling method as shown in (1), which scales the data to a range of 0 to 1. This method has been proven to yield better results in terms of the number of variants and data differences compared to other normalization methods [15][16].

$$X_{norm} = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (1)$$

In (1), X is the original value, X_{norm} is the normalized value, X_{min} and X_{max} are the minimum and maximum values of the feature, respectively.

2. *Parameter Settings:* The parameters for the GA, including population size, number of iterations, crossover rate, and mutation rate, were carefully selected based on prior studies and empirical testing [17]. Similarly, for the MLP, the number of hidden neurons and the learning rate were fine-tuned to optimize the model's performance,

3. *Feature Selection:* Feature selection was carried out before prediction. The GA was used to select features, with a binary representation for the genes in the chromosomes. Each gene value of 1 indicated that the parameter was selected, while a value of 0 indicated it was not used in the prediction. The initial population consisted of five chromosomes with 14 genes each. Fitness values were calculated using MLP to measure the quality of each chromosome. The fitness function was based on the F1 score of the wind speed predictions made by the MLP, with higher F1 scores indicating better fitness.

4. *Prediction:* The prediction process involved using the MLP method. The fitness calculation for each chromosome was performed by determining the input layer, hidden layer, and output layer neurons based on the selected features. The prediction results were then evaluated using a confusion matrix to compare the model's predicted values with the actual values.

C. Evaluation

Evaluation was conducted to assess the performance and accuracy of the wind speed prediction model. The primary tool for evaluation was the confusion matrix, which provides a comprehensive breakdown of the model's performance across different classes. A confusion matrix can be used in both binary and multi-class classification case studies [18]. In this study, a confusion matrix was used to measure performance using metrics such as accuracy, recall, precision, and F1-score.

III. Results and Discussion

The testing phase of this research focused on evaluating the accuracy of wind speed prediction models using MLP. The evaluation utilized confusion matrix metrics to measure the effectiveness of the models in predicting wind speed classes. The following sections detail the testing outcomes, including GA testing, MLP hyperparameter testing, feature selection testing, prediction model accuracy testing, and data distribution testing.

A. *Results of Genetic Algorithm Parameter Testing Results*

The results of GA parameter testing included the number of iterations testing, population size testing, and crossover and mutation rate testing.

a) Number of Iterations Testing: The number of iterations represented the number of times the feature selection process was repeated to reach an optimal solution. The more iterations, the higher the likelihood of increasing the fitness value. The average fitness value was calculated from the results of each iteration test conducted in three trials, each with 100 iterations [19]. In this test, a population of 5 was used with a combination of cr at 0.7 and mr at 0.3. The fitness calculation with MLP used 50 epochs with 6 neurons in the hidden layer and a learning rate of 0.1 [17]. The number of iterations testing results are shown in Fig. 1.

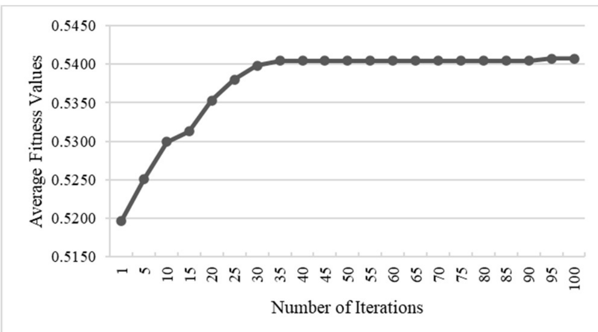

Fig. 1. Results of number of iterations

Based on the testing results in Fig.1, not all iterations resulted in significant changes in fitness value, as the GA is stochastic and variations between iterations can occur. However, the graph in Fig. 1 shows that the average fitness value tended to increase with each iteration. The graph indicates that the fitness value reaches its maximum, 0.5404222, at the 35th iteration. After the 35th iteration, the fitness value tended to stabilize and did not change, indicating that the 35th iteration achieved the optimal value as no further increase was observed beyond this point.

b) Population Size Testing: The population in GA is a collection of individuals or chromosomes representing the solutions for the features to be selected. Population testing was conducted in multiples of 5, ranging from 5 to 30 populations, and each variation was tested three times. The average fitness value from each trial was calculated to determine the optimal fitness value. The population size testing results are shown in the graph in Fig. 2.

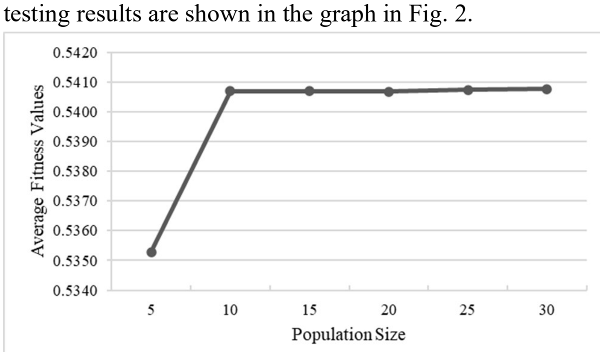

Fig. 2. Results of population size

Based on the population size testing results shown in the graph in Fig. 2, the average fitness value tended to increase from a population of 5 to 10. However, after testing a population of 10, the average fitness value did not show a significant increase. This indicates that a population size of 10, with a fitness value of 0.54069545, is optimal for this feature selection process.

c) Crossover and Mutation Rate Testing: The crossover rate (cr) is a parameter that determines the number of crossover processes that occur in each generation. Similarly, the mutation process has a parameter that determines the number of mutation processes that occur in each iteration, known as the mutation rate (mr). In this testing, the crossover rate and mutation rate were combined such that their sum equals 1, to maintain the balance between the number of offspring produced by crossover and mutation processes, ensuring it did not exceed the initial population size [20]. The testing was conducted three times to obtain the highest average fitness value. The results of the cr and mr testing are shown in the graph in Fig. 3.

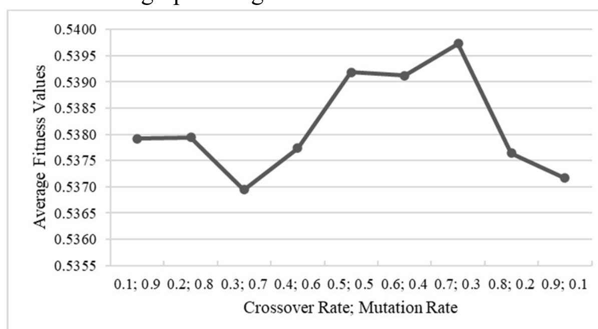

Fig. 3. Results of crossover and mutation testing

Based on the graph in Fig. 3, the cr and mr testing results show that the average fitness value varied. In the search for the highest average fitness value, the combination of cr at 0.7 and mr at 0.3 achieved the highest average fitness value of 0.5397317. Therefore, this combination is considered optimal as it achieves the best fitness value compared to others.

B. *MLP Hyperparameter Testing Results*

The results of MLP hyperparameter testing included three parts, which are epoch testing, hidden layer neuron count testing, and learning rate testing.

a) Epoch Testing: In neural networks, an epoch represents one complete pass through the entire training dataset used to train the model. Epoch testing was conducted in multiples of 10, ranging from 10 to 80. In this test, each epoch was evaluated three times. The best fitness value from the last iteration of the GA was taken to calculate the average. The epoch testing results are shown in Fig. 4.

Based on Fig. 4, the fitness value increased with each epoch. However, at the 50th epoch, the fitness value reached its highest, which was 0.5397317. After the 50th epoch, the fitness value no longer increased but instead decreased. The decrease in fitness value after the 50th epoch was caused by an increase in validation errors. Stopping training when validation error increases can result in a better model and prevent overfitting [21]. Therefore, the 50th epoch is considered optimal.

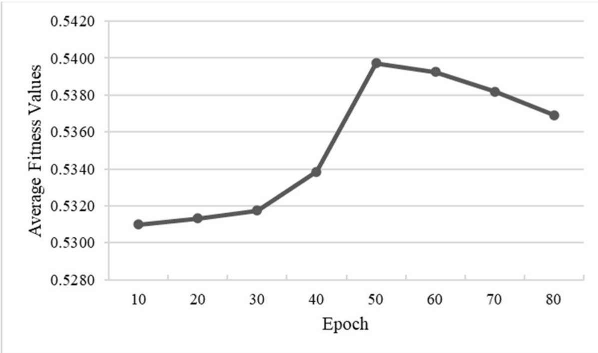

Fig. 4. Results of epoch testing

b) Number of Hidden Neurons Testing: Hidden neurons are processing units located in the hidden layer, which is between the input layer and the output layer. In testing the number of hidden neurons, three trials were conducted to obtain the best fitness value in the GA iteration to calculate the average. The number of hidden neurons testing results are shown in the graph in Fig. 5.

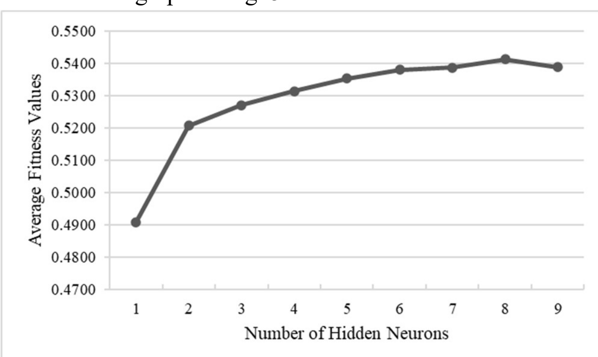

Fig. 5. Number of hidden neurons results

Based on the testing results in the graph in Fig. 5, it is shown that the more neurons in the hidden layer, the higher the fitness value. However, at eight neurons, the fitness value reached its highest, which was 0.54131148. After that, the fitness value did not increase but instead decreased. The decrease in fitness value after eight neurons was caused by an increase in validation error. Stopping training when validation error increases can result in a better model and prevent overfitting [21]. This indicates that with eight neurons in the hidden layer, the model achieved a very good fitness value.

c) Learning Rate Testing: The learning rate is a parameter that determines the size of the weight changes applied at each iteration during the model training process. Learning rate testing started from 0.01 to 0.9 with three trials to obtain the best fitness value in GA iteration to calculate the average. The learning rate testing results are shown in the graph in Fig. 6.

Based on Fig. 6, there were variations in fitness values. However, at a learning rate of 0.1, the fitness value reached its highest, which is 0.5446114. After that, the fitness value did not increase but instead decreased. The decrease in fitness value after the learning rate of 0.1 was caused by an increase in validation error. Stopping training when validation error increased resulted in a better model and prevented overfitting [21]. Therefore, a learning rate of 0.1 is considered optimal.

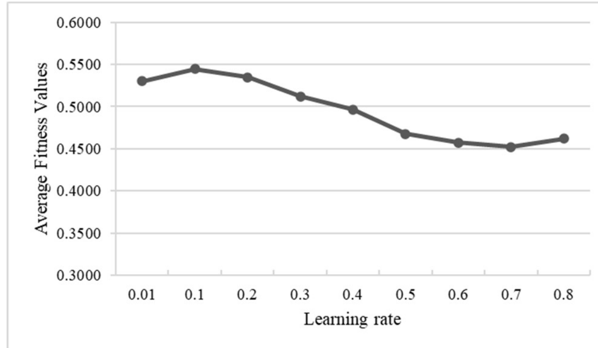

Fig. 6. Results of learning rate

C. Feature Selection Testing Results

Feature selection testing was conducted using the results of the optimal GA and MLP parameters. The feature testing was carried out three times to obtain the best features for wind speed prediction. The parameters used, based on the previous testing results, are shown in Table II.

TABLE II. OPTIMAL PARAMETERS

No	Parameters	Values
1	Number of iterations	35
2	Population size	10
3	Crossover rate (cr)	0.7
4	Mutation rate (mr)	0.3
5	Epoch	50
6	Number of Hidden Neurons	8
7	Learning rate	0.1

After conducting three trials, the testing results were obtained after the GA and MLP parameter testing achieved optimal parameters, as shown in Table III.

TABLE III. FEATURE SELECTION TESTING RESULTS

Trials	Selected Feature	Selected Variable	Number of Selected Feature	Fitness Values
1	[1 0 1 0 0 1 0 0 0 0 0 0 0 1]	['T', 'P', 'DD', 'Td']	4	0.541274
2	[1 0 1 0 0 1 0 0 0 0 0 0 0 1]	['T', 'P', 'DD', 'Td']	4	0.539096
3	[1 0 1 0 0 1 0 0 0 0 0 0 0 1]	['T', 'P', 'DD', 'Td']	4	0.536390

Based on Table III, it is evident that from the three trials conducted, there is consistency in the number of features and the position of the feature indices selected optimally. The selected features are 'T' (air temperature), 'P' (atmospheric pressure less than sea level), 'DD' (wind direction), and 'Td' (dew point temperature), which are located at indices 1, 3, 6, and 14, respectively. This consistency indicates that the resulting chromosomes are the optimal solution for predicting wind speed. Therefore, the features 'T', 'P', 'DD', and 'Td' can be considered the most significantly influential features in predicting wind speed.

D. Prediction Model Accuracy Testing Results

Accuracy testing was conducted to compare the prediction results using the four selected features and the entire set of 14 original dataset features, with each test conducted three times.

The results of this testing are shown in the graphs in Fig. 7, and computation time results are shown in Fig. 8.

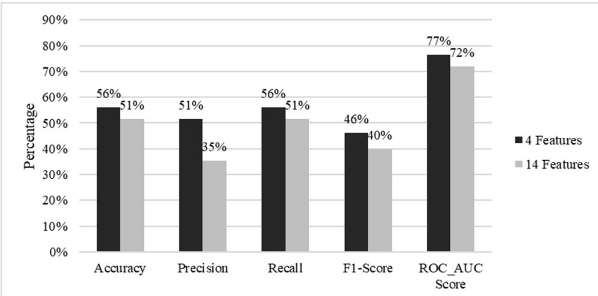

Fig. 7. Results of prediction model accuracy

In Fig. 7, there were differences in testing results between using the four selected features and the 14 original dataset features. The average accuracy with the four selected features was 56%, while it was 51% with the 14 features, indicating that optimized features improved accuracy. The average precision with the four selected features was 51%, compared to 35% with the 14 features, showing better and more accurate identification of positive classes. The average recall with the four selected features was 56%, while it was 51% with the 14 features, indicating a better ability to identify all actual positive samples. The average F1-score with the four selected features was 46%, compared to 40% with the 14 features, demonstrating a better balance between precision and recall. The ROC-AUC score with the four features was 77%, compared to 72% with the 14 features, indicating better differentiation and separation of positive and negative classes with the optimized features.

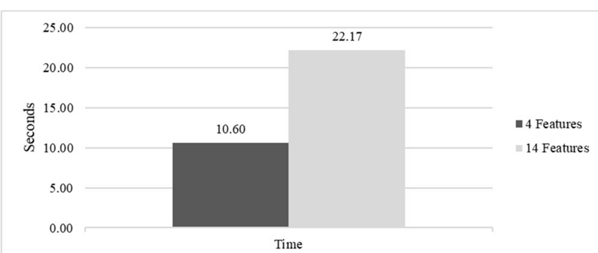

Fig. 8. Results of computation time

Based on the analysis of the graph in Fig. 8, there was a significant difference in computation time between using the four selected features and the 14 original features. The four selected features had an average computation time of 10.60 seconds, while the 14 original features required 22.17 seconds. Faster computation time can enhance overall computational resource efficiency and potentially reduce latency in generating predictions, especially with large datasets. Therefore, reducing feature dimensions can significantly improve model performance and optimize the time required for predictions.

However, the accuracy was still not optimal as it was far from 100%. This condition was caused by an imbalance in the number of samples in each class, meaning some classes had significantly more or fewer samples than others. In this case, accuracy alone did not provide a complete picture of the model's performance, as the model tended to correctly predict the majority class while potentially neglecting the minority class. Therefore, evaluation using sensitivity and specificity metrics was necessary to assess the model's performance in classifying each positive and negative class specifically [22]. The sensitivity and specificity results from testing the 4 features, each conducted three times, can be seen in Tables IV. These results indicated that each class had imbalanced sensitivity and specificity values, affecting the overall evaluation.

TABLE IV. SENSITIVITY AND SPECIFICITY TESTING OF 4 FEATURES

Trials	Class Target						
	0	1	2	3	4	5	6
Sensitivity							
1	0.97	0	0.89	0.34	0	0	0
2	0.97	0	0.95	0.17	0.10	0	0
3	0.97	0.02	0.91	0.25	0.10	0	0
Specificity							
1	0.95	0.80	0.16	0.70	0.94	0.99	0.99
2	0.95	0.80	0.12	0.76	0.94	0.99	0.99
3	0.95	0.80	0.14	0.74	0.94	0.99	0.99

E. Data Distribution Testing Results

The preprocessing steps, including data normalization and handling of missing values, played a crucial role in the model's performance. The model learned more effectively by decreasing the impact of noise and outliers by ensuring consistent and high-quality input data. The KNNImputer method kept the data robust and representative, preventing bias that could affect prediction outcomes.

Data distribution testing was conducted to evaluate the imbalance in the target data in this study. The aim of this testing was to determine whether data imbalance affected the model evaluation results, especially leading to low accuracy, even if the model was considered effective and suitable for good predictions. The data distribution test was performed by removing imbalanced target data, focusing on classes C1 and C3, which showed a more balanced tendency compared to other classes, as seen in Fig. 9. The evaluation results of the data distribution testing are shown in Table V.

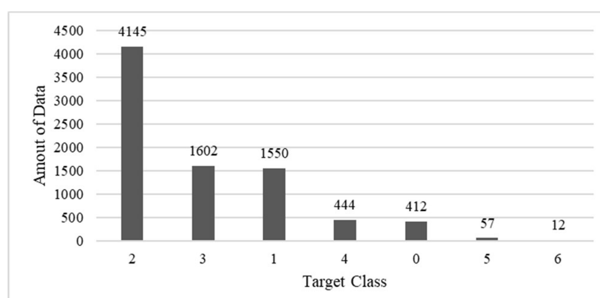

Fig. 9. Data distribution

TABLE V. DATA DISTRIBUTION TESTING RESULTS

Confusion Matrix	Trials			
	1	2	3	Average
Accuracy	0.85	0.84	0.85	0.85
Precision	0.85	0.85	0.85	0.85
Recall	0.85	0.84	0.85	0.85
F1-Score	0.85	0.84	0.85	0.85
Time (s)	11.02	10.36	6.07	9.15

Table V showed that the average accuracy reached 85%. From these results, it can be concluded that the imbalance in the target data in multi-class classification cases can affect the model's performance and result in a decrease in accuracy.

IV. Conclusion

The study successfully applied the GA for feature selection, leveraging concepts of evolution and reproduction within a chromosome population, and identified the optimal subset of features (air temperature, atmospheric pressure, wind direction, and dew point temperature) for wind speed prediction. The wind speed prediction process using MLP with these selected features involved initialization, one-hot encoding, sigmoid and SoftMax activation functions, and optimization with Adam. Evaluation results indicated that the model using the selected features achieved an accuracy of 56%, precision of 51%, recall of 56%, F1-score of 46%, and ROC-AUC score of 77%, with a computation time of 10.60 seconds. In comparison, the model using all 14 original features had lower accuracy and longer computation time. Further analysis through data distribution testing, which yielded an accuracy of 85%, demonstrated that the lower accuracy results were primarily due to class imbalance. This highlights the need for addressing data imbalance to significantly improve model performance and overall accuracy in multi-class classification.

Based on this study, several recommendations can be made for future research to improve the accuracy of multi-class wind speed prediction using MLP and address class imbalance. To mitigate class imbalance, algorithms such as SMOTE (Synthetic Minority Over-sampling Technique) and ADASYN (Adaptive Synthetic Sampling) can be implemented to create synthetic samples for minority classes, balancing the dataset and enhancing accuracy. Furthermore, employing extensive hyperparameter tuning methods such as Grid Search or Bayesian Optimization can optimize MLP performance by systematically searching through a broader range of hyperparameters, leading to improved model accuracy and efficiency. These strategies can build on the findings of this study to further enhance the performance and reliability of wind speed prediction models.

References

[1] D. Karamanis, "Management of moderate wind energy coastal resources," *Energy Convers Manag*, vol. 52, no. 7, pp. 2623–2628, 2011, doi: 10.1016/j.enconman.2011.01.002.

[2] A. Musyafa', I. Made, Y. Negara, and I. Robandi, "Wind-Electric Power Potential Assessment for Three Locations in East Java-Indonesia," 2011.

[3] S. E. Hosseini and M. A. Wahid, "Hydrogen production from renewable and sustainable energy resources: Promising green energy carrier for clean development," May 01, 2016, *Elsevier Ltd*. doi: 10.1016/j.rser.2015.12.112.

[4] C. Ren, N. An, J. Wang, L. Li, B. Hu, and D. Shang, "Optimal parameters selection for BP neural network based on particle swarm optimization: A case study of wind speed forecasting," *Knowl Based Syst*, vol. 56, pp. 226–239, Jan. 2014, doi: 10.1016/j.knosys.2013.11.015.

[5] T. Adedipe, M. Shafiee, and E. Zio, "Bayesian Network Modelling for the Wind Energy Industry: An Overview," Oct. 01, 2020, *Elsevier Ltd*. doi: 10.1016/j.ress.2020.107053.

[6] Ü. B. Filik and T. Filik, "Wind Speed Prediction Using Artificial Neural Networks Based on Multiple Local Measurements in Eskisehir," in *Energy Procedia*, Elsevier Ltd, Feb. 2017, pp. 264–269. doi: 10.1016/j.egypro.2016.12.147.

[7] X. Mi, H. Liu, and Y. Li, "Wind speed prediction model using singular spectrum analysis, empirical mode decomposition and convolutional support vector machine," *Energy Convers Manag*, vol. 180, pp. 196–205, Jan. 2019, doi: 10.1016/j.enconman.2018.11.006.

[8] Z. Tian, "Short-term wind speed prediction based on LMD and improved FA optimized combined kernel function LSSVM," *Eng Appl Artif Intell*, vol. 91, May 2020, doi: 10.1016/j.engappai.2020.103573.

[9] A. I. Arvanitidis, D. Kontogiannis, G. Vontzos, V. Laitsos, and D. Bargiotas, "Stochastic Heuristic Optimization of Machine Learning Estimators for Short-Term Wind Power Forecasting," in *2022 57th International Universities Power Engineering Conference: Big Data and Smart Grids, UPEC 2022 - Proceedings*, Institute of Electrical and Electronics Engineers Inc., 2022. doi: 10.1109/UPEC55022.2022.9917957.

[10] S. Barhmi, O. Elfatni, and I. Belhaj, "Forecasting of wind speed using multiple linear regression and artificial neural networks," *Energy Systems*, vol. 11, no. 4, pp. 935–946, Nov. 2020, doi: 10.1007/s12667-019-00338-y.

[11] B. K. Khotimah, M. Miswanto, and H. Suprajitno, "Optimization of feature selection using genetic algorithm in naïve Bayes classification for incomplete data," *International Journal of Intelligent Engineering and Systems*, vol. 13, no. 1, pp. 334–343, Feb. 2020, doi: 10.22266/ijies2020.0229.31.

[12] R. A. Gupta, R. Kumar, and A. Kumar Bansal, "Selection of Input Variables for the Prediction of Wind Speed in Wind Farms Based on Genetic Algorithm," 2018.

[13] M. R. Dehbozorgi, M. Rastegar, and M. Dabbaghjamanesh, "Decision tree-based classifiers for root-cause detection of equipment-related distribution power system outages," *IET Generation, Transmission and Distribution*, vol. 14, no. 24, pp. 5816–5822, Dec. 2020, doi: 10.1049/iet-gtd.2020.0570.

[14] S. Samadianfard *et al.*, "Wind speed prediction using a hybrid model of the multi-layer perceptron and whale optimization algorithm," *Energy Reports*, vol. 6, pp. 1147–1159, Nov. 2020, doi: 10.1016/j.egyr.2020.05.001.

[15] V. N. G. Raju, K. P. Lakshmi, V. M. Jain, A. Kalidindi, and V. Padma, "Study the Influence of Normalization/Transformation process on the Accuracy of Supervised Classification," in *Proceedings of the 3rd International Conference on Smart Systems and Inventive Technology, ICSSIT 2020*, Institute of Electrical and Electronics Engineers Inc., Aug. 2020, pp. 729–735. doi: 10.1109/ICSSIT48917.2020.9214160.

[16] G. Aksu, C. O. Güzeller, and M. T. Eser, "The Effect of the Normalization Method Used in Different Sample Sizes on the Success of Artificial Neural Network Model," *International Journal of Assessment Tools in Education*, vol. 6, no. 2, pp. 170–192, Jul. 2019, doi: 10.21449/ijate.479404.

[17] A. Adorada and A. Wibowo, "Genetic Algorithm-Based Feature Selection and Optimization of Backpropagation Neural Network Parameters for Classification of Breast Cancer Using MicroRNA Profiles," *International Conference on Informatics and Computational Sciences*, 2019.

[18] I. Markoulidakis, I. Rallis, I. Georgoulas, G. Kopsiaftis, A. Doulamis, and N. Doulamis, "Multiclass Confusion Matrix Reduction Method and Its Application on Net Promoter Score Classification Problem," *Technologies (Basel)*, vol. 9, no. 4, Dec. 2021, doi: 10.3390/technologies9040081.

[19] M. S. Mahmod, Z. A. H. Alnaish, and I. A. Al-Hadi, "Hybrid Intrusion Detection System Using Artificial Bee Colony Algorithm and Multi-Layer Perceptron," *International Journal of Computer Science and Information Security*, vol. 13, no. 2, 2015

[20] V. N. Wijayaningrum and W. F. Mahmudy, "Optimization of Ship's Route Scheduling Using Genetic Algorithm," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 2, no. 1, pp. 180–186, 2016, doi: 10.11591/ijeecs.v2.i1.pages.

[21] A. Khamis and S. N. S. B. Abdullah, "Forecasting Wheat Price Using Backpropagation And NARX Neural Network," *The International Journal Of Engineering And Science (IJES)*, pp. 19–26, 2014

[22] M. Buda, A. Maki, and M. A. Mazurowski, "A systematic study of the class imbalance problem in convolutional neural networks," *Neural Networks*, vol. 106, pp. 249–259, Oct. 2018, doi: 10.1016/j.neunet.2018.07.011.