IMPLEMENTATION OF MLP AND MEDIAPIPE ON A MOBILE APPLICATION TO ENHANCE EXERCISE MOVEMENT DETECTION ACCURACY

**Muhammad Mizzy1, Prayitno1, Kuwat Santoso1**

1 Department of Electrical Engineering, Semarang State Polytechnic, Semarang, Indonesia

|  |  |  |
| --- | --- | --- |
| **Article Info** |  | **ABSTRACT** |
| ***Article history:***  Received month dd, yyyy  Revised month dd, yyyy  Accepted month dd, yyyy |  | This study aims to develop an intelligent Android application capable of detecting and counting exercise movements automatically using Deep Learning techniques, specifically leveraging MediaPipe for pose estimation and a Multi-Layer Perceptron (MLP) model for classification. The system is designed to support home workouts by providing real-time feedback and tracking of exercise repetitions. Data collection involved five types of exercises: push-ups, squats, jumping jacks, planks, and ab crunches. The dataset comprised 3,187 samples, split into 80% for training and 20% for testing. The MLP model was compared with other Deep Learning models, including 1D CNN, GRU, and LSTM, and achieved a maximum accuracy of 99%, outperforming other models. The application features a user-friendly interface, enabling users to start workout sessions, monitor repetitions, and access their training history. The results demonstrate the system's effectiveness in facilitating independent and accurate exercise tracking at home. |
| ***Keywords:***  Deep Learning  Mediapipe  MLP  Exercise Detection  Mobile Application  Human Pose Estimation |
| *This is an open access article under the* [*CC BY-SA*](https://creativecommons.org/licenses/by-sa/4.0/) *license.* |
| ***Corresponding Author:***  Prayitno Department of Electrical Engineering, Semarang State Polytechnic, Indonesia Jl. Prof. Soedarto, Tembalang, Semarang, Jawa Tengah 50275 Email: prayitno@polines.ac.id | | |

1. **INTRODUCTION**

Currently, public awareness of the importance of a healthy and fit lifestyle remains low. To achieve a healthy and fit life, individuals and communities must enhance their physical fitness by engaging in low, moderate, and high-intensity physical activities. Despite advancements in modern society, interest in sports and physical exercise remains limited. According to a WHO study, over 2 million deaths annually are attributed to insufficient physical activity. Globally, the majority of deaths are caused by physical inactivity, accounting for 60% to 85% of cases where individuals fail to maintain their physical health, alongside other contributing factors such as smoking and unhealthy eating habits [1].

One of the most crucial aspects of human life that must be maintained is health. To preserve it, individuals need to engage in activities that promote physical well-being. Physical fitness refers to the body's ability to adapt to physical demands, enabling individuals to carry out daily activities without excessive fatigue or physical strain. It serves as an indicator of overall physical condition, particularly cardiovascular health. Physical fitness is a key component of physical conditioning, which is fundamental in sports performance development. Thus, a thorough understanding of fitness conditioning is essential [2].

In this context, technologies such as Human Pose Estimation (HPE) become highly relevant. A pose can be defined as the configuration of human joints in a specific arrangement. Therefore, HPE refers to the process of identifying the locations of joints or specific landmarks on the human body. In image and video analysis, various types of pose estimation exist, including body, facial, and hand pose estimation, particularly in the field of computer vision [3].

Furthermore, the development of Android-based applications for exercise movement detection and counting has been the focus of multiple studies. For instance, one study developed a yoga pose detection application using a Convolutional Neural Network (CNN) model to classify various yoga poses with high accuracy. Yoga movement images were divided into 80% for training and 20% for testing, resulting in an effective model for real-time yoga pose recognition [4].

The implementation of deep learning in such applications not only improves movement detection accuracy but also enables the development of adaptive and responsive systems for various types of physical activities. Consequently, users can maximize the benefits of their workouts through precise monitoring and personalized training programs.

Overall, the integration of deep learning methods in developing Android-based exercise movement detection and counting applications offers an innovative solution to support a healthy and active lifestyle through intelligent technology.

.

1. **METHOD**

The dataset used in this study is derived from sports movement videos, combining videos sourced from YouTube with videos that were independently recorded.

* 1. **Data Collection and Preparation**

A dataset consisting of 3,187 samples was collected, encompassing ten exercise movement classes: plank, cobra, push-up (up and down), ab crunch (up and down), squat (up and down), and jumping jack (start and end). Landmark data were extracted using MediaPipe for each frame.



Figure 1 Dataset acquisition scheme

This system utilizes OpenCV to access the camera and capture each video frame in real-time. Each captured frame is then processed using the Mediapipe model to detect the pose landmarks on the user’s body, such as joint and limb positions. From these detected poses, the system performs feature extraction by collecting the coordinates of the body pose points, which represent the position and orientation of the body in each frame. These coordinate data are then gathered and structured into a dataset that will be used in the classification model training process.

* 1. **Data Splitting**

The dataset was split into 80% training and 20% testing sets, with an additional 20% of the training set used as validation data during model training:

Table 1 Dataset collected for training the classification model.

|  |  |  |
| --- | --- | --- |
| **Training** | **Validation** | **Testing** |
| 2550 | 510 | 637 |

The dataset presented here is used to train and evaluate the classification model, which is designed to classify sports movements. The dataset is divided into 80% training data and 20% testing data, based on a total of 3,187 landmark data points that have been collected.

* 1. **Model Development**

Output

(bath\_size, prob\_map)

Linear(keypoints,

hidden\_din)

ReLU()

Dropout()

Linear(hidden\_din,

output\_class)

Softmax()

MLP

Input

(bath\_size, keypoints)

The MLP model architecture included; Input layer: 128 neurons, ReLU activation; Dropout layer (0.2 rate) for regularization; Two hidden layers (64 and 32 neurons, ReLU activation); Output layer: Softmax activation for multi-class classification

The model was compiled with the Adam optimizer, a sparse categorical cross-entropy loss function, and a learning rate of 0.001. Early stopping was employed to prevent overfitting.

* 1. **Comparative Models**

Other deep learning models (1D CNN, GRU, and LSTM) were also implemented and compared to the MLP in terms of accuracy and stability.

Model performance evaluation was conducted using the metrics of Precision, Recall, and F1-Score for each class. A brief explanation of each metric is as follows:

1. Precision is the ratio between the number of correct predictions for a particular class and the total number of predictions made for that class. A high Precision indicates that few false positive predictions were made.
2. Recall is the ratio between the number of correct predictions for a particular class and the total number of actual instances of that class. A high Recall indicates that the model has missed only a small number of actual instances.
3. The F1-Score is the harmonic mean of Precision and Recall, providing a balance between the two metrics.

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Class | MLP | | | LSTM | | | GRU | | | 1D-CNN | | |
| P | R | F1 | P | R | F1 | P | R | F1 | P | R | F1 |
| pushup\_up | 0.99 | 1.00 | 0.99 | 0.96 | 1.00 | 0.98 | 0.99 | 0.99 | 0.99 | 0.99 | 1.00 | 0.99 |
| pushup\_down | 1.00 | 0.99 | 0.99 | 0.98 | 0.93 | 0.96 | 0.97 | 0.99 | 0.98 | 1.00 | 0.99 | 0.99 |
| plank | 1.00 | 1.00 | 1.00 | 0.98 | 0.99 | 0.99 | 1.00 | 0.99 | 1.00 | 1.00 | 1.00 | 1.00 |
| squat\_up | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |
| squat\_down | 1.00 | 1.00 | 1.00 | 0.98 | 1.00 | 0.99 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |
| jumping\_jack\_start | 0.82 | 1.00 | 0.90 | 0.86 | 0.94 | 0.90 | 0.59 | 1.00 | 0.74 | 0.93 | 0.78 | 0.85 |
| jumping\_jack\_end | 1.00 | 0.78 | 0.88 | 0.93 | 0.81 | 0.87 | 1.00 | 0.31 | 0.48 | 0.81 | 0.94 | 0.87 |
| ab\_crunch\_down | 1.00 | 1.00 | 1.00 | 1.00 | 0.95 | 0.97 | 0.95 | 1.00 | 0.97 | 1.00 | 1.00 | 1.00 |
| ab\_crunch\_up | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 0.96 | 0.98 | 1.00 | 1.00 | 1.00 |
| cobra | 1.00 | 1.00 | 1.00 | 0.99 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |

1. **RESULTS AND DISCUSSION**

Table 2 Total Accuracy Model

|  |  |
| --- | --- |
| **Model** | **Total Accuracy** |
| MLP | 0.99 (99%) |
| 1D CNN | 0.98 (98%) |
| GRU | 0.96 (96%) |
| LSTM | 0.97 (97%) |

The MLP model achieved an overall testing accuracy of 99%, outperforming the other models.

This demonstrates the effectiveness of MLP in learning spatial relationships in the landmark data for exercise movement detection.

1. MLP

The MLP model demonstrated excellent performance across most classes. Nearly all movements achieved Precision, Recall, and F1-Score values close to 1.00, indicating highly consistent and accurate predictions.

However, a slight decrease in performance was observed in the jumping\_jack\_start class (F1-Score 0.90) and the jumping\_jack\_end class (F1-Score 0.88). This suggests that although the MLP model is capable of predicting the majority of classes very well, it still faces challenges in distinguishing the transitional phases of the jumping jack movement.

Table 3 Hyperparameter Classification Model MLP

|  |  |
| --- | --- |
| ***Hyperparameter* Classification Model MLP** | |
| Epochs | 10 |
| Batch Size | 32 |
| Learning Rate | 0.0001 |
| Optimizer | Adam |

1. LSTM

The LSTM model also demonstrated generally good performance overall. Precision, Recall, and F1-Score values for most classes exceeded 0.95.

Similar to the MLP model, the LSTM model exhibited a performance drop in the jumping\_jack\_start class (F1-Score 0.90) and the jumping\_jack\_end class (F1-Score 0.87). This decrease in performance may be due to the similar movement characteristics during the transitional phases of the jumping jack, which are difficult to distinguish with only a limited temporal sequence.

Table 4 Hyperparameter Classification Model LSTM

|  |  |
| --- | --- |
| ***Hyperparameter* Classification Model LSTM** | |
| Epochs | 17 |
| Batch Size | 32 |
| Learning Rate | 0.0001 |
| Optimizer | Adam |

1. GRU

The GRU model exhibited a trend similar to that of the LSTM and MLP models. Overall, its performance was very good, with some classes achieving perfect F1-Score values (1.00), such as plank, squat\_up, squat\_down, and cobra.

However, the GRU model faced the greatest difficulty with the jumping\_jack\_end class (F1-Score only 0.48), indicating that although the GRU is capable of capturing temporal patterns, it struggles to distinguish the final transition of the jumping jack, which may involve more significant variations in posture.

Table 5 Hyperparameter Classification Model GRU

|  |  |
| --- | --- |
| ***Hyperparameter* Classification Model GRU** | |
| Epochs | 27 |
| Batch Size | 32 |
| Learning Rate | 0.0001 |
| Optimizer | Adam |

1. 1D-CNN

Model 1D-CNN secara umum memiliki performa yang sangat stabil dan baik pada mayoritas kelas, dengan nilai F1-Score sempurna (1.00) pada gerakan plank, squat\_up, squat\_down, ab\_crunch\_down, ab\_crunch\_up, dan cobra.

Kelas jumping\_jack\_start (F1-Score 0.85) dan jumping\_jack\_end (F1-Score 0.87) menunjukkan adanya potensi false positive atau false negative pada model ini. Meskipun demikian, dibandingkan dengan model GRU, model 1D-CNN memiliki performa yang lebih stabil untuk gerakan jumping jack.

Table 6 Hyperparameter Classification Model 1D-CNN

|  |  |
| --- | --- |
| ***Hyperparameter* Classification Model 1D-CNN** | |
| Jumlah Epochs | 15 |
| Batch Size | 32 |
| Learning Rate | 0.0001 |
| Optimizer | Adam |

* 1. **Implementation on Android**

The trained MLP model and MediaPipe pipeline were integrated into an Android application. The app provides real-time feedback to users during workouts, including repetition counting and exercise recognition.

**3.2. Implementation of Prediction Features into Website**

In this research, frameworks such as Laravel are utilized as one of the frameworks that allow for rapid application development [11]. Flask, which uses Python, also provides a solid foundation with essential features, while additional functionalities are delegated to extensions [12]. For the implementation, Laravel framework is used to create the pages on the dashboard, while Flask framework is employed to build APIs for the prediction results. Through APIs, communication or data exchange between different systems can be facilitated using virtual interface technology. The use of APIs simplifies the process of integrating new application components into existing systems, thereby aiding in team collaboration [13], [14], [15]. Additionally, databases are used for data storage. To further understand, a system overview diagram is provided below, encompassing the entire feature development of this application.

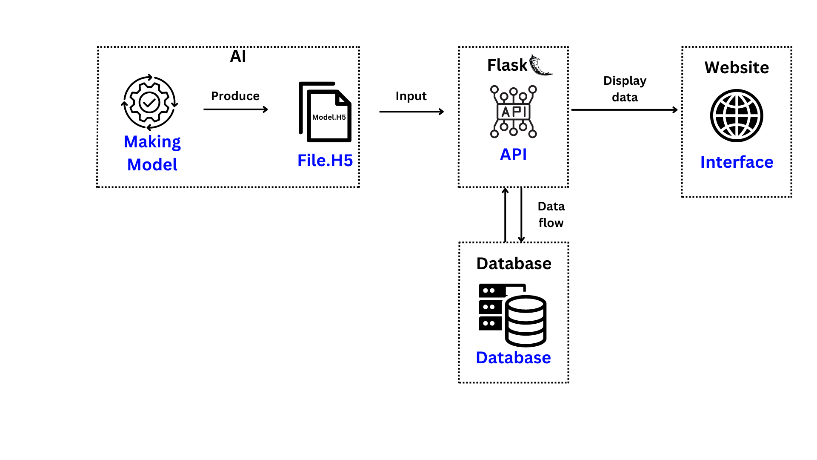


Figure 5. Prediction feature’s Overview

After obtaining the best model and integrating it into the Flask framework using the H5 file, the next step is to create an API with Flask framework that will provide responses as shown in Figure 6.

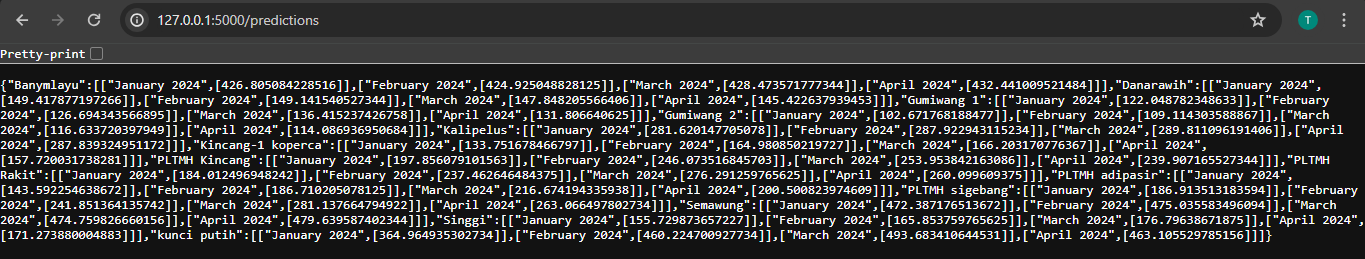


Figure 6. API Framework Flask Response

This feature is located on the dashboard page, where it will display the results of predictions previously made using Flask, and retrieved by the website program using the "GET" method.

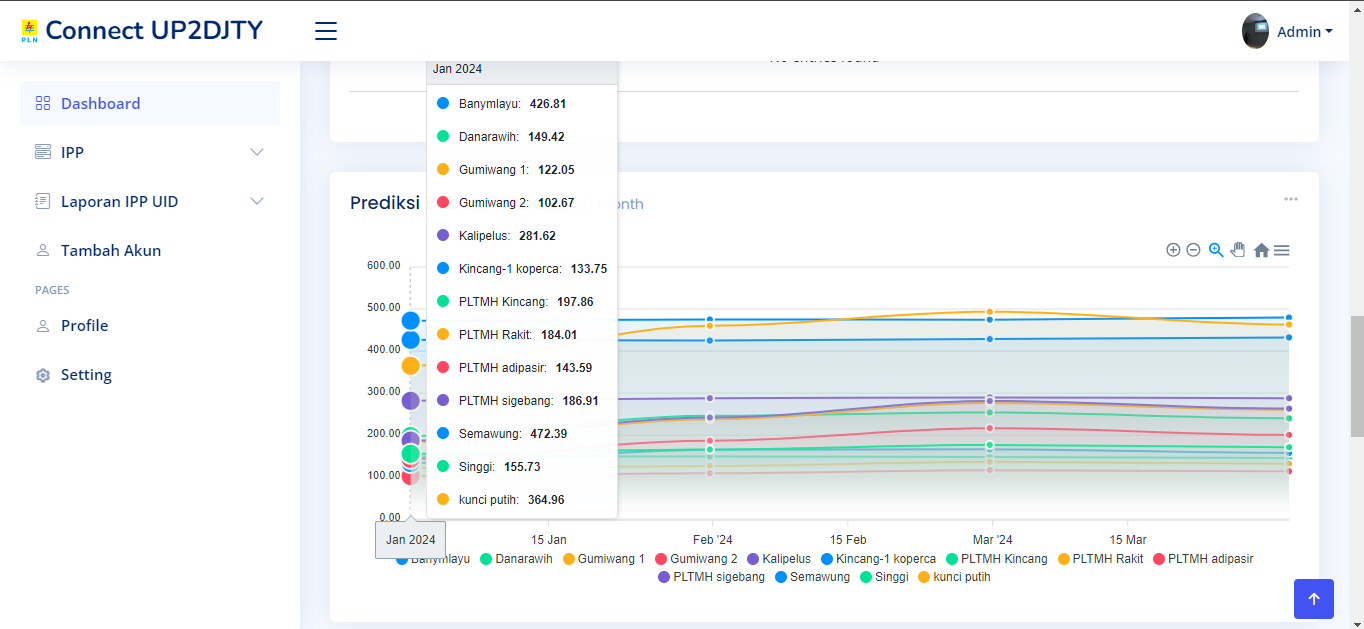
******

Figure 7. January Prediction Graphic Display

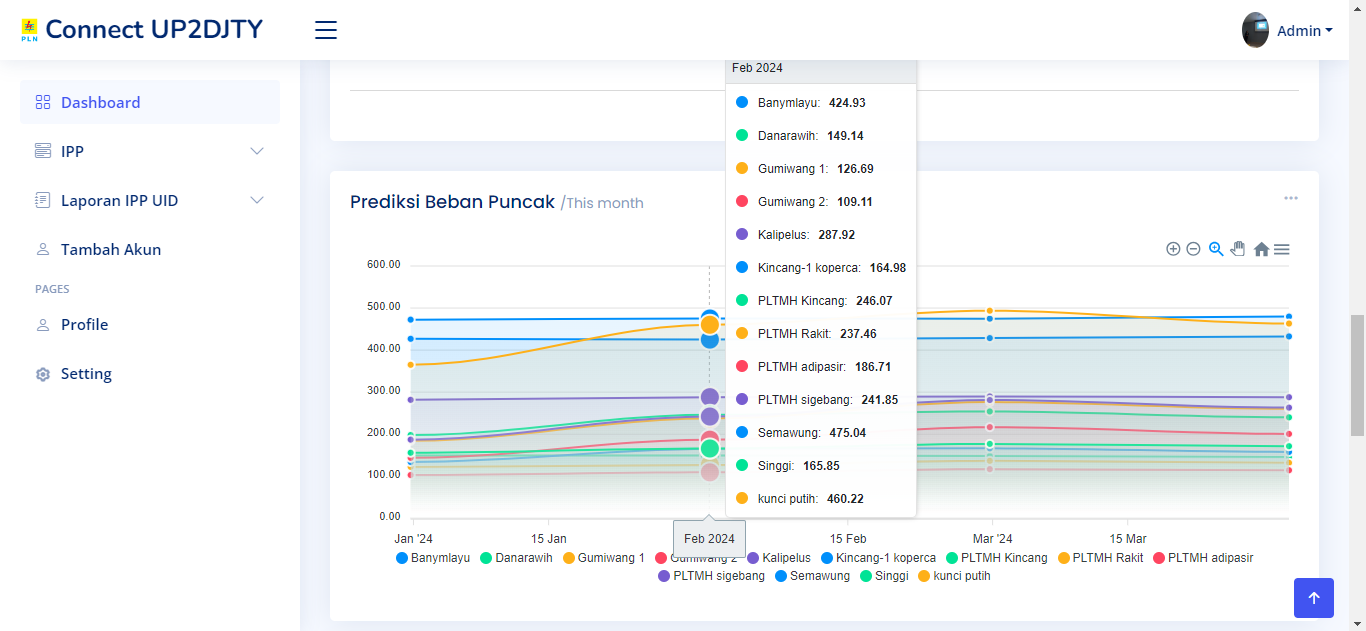
******

Figure 8. February Prediction Graphic Display

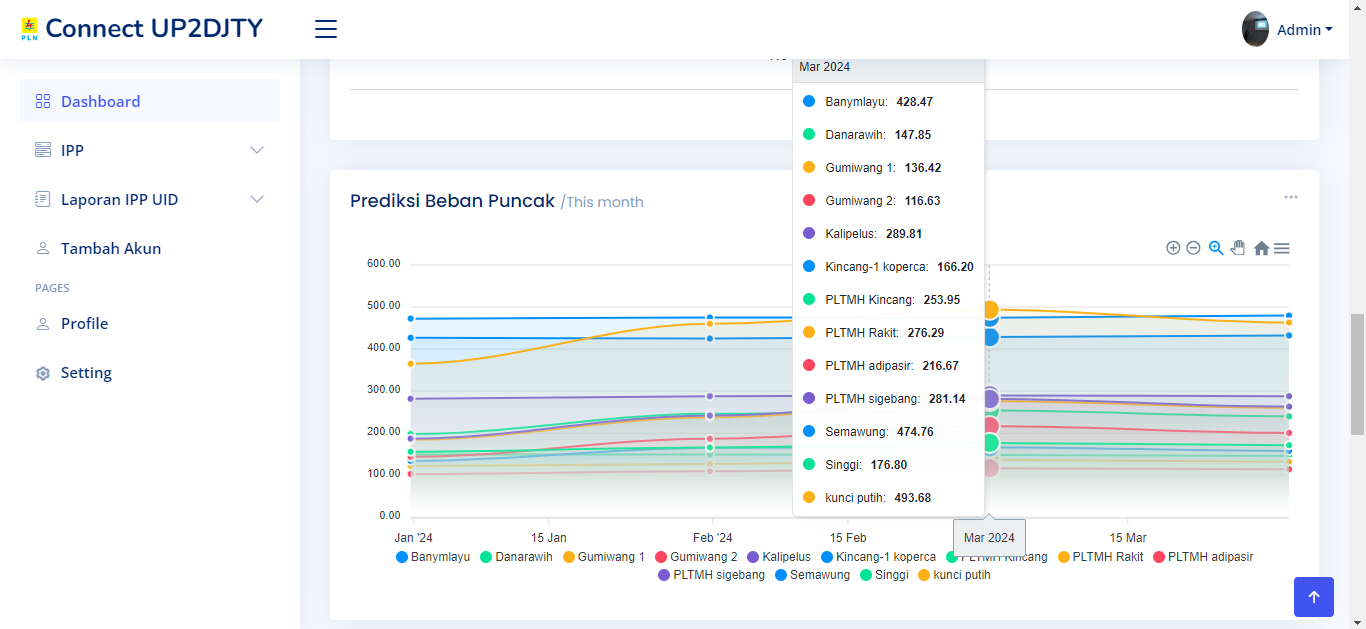
******

Figure 9. March Prediction Graphic Display

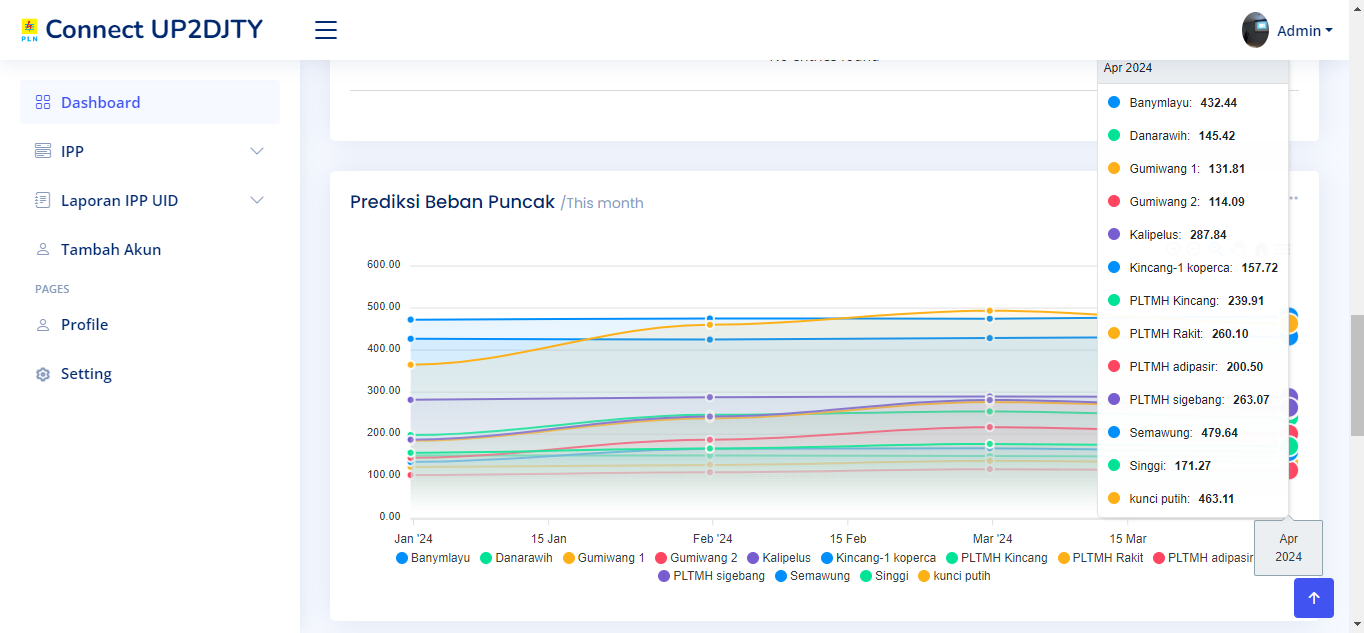
******

Figure 10. April Prediction Graphic Display

1. **CONCLUSION**

This feature has been able to predict peak generator loads using available data and utilizing both frameworks to create the feature interface. The feature employs the GRU method with 64 units, a batch size of 70, and a validation split of 0.4. The GRU error results show an RMSE of 24.87%, MSE of 6.18%, and MAE of 15.05%. Based on these error results, GRU has been compared with other models, and in the third scenario, it has proven to be the most effective model. These predictions need improvement by utilizing more data from the company to enhance pattern accuracy. Additionally, using different layer arrangements can further enhance the accuracy of the prediction results.

**REFERENCES**

[1] W. Majid, “PERILAKU AKTIVITAS OLAHRAGA TERHADAP PENINGKATAN KEBUGARAN JASMANI PADA MASYARAKAT”.

[2] I. Mahfud, A. Gumantan, and R. A. Nugroho, “PELATIHAN PEMBINAAN KEBUGARAN JASMANI PESERTA EKSTRAKURIKULER OLAHRAGA,” *Wahana Dedik. J. PkM Ilmu Kependidikan*, vol. 3, no. 1, p. 56, May 2020, doi: 10.31851/dedikasi.v3i1.5374.

[3] R. Josyula and S. Ostadabbas, “A Review on Human Pose Estimation,” Oct. 13, 2021, *arXiv*: arXiv:2110.06877. doi: 10.48550/arXiv.2110.06877.

[4] M. F. Nur Syahbani and N. G. Ramadhan, “Klasifikasi Gerakan Yoga dengan Model Convolutional Neural Network Menggunakan Framework Streamlit,” *J. MEDIA Inform. BUDIDARMA*, vol. 7, no. 1, p. 509, Jan. 2023, doi: 10.30865/mib.v7i1.5520.

**BIOGRAPHIES OF AUTHORS**

|  |  |
| --- | --- |
|  | **Elza Safira Permatasari** is a final semester student of D4 Computer Science study program of Politeknik Negeri Semarang. During her lectures, she has gained a deep understanding of programming, networking, and in her final semester, she focused on web and artificial intelligence. Based on her interest in this field, she decided to take the final project on the topic of deep learning, specifically design and development of a web-based peak load prediction application for electricity usage at power plants using deep learning models: a case study of the "connect UP2DJTY" website. She can be contacted via email: elza.safira@polines.ac.id |
|  |  |
|  | **Prayitno, S.ST., M.T., Ph.D.**     received the B.Sc. degree in computer science from the Politeknik Elektronika Negeri Surabaya, the M.Sc. degree in electrical engineering from the Institute of Technology Bandung, and the Ph.D. degree in computer science and information engineering from the Asia University, Taiwan. He is currently an assistant professor with the department of electrical engineering Politeknik Negeri Semarang (Polines). His research interests include deep learning, data mining, and artificial intelligence for education. He can be contacted at email: prayitno@polines.ac.id |
|  |  |
| https://web.polines.ac.id/wp-content/uploads/2022/03/wiktasari-EL-768x1024.jpg | **Wiktasari, S.T., M.Kom.** received the B.E. degree Universitas Islam Sultan Agung, the M.Sc. degree in Diponegoro University. she is currently an expert assistant with the department of electrical engineering Politeknik Negeri Semarang (Polines). Her interest is web development. she can be contacted at email: wiktasari@polines.ac.id |