IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received May 29, 2021, accepted June 7, 2021, date of publication June 11, 2021, date of current version June 23, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3088152

Aggregating Time Series and Tabular Data in Deep Learning Model for University Students' GPA Prediction

HARJANTO PRABOWO[1], ALAM AHMAD HIDAYAT[2], TJENG WAWAN CENGGORO[2,3], REZA RAHUTOMO[2], KARTIKA PURWANDARI[2], AND BENS PARDAMEAN[2,4]

[1]Management Department, BINUS Business School Undergraduate Program, Bina Nusantara University, Jakarta 11480, Indonesia
[2]Bioinformatics and Data Science Research Center, Bina Nusantara University, Jakarta 11480, Indonesia
[3]Computer Science Department, School of Computer Science, Bina Nusantara University, Jakarta 11480, Indonesia
[4]Computer Science Department, BINUS Graduate Program—Master of Computer Science Program, Bina Nusantara University, Jakarta 11480, Indonesia

Corresponding author: Bens Pardamean (bdsrc@binus.edu)

ABSTRACT Current approaches of university students' Grade Point Average (GPA) prediction rely on the use of tabular data as input. Intuitively, adding historical GPA data can help to improve the performance of a GPA prediction model. In this study, we present a dual-input deep learning model that is able to simultaneously process time-series and tabular data for predicting student GPA. Our proposed model achieved the best performance among all tested models with 0.4142 MSE (Mean Squared Error) and 0.418 MAE (Mean Absolute Error) for GPA with a 4.0 scale. It also has the best R^2-score of 0.4879, which means it explains the true distribution of students' GPA better than other models.

INDEX TERMS Educational data mining, deep learning, GPA prediction, time-series data, tabular data.

I. INTRODUCTION

ANecessity of providing well-targeted academic consultation services in educational sectors becomes one of the major concerns in improving the quality of school and academic institutions. One of the most important features of such services is the use of educational data mining of student's academic performance, which is capable to reveal latent information that can improve the existing educational system within the institution. For instance, a predictive model can be employed by a university to forecast students' future academic performance, such that the university can identify the students that may have a poor grade. Thus, the university can foster them to have better academic performance, which leads to the improvement of the overall students' performance. Moreover, the accurate prediction of students' academic performance is also an effective strategy for student recruitment, admission, retention, and individualized educational support throughout a students' studies [1]. To measure the academic performance, the Grade Point Average (GPA) is commonly used [2]. The result of academic productivity via GPA values can provide a more straightforward approach to measure the students' satisfaction that includes environmental, academic, social, cultural, economic, and health aspects.

Numerous studies have been conducted to develop a prediction model for student GPA [1], [3]–[9]. In many studies, the input to the prediction model is tabular data [1], [3]–[7]. A summary of the tabular data used by these studies is provided in Table 1. Alternatively, historical GPA data can also be employed. Structurally, historical GPA data can be categorized as time-series data, which has different nature than tabular data. An example of study that employed historical GPA is the study by Patil *et al.* [8]. Furthermore, the combination of both tabular and historical GPA data has been proven to be beneficial by Iqbal *et al.* [9]. However, Iqbal *et al.* treat the historical GPA data as tabular data instead of time-series data, which is its most natural form. We argue that such a treatment remove useful information from the historical GPA data. Thus, in this study, we propose a model that can combine tabular data and historical GPA as time-series data for GPA prediction. The proposed model is a dual-input deep learning model that can take both tabular and time-series data as input simultaneously. The model consists of a Multi-Layer Perceptron (MLP) branch and a Long Short-Term Memory (LSTM) branch which are concatenated for a single GPA prediction.

The associate editor coordinating the review of this manuscript and approving it for publication was Long Xu.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

TABLE 1. Previous studies of GPA predictive model development with tabular data.

Author(s)	Tabular Features
Zollanvari et al. [1]	The answer of 19 questions from a questionnaire. Example question: "Do you allow time for exercise and socializing with friends?"
Bosch [3]	Pre-experiment GPA, Self-reported typical grades, Free or reduced-price lunch eligibility, Expectations of success in high school math, Gender, Highest level of parental education, Expectations of success in high school math, First year freshman, Race/ethnicity, and Fixed mindset.
Pojon [4]	Gender, Nationality, Place of Birth, Educational Stage, Grade, Classroom ID, Course Topic, Semester, Parental Relationship, the Parent Answering the School Survey, the Parent Level of Satisfaction, Number of times the student: Raised Hands, Visited the Course Content, Checked New Announcement, Joined the Discussion Group, Absent.
Putpuek et al. [5]	Gender, Previous Education, Province, Talent, Loan Status, Admission Type.
Musso et al. [6]	The result of the following tests/questionnaire: Attention Network Test, Automated Ospan, Learning Strategies Questionnaire, Adolescent Coping Scale, Perceived Social Support Scale, SMU Health Questionnaire, Remoralization Scale, and Socio-Demographic Questionnaire.
Davidson [7]	The result of Fragile Families and Child Wellbeing Survey.

II. RELATED WORKS

Most of the studies that developed GPA prediction models can be grouped in the domain of Educational Data Mining (EDM). It is defined as the utilization of various data mining techniques to analyze data and provide solution for educational problems [10]. Essentially, the utilization of EDM is motivated by study cases such as identifying at-risk students, prioritize learning needs, identify graduation rates, performance assessment, maximizing resources, optimizing curriculum renewal, and also GPA prediction. To solve these cases, a variety of Data Mining methods can be applied. Specifically for GPA prediction, it is common to employ popular supervised learning algorithms for Data Mining, such as decision-tree-based algorithms [11]–[16], Naive Bayes [5], [17], logistic regression [18], and rule-based classification [19]. Currently, there is no consensus which supervised algorithm is the best for educational data because the performance was varied from study to study. Despite the popularity of the supervised learning algorithms for Data Mining, it should be noticed that they naturally can only model tabular data. As the consequence, the aforementioned studies used only tabular data.

In recent years, however, there is a tendency in almost all research in data modeling to use deep learning, an umbrella term for recent advances in neural networks. Since the 2010s, deep learning has been adopted in numerous cases with a stunning performance. This trend is starting to permeate the research in GPA prediction as well. For example, Arsad *et al.* [20] used Multi-Layer Perceptron (MLP) for predicting the GPA of engineering students in the Faculty of Electrical Engineering, University of Technology MARA (UiTM), Malaysia. The input to the MLP model was the students' score in several subjects in the first semester. The target output was the students' CGPA at semester eight. Arsad *et al.* showed that fundamental courses in the first and third semesters have a strong influence on the final GPA upon graduation. Similarly, Sikder *et al.* [21] employed MLP for GPA prediction with data from the Department of Computer Science and Engineering at Bangabandhu Sheikh Mujibur Rahman Science and Technology University (BSMRSTU). Notice that, like other aforementioned supervised learning algorithms, MLP also suitable only for tabular data.

In addition to the powerful performance, adopting deep learning also provides a possibility to use time-series data. In the context of GPA prediction, it is possible to utilize historical GPA data by the use of deep learning. An example of this is the study by Patil *et al.* [8] that utilized Bidirectional Long-Short Term Memory (BLSTM), a variant of Long Short-Term Memory (LSTM) [22], which is a deep learning architecture specialized for modeling time-series data. By using LSTM, they can use historical GPA data instead of the common tabular data used by common GPA prediction models.

Meanwhile, Iqbal *et al.* used Restricted Boltzmann Machine (RBM), which is also a variant of deep learning, to predict GPA with both tabular and time-series data [9]. The data were acquired from the students in the Information Technology University (ITU), Lahore, Pakistan. Because RBM is not a time-series model, they straightforwardly translated the time-series data into tabular data by assuming that each element in the time-series as a separate column. Although this approach is not impossible to be implemented, it is theoretically less plausible than using LSTM for the time-series data.

III. METHODS

A. DATASET

In this study, we used an undergraduate database obtained from the Student Advisory and Support Center at Bina Nusantara University that comprises a total of 46,670 students enrolled from 2010 to 2017. The data consisted of two different types of data, which are tabular data and time-series data. The tabular data contains basic information of students including two categorical features: the campus locations (3 locations) and academic programs (50 programs); and numerical features: enrollment year (2011 - 2017), TOEFL score, student orientation score, and leave counts.

Meanwhile, the time-series data comprised 46,670 sequences of student academic grades—i.e., Grade Point Averages (GPA), reported biannually (semester systems) throughout their active study period. The GPA was computed on the standard 4.00 scale, which is the most common grading system in Indonesia. The longest study period was observed up to 16 semesters and the shortest was three semesters (students enrolled in 2017).

B. DATA PRE-PROCESSING

Because a deep learning model can only process numerical features, data pre-processing is necessary for the

categorical data. For this reason, the two categorical features in the tabular data were converted into a binary form using one-hot encoding. Hence, we ended up with a total 53 number of features as the tabular data. All features were subsequently standardized with Z-Score Standardization, which is formulated as $f(x_i) = \frac{x_i - \mu}{\sigma}$, where x is the value of a feature in a data point i, μ is the mean of the attribute, and σ is the standard deviation of the feature. The features in the dataset and their corresponding pre-processing are listed in Table 2.

TABLE 2. List of the tabular data features.

Feature	Data Type	Pre-processing
Campus Location	Categorical	One-Hot Encoding and Z-Score Standardization
Academic Programs	Categorical	One-Hot Encoding and Z-Score Standardization
Enrollment Year	Numerical	Z-Score Standardization
TOEFL Score	Numerical	Z-Score Standardization
Student Orientation Score	Numerical	Z-Score Standardization
Leave Counts	Numerical	Z-Score Standardization

Furthermore, we also imputed missing values in each series of grades with zero, assuming that these students took semester leave. From each series, the last GPA value was pulled out and used as the target variable. Hence, all series lengths were reduced by one unit (semester). For time-series based analysis, we arranged the input as an array containing 46,670 univariate time-series. One single time-series belongs to one sample (one student) and represents the series of GPA values of that student from the first year up to the year before the recent year. Due to variations in the study duration of students across different academic years, the length of each sample is variable. Unfortunately, the variable-length posed a problem to the deep learning frameworks we used (Keras [23] and Tensorflow [24]). The frameworks require the model to be defined as a static computational graph. To overcome this problem, we padded the samples with shorter lengths than the longest sample with some value that falls outside of the GPA scale. This padding value should be necessarily chosen outside of the GPA scale so that our deep learning model can learn to distinguish between 'real inputs' and padded inputs. For simplicity, we took an arbitrary value of -1.0 as our padding value. We observed that different choices of values (either positives and negatives) did not affect the performance of our model. An illustration of the padded series is depicted in Figure 1.

C. PROPOSED MODEL

For a deep learning model to process time-series data Long Short-Term Memory (LSTM) [22], [25] layer is commonly employed. LSTM is a special form of Recurrent Neural Network (RNN), a deep learning layer that takes input from the previous time-step when processing the current time-step. LSTM improves standard RNN with better long-range dependencies modelling. The novel innovation in LSTM is its memory cell c_t, which is used as an accumulator of the state information. For controlling the gates inside the LSTM layer, the cell is accessed, written, and cleared by three self-parameterized gates: input, forget, and output gate. When receiving input, its information is accumulated to the cell if the input gate i_t is activated. The information from the past cell c_{t-1} could be "forgotten" if the forget gate f_i is switched on. Whether the latest cell output c_t is propagated to the final state h_t is further controlled by the output gate o_t [26]. In summary, the LSTM memory cell is implemented as the following:

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i) \quad (1)$$

$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f) \quad (2)$$

$$c_t = f_t c_{t-1} + i_t \tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \quad (3)$$

$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_t + b_o) \quad (4)$$

$$h_t = o_t \tanh(c_t) \quad (5)$$

where σ is the logistic sigmoid function, c is the cell state, and i, f, and o are the input gate, forget gate, and output gate, respectively. All of the hidden vector h in each gate has the same size. Each gate has its own weights W and a hidden vector h that has the same size for all gates. LSTM is currently one of the most powerful algorithms for time-series analysis that has been applied in many forms of time-series data such as text [27], internet bandwidth [28], and also historical GPA [8].

Taking advantage of the LSTM capability in processing time-series data, our proposed deep learning model combined MLP and LSTM to allow simultaneous processing of tabular and time-series data. To be concise, we called our proposed model MLP-LSTM in this paper. As illustrated in Figure 2, the architecture of the model consists of an MLP branch and an LSTM branch which each has five layers. Each branch receives different data as input. The MLP branch takes the tabular data, while the LSTM branch takes time-series data. The information of the tabular and time-series data was fused by adding the output of each layer in the LSTM branch to the output of the MLP layer at the same level.

D. EXPERIMENT

The first model was an MLP trained with only the tabular data. The second model was also an MLP, but it was trained with the mean of historical GPA data, in addition to the tabular data. The third model was an LSTM trained with only the historical GPA data. The first and third models were included in the experiment to check our hypothesis that the combination of MLP and LSTM is better than models with only MLP and LSTM. Additionally, we compared the proposed model with the second model to see whether MLP-LSTM can outperform the simplest approach to combine tabular and time-series data. Furthermore, based on the MLP and LSTM performance in this comparison, we can measure the individual contribution of the tabular and time-series data. By comparing it to the performance of models with combined data (MLP-LSTM and MLP Mean), we could investigate if the combination of both data is beneficial in term of performance.

Student 1	1.50	2.50	2.20	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
Student 2	2.15	0.00	2.50	3.00	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
Student 3	3.50	3.67	4.00	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
Student 4	2.90	2.00	3.72	3.56	3.21	4.00	3.38	3.50	-1	-1	-1	-1	-1	-1	-1
Student 5	4.00	3.66	3.50	3.73	3.70	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1

GPA entries Padded entries

FIGURE 1. An Illustration of padded series of GPA where one series belongs to one student. The `last_GPA` values have been extracted prior to the padding procedure.

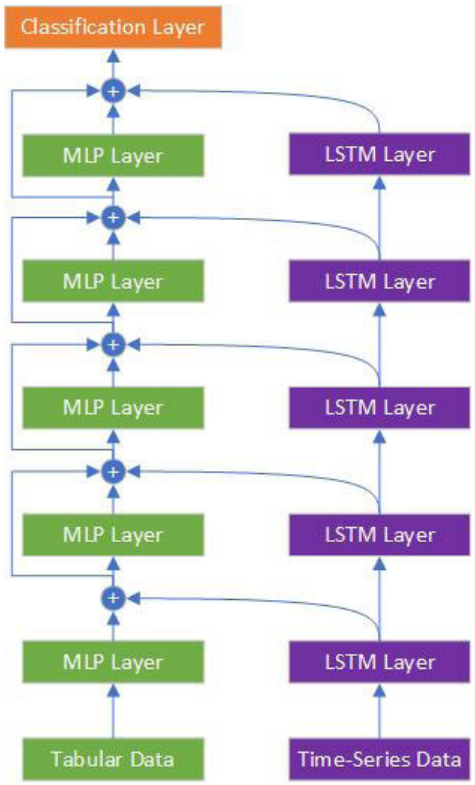

FIGURE 2. The proposed MLP-LSTM architecture.

To further analyze the contribution of each feature in the data, we ran a feature importance algorithm based on permutation importance using the eli5 library in Python. The feature importance algorithm was applied to the MLP Mean model to check the contribution of both the tabular and historical GPA in one model. This is possible because MLP Mean considered both the tabular and historical GPA (in the form of average GPA) in the model. We did not apply the feature importance algorithm to the other model that considered both data, MLP-LSTM, because standard feature importance techniques such as the permutation importance in this study cannot be applied to models with time-series data input. In this case, it is not possible to apply the permutation importance algorithm to the LSTM part of the MLP-LSTM model.

TABLE 3. Models for comparison.

Model Name	Data	Architecture
MLP	Tabular data	MLP
MLP Mean	Tabular data + the mean of the historical GPA as tabular data	MLP
LSTM	Historical GPA data as time-series data	LSTM
MLP-LSTM	Tabular data + historical GPA data as time-series data	Hybrid of MLP and LSTM

To tune the architecture and hyperparameters of all models, we employed the Tree of Parzen estimator [29] in five-fold cross-validation. For the architecture, we fixed the number of layers of all models to five, following the MLP-LSTM architecture. The activation function of layers was set to Rectified Linear Unit (ReLU) [30]. The number of units in each layer of all models was tuned within a set $NL \in \{32, 64, 128, 256, 512\}$, whose weights were initialized with Glorot Uniform Randomization [31]. Specifically for MLP-LSTM, the same number of units was applied to all layers during the hyperparameter tuning process. The learning rate of each model was also included in the hyperparameter tuning, with the option within a set $LR \in \{10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}, 10^{-6}\}$. To reduce overfitting, Dropout [32] was applied to all models. The drop rate was also tuned for each layer in each model within a set $DR \in \{0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6\}$. Specifically for the LSTM layer, we employed recurrent dropout [33] instead of standard dropout. All models were trained using Adam optimization algorithm [34].

E. PERFORMANCE EVALUATION

To compare the performance of MLP-LSTM to the baseline model, we use three regression metrics: Mean Squared Error (MSE), Mean Absolute Error (MAE), and R^2. MSE is calculated as in equation 6:

$$MSE = \frac{1}{N}\sum_{n=1}^{N}(y_n - \hat{y}_n)^2 \tag{6}$$

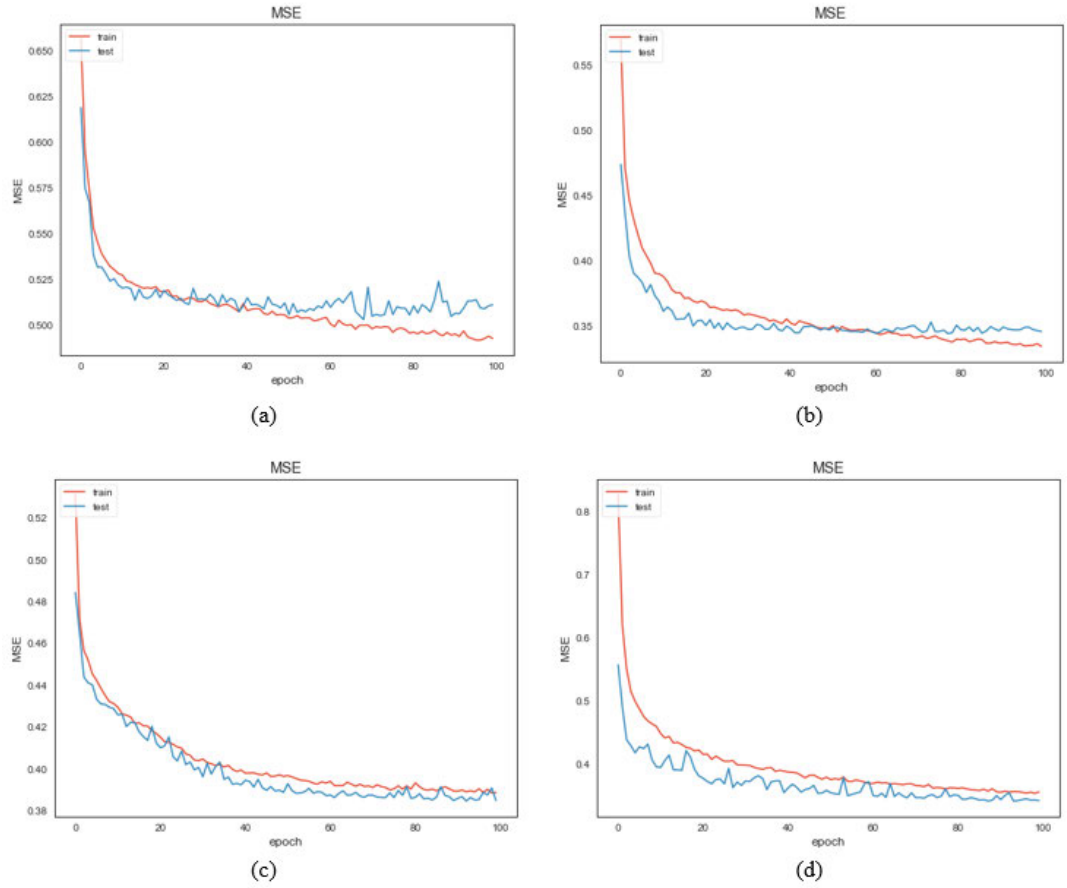

FIGURE 3. Training and validation loss (MSE) plot of the (a) MLP, (b) MLP Mean, (c) LSTM, and (d) MLP-LSTM. A gap between training and validation loss can be observed in the MLP and MLP Mean plot, which indicates that both model suffered overfitting.

where N is the total number of data, y_n is the actual n^{th} GPA entries in the dataset, and $\hat{y}_n$ is the predicted n^{th} GPA entries.

It is known that MSE tends to be more sensitive to large errors. While this characteristic might be appropriate in particular cases, it is not always applicable to all cases. Therefore, it is important to also evaluate the regression performance with Mean Absolute Error (MAE). This metrics has a linear error growth from the absolute difference of 0 towards infinity, as opposed to the MSE that has a quadratic error growth. Low value from both MSE and MAE can ensure the reliability of the model. The calculation of MAE is expressed as in equation 7.

$$MAE = \frac{1}{N}\sum_{n=1}^{N}|y_n - \hat{y}_n| \tag{7}$$

Additionally, we also use R^2 metric to measure the performance for both baseline and MLP-LSTM. R^2 is commonly used to measure regression performance, notably in the statistics field for assessing a linear model. Mathematically, R^2 is defined as in equation 8:

$$R^2 = 1 - \frac{\sum_{n=1}^{N}(y_n - \hat{y}_n)^2}{\sum_{n=1}^{N}(y_n - \bar{y})^2} \tag{8}$$

where $\bar{y}$ is the mean of actual GPA entries in the dataset. The value of R^2 is ranged from 0 to 1, representing the percentage of explained variance of the actual GPA by the assessed model in our case.

IV. RESULTS AND DISCUSSION

Figure 3 shows the performance of all models during the training process. Analyzing the plots, we observed that LSTM and MLP-LSTM did not suffer overfitting, while MLP and MLP Mean did. On the one hand, both LSTM and MLP-LSTM considered the time-series data without aggregation. On the other hand, MLP Mean only received the time-series data that has been aggregated with the mean operation. Based on those facts, we can conclude that the unaggregated information in the time-series data can help models to generalize better. This conclusion is further supported by the fact that the MLP model, which was not exposed to the time-series data, also suffered overfitting.

Meanwhile, the performance for all models on the test set is reported in Table 4. The proposed MLP-LSTM model achieved the best MSE and MAE among all models in the experiment. The R^2 of the proposed model is also larger than other models, which means that the MLP-LSTM explained the variance of the student GPA better than the other models.

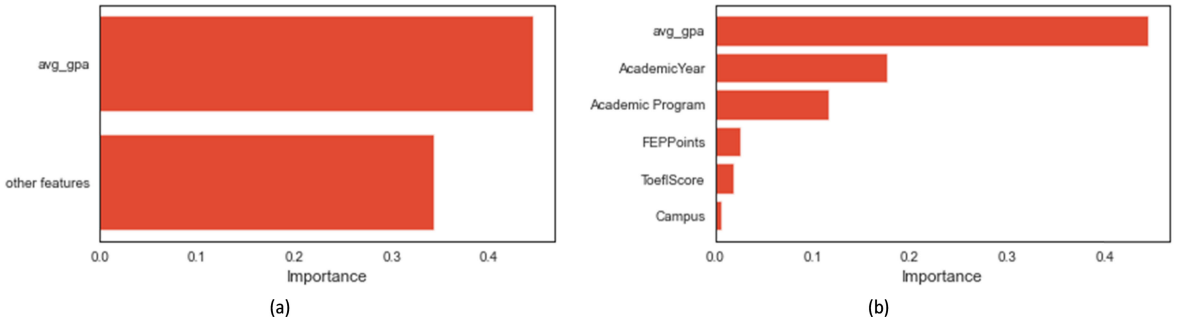

FIGURE 4. (a) Feature importance of the MLP mean model. (b) Feature importance of the MLP mean model, aggregated on the features other than the last GPA.

Meanwhile, the MLP Mean model achieved the second-best performance compared to the model that uses only tabular data (MLP) and only time-series data (LSTM). Ignoring the different characteristics of the models, this fact suggested that the contribution of both the tabular and historical GPA was observable. Thus, the combination of both data in one model was a beneficial factor to the superiority of the MLP-LSTM and MLP Mean model. Moreover, with the fact that MLP-LSTM was better than MLP Mean, we could infer that the more complex architecture also contributed to the predictive performance improvement. Additionally, we observed that the time-series data was more informative than the tabular data, based on the fact that LSTM has better performance than MLP.

In the investigation of the feature importance of the MLP Mean model, we observed that the most important feature was the average GPA with an importance value of 0.4448. Although the average GPA was the most important feature, it did not mean that the other features were not important. Altogether, the importance value of the other features was 0.3423, which is comparable in magnitude to the importance value of the average GPA. In detail, the importance value of the features other than average GPA were respectively 0.1758, 0.1171, 0.0251, 0.0187, and 0.0056 for the academic year, academic program, student orientation score, TOEFL score, and campus location. These importance values were plotted in Figure 4 to visualize the feature importance information more clearly. This feature importance information could provide us a rough estimation of the contribution of the role of the data, especially for each feature, to the performance of all models. However, we should notice that the other model that considered both the tabular and historical GPA data, MLP-LSTM, might provide a different profile of feature importance, which is unfortunately not possible to be derived with standard feature importance algorithms. Despite that, the feature importance information also supported the fact that both data have an observable contribution, as previously demonstrated by the result of the model comparison.

In addition to the general performance of the MLP-LSTM, we found an interesting fact that the performance of the MLP-LSTM is related to the smoothness of the distribution of the actual GPA data. To discover this trait, we split the dataset

TABLE 4. Performance comparison.

Model Architecture	MSE	MAE	R^2-score
MLP	0.5297	0.5092	0.2371
MLP Mean	0.4167	0.3459	0.4819
LSTM	0.4542	0.3908	0.4146
MLP-LSTM	**0.4142**	**0.3418**	**0.4879**

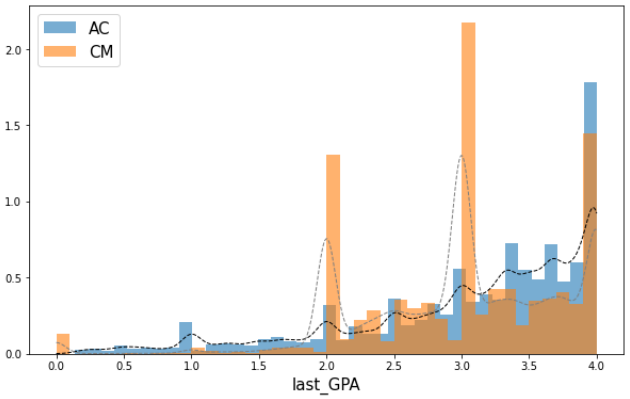

FIGURE 5. The normalized distribution of `last_GPA` grouped by two academic status of students: "active" (AC) and "completed/graduated" (CM). Dashed lines that indicate density plots are shown for clarity.

into two subsets by the two categories in the academic status field: "active" (AC) and "completed/graduated" (CM). As visualized in Figure 5, the AC subset has a relatively smooth distribution of actual GPA value with only a distinct spike at the GPA of 4.00. In contrast, the CM subset exhibits a multimodal distribution with three distinct spikes at values of 2.00, 3.00, and 4.00. After applying the MLP-LSTM to the two subsets, we found that the performance was better for the AC subset with 0.2863 MSE and 0.3628 MAE. These are to be compared with the performance for the CM subset with 0.3807 MSE and 0.4502 MAE.

Moreover, we also made another observation to examine the MLP-LSTM performance in relation to the enrollment year field. In this experiment, the dataset was split into seven subsets according to the student's enrollment year. The performance of the MLP-LSTM for each enrollment year is shown in Figure 6. The general trend was that the error grows larger as the enrollment year decreases. This trend

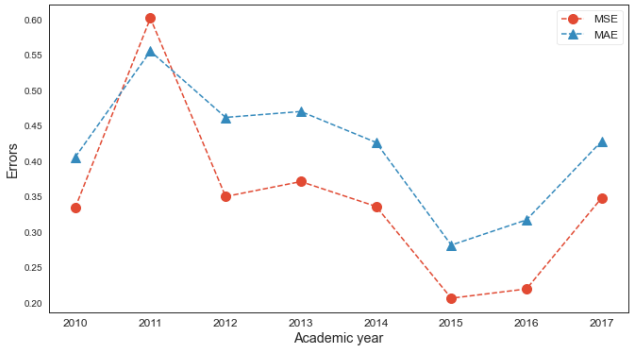

FIGURE 6. The MSE and MAE performance of the MLP-LSTM model on the test set grouped by the enrollment year.

can be explained by the trait of the RNN that usually has less performance as the length of the sequence gets longer. Even though LSTM improves the basic RNN in modeling longer sequences, the trend was still emerged in this study. Noticeably, the error for the enrollment year 2017 suddenly grows larger. This trend might be attributed to the insufficient historical data, where the 2017 subset has the smallest time-series length with only three elements.

V. CONCLUSION

In this study, we proposed MLP-LSTM, a dual-input deep learning model that concurrently processes time-series and tabular data, for modeling student GPA. The result of this study showed that MLP-LSTM was the best model among other models that was exposed to only tabular data, only time-series data, and the combination of tabular and aggregated time-series data. Based on this result, we can conclude that the complex architecture of MLP-LSTM was beneficial to improve the performance of a student GPA model by enabling the use of both unaggregated time-series dan tabular data.

An additional analysis in this paper revealed that the MLP-LSTM needs a smooth target GPA distribution to work well. Unfortunately, the actual data rarely has this convenient feature. To improve the future model, it would be intriguing to incorporate a non-parametric approach in a deep learning model. In the statistics field, it is known that the non-parametric approach can model an arbitrary distribution better than the parametric counterparts. Although it is not a prevalent approach in machine learning studies, we found studies that have attempted to utilize a non-parametric approach for machine learning algorithms [35], [36].

Further observations also showed that the long-range dependencies problem was still apparent even though the LSTM variant was utilized instead of standard RNN. A possible solution for future works is to use transformer [37] instead of RNN variants. Transformer models time-series data with self-attention instead of recurrent connection, which is theoretically better to model long-range dependencies. Transformer has been successfully applied to model text [38]–[40], which can also be viewed as time-series data.

ACKNOWLEDGMENT

The experiments in this study were conducted using NVIDIA Tesla P100 from NVIDIA—BINUS AI Research and Development Center. The dataset was collected and compiled by IT Directorate, Bina Nusantara University.

REFERENCES

[1] A. Zollanvari, R. C. Kizilirmak, Y. H. Kho, and D. Hernández-Torrano, "Predicting students' GPA and developing intervention strategies based on self-regulatory learning behaviors," *IEEE Access*, vol. 5, pp. 23792–23802, 2017.
[2] V. E. Johnson, "An alternative to traditional GPA for evaluating student performance," *Stat. Sci.*, vol. 12, no. 4, pp. 251–269, Nov. 1997.
[3] N. Bosch, "Identifying supportive student factors for mindset interventions: A two-model machine learning approach," *Comput. Educ.*, vol. 167, Jul. 2021, Art. no. 104190.
[4] M. Pojon, "Using machine learning to predict student performance," M.S. thesis, Dept. Natural Sci., Univ. Tampere, Tampere, Finland, 2017.
[5] N. Putpuek, N. Rojanaprasert, K. Atchariyachanvanich, and T. Thamrongthanyawong, "Comparative study of prediction models for final GPA score: A case study of Rajabhat Rajanagarindra University," in *Proc. IEEE/ACIS 17th Int. Conf. Comput. Inf. Sci. (ICIS)*, Jun. 2018, pp. 92–97.
[6] M. F. Musso, C. F. R. Hernández, and E. C. Cascallar, "Predicting key educational outcomes in academic trajectories: A machine-learning approach," *Higher Educ.*, vol. 80, no. 5, pp. 1–20, Nov. 2020.
[7] T. Davidson, "Black-box models and sociological explanations: Predicting high school grade point average using neural networks," *Socius*, vol. 5, Jan. 2019, Art. no. 2378023118817702.
[8] A. P. Patil, K. Ganesan, and A. Kanavalli, "Effective deep learning model to predict student grade point averages," in *Proc. IEEE Int. Conf. Comput. Intell. Comput. Res. (ICCIC)*, Dec. 2017, pp. 1–6.
[9] Z. Iqbal, J. Qadir, A. N. Mian, and F. Kamiran, "Machine learning based student grade prediction: A case study," 2017, *arXiv:1708.08744*. [Online]. Available: http://arxiv.org/abs/1708.08744
[10] A. Algarni, "Data mining in education," *Int. J. Adv. Comput. Sci. Appl.*, vol. 7, no. 6, pp. 456–461, 2016.
[11] M. Imran, S. Latif, D. Mehmood, and M. S. Shah, "Student academic performance prediction using supervised learning techniques," *Int. J. Emerg. Technol. Learn.*, vol. 14, no. 14, pp. 92–104, 2019.
[12] M. A. Al-Barrak and M. Al-Razgan, "Predicting students final GPA using decision trees: A case study," *Int. J. Inf. Educ. Technol.*, vol. 6, no. 7, p. 528, 2016.
[13] S. Hussain, N. A. Dahan, F. M. Ba-Alwib, and N. Ribata, "Educational data mining and analysis of students' academic performance using WEKA," *Indonesian J. Electr. Eng. Comput. Sci.*, vol. 9, no. 2, pp. 447–459, 2018.
[14] A. A. Saa, "Educational data mining & students' performance prediction," *Int. J. Adv. Comput. Sci. Appl.*, vol. 7, no. 5, pp. 212–220, 2016.
[15] L. Ramanathan, S. Dhanda, and D. S. Kumar, "Predicting students' performance using modified ID3 algorithm," *Int. J. Eng. Technol.*, vol. 5, no. 3, pp. 2491–2497, 2013.
[16] M. Nasiri, B. Minaei, and F. Vafaei, "Predicting GPA and academic dismissal in LMS using educational data mining: A case mining," in *Proc. 6th Nat. 3rd Int. Conf. E-Learn. E-Teach.*, Feb. 2012, pp. 53–58.
[17] E. B. Belachew and F. A. Gobena, "Student performance prediction model using machine learning approach: The case of wolkite university," *Int. J. Adv. Res. Comput. Sci. Softw. Eng.*, vol. 7, no. 2, pp. 46–50, Feb. 2017.
[18] A. I. Adekitan and E. Noma-Osaghae, "Data mining approach to predicting the performance of first year student in a university using the admission requirements," *Educ. Inf. Technol.*, vol. 24, no. 2, pp. 1527–1543, Mar. 2019.
[19] F. Ahmad, N. H. Ismail, and A. A. Aziz, "The prediction of students' academic performance using classification data mining techniques," *Appl. Math. Sci.*, vol. 9, no. 192, pp. 6415–6426, 2015.
[20] P. M. Arsad, N. Buniyamin, and J.-L.-A. Manan, "A neural network students' performance prediction model (NNSPPM)," in *Proc. IEEE Int. Conf. Smart Instrum., Meas. Appl. (ICSIMA)*, Nov. 2013, pp. 1–5.
[21] M. F. Sikder, M. J. Uddin, and S. Halder, "Predicting students yearly performance using neural network: A case study of BSMRSTU," in *Proc. 5th Int. Conf. Inform., Electron. Vis. (ICIEV)*, May 2016, pp. 524–529.
[22] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[23] F. Chollet *et al.* (2015). *Keras*. [Online]. Available: https://keras.io and https://keras.io/getting_started/faq/#how-should-i-cite-keras
[24] M. Abadi *et al.*, "TensorFlow: A system for large-scale machine learning," in *Proc. 12th USENIX Symp. Operating Syst. Design Implement. (OSDI)*, 2016, pp. 265–283.
[25] F. A. Gers, J. Schmidhuber, and F. Cummins, "Learning to forget: Continual prediction with LSTM," *Neural Comput.*, vol. 12, no. 10, pp. 2451–2471, Oct. 2000.
[26] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-C. Woo, "Convolutional LSTM network: A machine learning approach for precipitation nowcasting," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 28, 2015, pp. 802–810.
[27] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, vol. 27, 2014, pp. 3104–3112.
[28] T. W. Cenggoro and I. Siahaan, "Dynamic bandwidth management based on traffic prediction using deep long short term memory," in *Proc. 2nd Int. Conf. Sci. Inf. Technol. (ICSITech)*, Oct. 2016, pp. 318–323.
[29] J. Bergstra, R. Bardenet, Y. Bengio, and B. Kégl, "Algorithms for hyper-parameter optimization," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 24, 2011, pp. 2546–2554.
[30] V. Nair and G. E. Hinton, "Rectified linear units improve restricted Boltzmann machines," in *Proc. 27th Int. Conf. Mach. Learn. (ICML)*, 2010, pp. 807–814.
[31] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proc. 13th Int. Conf. Artif. Intell. Statist.*, 2010, pp. 249–256.
[32] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.
[33] S. Semeniuta, A. Severyn, and E. Barth, "Recurrent dropout without memory loss," in *Proc. 26th Int. Conf. Comput. Linguistics, Tech. Papers (COLING)*, Osaka, Japan, Dec. 2016, pp. 1757–1766. [Online]. Available: https://www.aclweb.org/anthology/C16-1165
[34] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," Dec. 2014, *arXiv:1412.6980*. [Online]. Available: https://arxiv.org/abs/1412.6980
[35] E. Trentin and A. Freno, "Unsupervised nonparametric density estimation: A neural network approach," in *Proc. Int. Joint Conf. Neural Netw.*, Jun. 2009, pp. 3140–3147.
[36] O. Rippel and R. P. Adams, "High-dimensional probability estimation with deep density models," 2013, *arXiv:1302.5125*. [Online]. Available: http://arxiv.org/abs/1302.5125
[37] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5998–6008.
[38] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," *J. Mach. Learn. Res.*, vol. 21, no. 140, pp. 1–67, 2020.
[39] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut, "ALBERT: A lite BERT for self-supervised learning of language representations," 2019, *arXiv:1909.11942*. [Online]. Available: http://arxiv.org/abs/1909.11942
[40] W. Wang, B. Bi, M. Yan, C. Wu, Z. Bao, J. Xia, L. Peng, and L. Si, "StructBERT: Incorporating language structures into pre-training for deep language understanding," 2019, *arXiv:1908.04577*. [Online]. Available: http://arxiv.org/abs/1908.04577

HARJANTO PRABOWO received the bachelor's degree in electrical engineering from Diponegoro University, the master's degree in information system management from Bina Nusantara University, Jakarta, Indonesia, and the Ph.D. degree in business management from Padjadjaran University. He is currently the Rector and a Professor of management information system with Bina Nusantara University. His research interests include knowledge management and information systems.

ALAM AHMAD HIDAYAT received the B.Sc. degree in physics from the Institut Teknologi Bandung, Indonesia, in 2014, and the M.Sc. degree in theoretical physics with the University of Bonn, Germany, in 2018.

He is currently a Research Assistant with the Bioinformatics and Data Science Research Center (BDSRC), Bina Nusantara University, Indonesia. His research interests include applications of deep learning and statistical models to analyze data from diverse topics, including health sciences.

TJENG WAWAN CENGGORO received the bachelor's degree in information technology from STMIK Widya Cipta Dharma, and the master's degree in information technology from Bina Nusantara University. He is currently an AI Researcher whose focus is in the development of deep learning algorithms for application in computer vision, natural language processing, and bioinformatics. He is also a Certified Instructor with the NVIDIA Deep Learning Institute. He led several research projects that utilize deep learning for computer vision, which is applied to indoor video analytics and plant phenotyping. He has published over 20 peer-reviewed publications and reviewed for prestigious journals, such as *Scientific Reports* and IEEE Access. He also holds two copyrights for AI-based video analytics software.

REZA RAHUTOMO received the bachelor's degree in information system and the master's degree in magister management and information system from Bina Nusantara University, in 2012 and 2018, respectively. He is currently working with the Bioinformatics and Data Science Research Center (BDSRC), as a Researcher. His research interests include deep learning modeling for time-series analysis, tabular data, and natural language processing.

KARTIKA PURWANDARI received the bachelor's degree in information technology from Brawijaya University, in 2015, and the master's degree in computer science from National Central University, Taiwan, in 2019. Since 2019, she has been a Researcher with the Bioinformatics and Data Science Research Center (BDSRC). Her research interests include deep learning model for signal, image, and text data.

BENS PARDAMEAN received the bachelor's degree in computer science and the master's degree in computer education from California State University, Los Angeles, and the Doctoral degree in informative research from the University of Southern California (USC). He has over 30 years of global experience in information technology, bioinformatics, and education. After successfully leading the Bioinformatics Research Interest Group, he currently holds a dual appointment as the Director of the Bioinformatics and Data Science Research Center (BDSRC), and an Associate Professor of computer science with Bina Nusantara University, Jakarta, Indonesia.