International Journal for Multidisciplinary Research (IJFMR)
E-ISSN: 2582-2160 ● Website: www.ijfmr.com ● Email: editor@ijfmr.com

A Deep Learning Model Using CNN & LSTM to Forecast Student Learning Outcomes in Learning Management System

Chaitrali Kadam[1], Darshan Deshmukh[2], Tejas Saraf[3]

[1,2]Student, Computer Science, Savitribai Phule Pune University
[3]Student, Information Technology, Savitribai Phule Pune University

Abstract:
The administration, monitoring, and reporting of educational activities are being done more and more with Learning Management Systems (LMSs). Blackboard is one such extensively utilized LMS at universities all over the world. This is because it may be used to match learning material pieces, student-teacher and student-student interactions, and assessment activities to predetermined objectives and student learning outcomes. This study aimed to determine the predictive power of various KPIs obtained from students' Blackboard interactions in order to forecast students' learning outcomes. Deep learning algorithms to forecast academic achievement were looked at as part of a mixed-methods study design.
The degree of linear relationship between these factors and measures of student performance was ascertained by correlational tests. Out of the four models that were assessed, the CNN-LSTM predictive model proved to be the most effective because it combined long short-term memory with convolutional neural networks. The primary inference made from this data is that the CNN-LSTM technique might result in solutions that maximize and enhance how universities use the Blackboard LMS.

Keyword: - Learning management systems, student prediction, deep learning, CNN, LSTM.

1. Introduction:

Learning Management Systems are applications of carefully selected software that are used in higher education institutions to facilitate learning. It functions as an automated system for organizing, tracking, and reporting educational activities and learning outcomes. By identifying and evaluating students and institutional learning goals, monitoring progress toward the goals, and gathering and presenting data for learning process supervision, learning management system developed and put into use to help streamline the education process, which includes teaching, learning, and administration. This makes an LMS helpful not just for distributing learning materials but also for monitoring student compliance and uptake as well as for analyzing knowledge and skill gaps.

Blackboard is a popular Learning Management System utilized by universities worldwide. Education establishments use this tech platform to facilitate the sharing of crucial learning resources and content, as well as assignments and reports from students and instructor announcements. Furthermore, real-time activities such as online chat rooms and discussion boards for student-teacher interactions as well as the exchange of papers, resources, and inquiries are made possible by Blackboard technology. As technology progresses, learning management systems are being used more frequently to track student performance and predict learning objectives.

This research aims to monitor students' performance throughout the educational process and forecast their learning outcomes. Data generated automatically from the online LMS is used in the process. We specifically look at how seven Key Performance Indicators in Blackboard that have been carefully chosen assist teachers in forecasting students' learning outcomes. Utilizing artificial intelligence models is necessary to accomplish this goal. Machine learning refers to the kinds of technologies and algorithms which support systems to recognize patterns, make decisions, and generate improvement through experience.

Artificial intelligence, on the other hand, is the general capacity of computers to imitate human thought and perform tasks in real-world setting. Data representations with many degrees of abstraction can be "learned" by computer models consisting of numerous processing layers thanks to deep learning. Although the artificial intelligence field has not yet entirely solved these problems, deep learning is linked to notable advancements in issue solving. The reason for this is that by employing a backpropagation technique, it can efficiently identify complex structures hidden inside big data sets. In order to compute the representation in each layer from the representation in the layer preceding it, the technique is employed to determine how the machine should modify its internal parameters. Consequently, each subsequent level of the deep learning process "learns" to convert the input data into a representation that is a little bit more abstract and composite.

This research develops a new deep learning model to predict student performance combining long short-term memory (LSTM) and convolutional neural networks (CNN). It gives institutions useful information that may be used to guarantee the caliber of their offerings. Along with guaranteeing student success, it can aid in strategy creation by giving them individualized guidance based on their expected performance. To summarize, the primary findings of this study are outlined below:

1. An examination of the degree of assistance given to colleges so they may make use of the student meta-data produced by the online LMS.
2. Examining deep learning algorithms to forecast academic achievement.
3. Correlation and time series analysis of university student performance by attended course.

2. Proposed System:

The suggested method's design purpose is to use CNN and LSTM to predict student learning outcomes and performance in an LMS at a university. The performance prediction accuracy was increased in comparison to state-of-the-art methods by integrating two techniques: 1) CNN to extract useful features from the data, and 2) LSTM to determine the dependency of data in time series data.

The below Figure displays the prediction framework, and the following is an introduction to the primary steps of the Convolution neural network and Long Short-term Models:

1. Gather student data via Blackboard, a popular LMS that's meant to help universities store student data effectively throughout time based on performance forecasts.

2. As indicated in Table, choose the important characteristics and exclude anomalous results for every course to get accurate information on students' performance. Seven courses were chosen, and each course's seven features were examined.

Feature ID	Feature Name
F1	Total Hours spent on the course
F2	No. of times logged into the course
F3	Total no of downloads per student
F4	Total no of projects & assignments from projects
F5	Total number of attended exams and quizzes
F6	Total number of messages sent from students
F7	Total no of participations per student in the course

Table 1: Student performance during the first and second semesters

1. Create STraining and STesting sets from the student data S. The training set size (STraining) is t, the test set size (STesting) is V − t, where V is the size of the students' features in St. The time series data for the students is St.
2. In the CNN model, the max pooling layer and the convolutional layers employ a feeding strain to extract the features of the students. The formula for calculating input data is Sconv = Strain ∗ K, Smaxpool = Max(Sconv). In this case, Sconv is defined as the training data's convolution layer result.
3. K is the convolutional kernel, or the convolution window size, while the symbol ∗ stands for the convolution procedure. Smaxpool is the outcome of max pooling the CNN layer.
4. Feed the LSTM model with the student Smaxpool's extracted features. Three gates are used to process Smaxpool: the LSTM's input gate, forget gate, and output gate.
5. Training and learning yield prediction results on STesting sets.

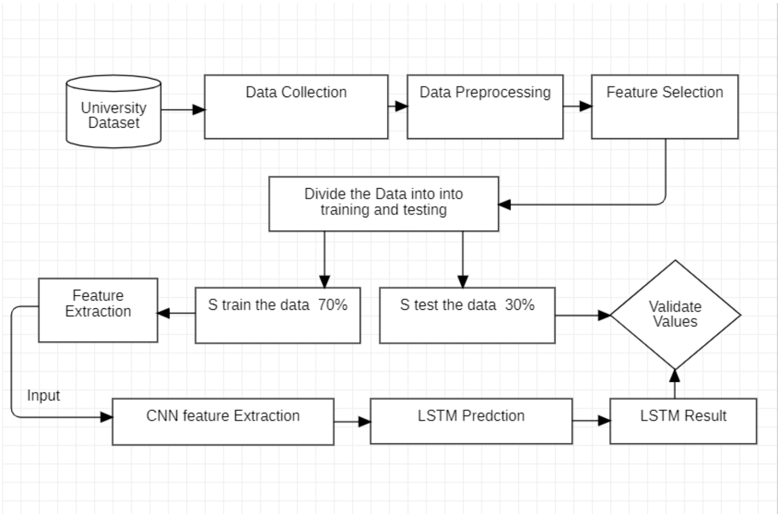

Fig 1. Proposed CNN-LSTM architecture diagram.

2.1. Data Collection:

The report of students' KPIs based on seven general preparation courses on Blackboard was where the student performance dataset was found. The report is a composite of the reports written for every student, every student, and every course. The targeted university's IT department provides these reports, which contain electronic data related to the Blackboard remote learning system. They list the general prerequisites for all undergraduates at the university based on their areas of expertise. Students' cumulative data, which includes: 1) courses; 2) in-course activities; 3) methods of assessment; 4) grades; and 5) resources, is how this is provided.

Since the dataset only includes attributes that represent students' academic achievements and online activities, we used it to collect student performance data. With the seven parameters listed in Table 3, the dataset comprises 35,000 student records. Four subjects were covered in seven classes taken by each student: Arabic language, English, mathematics, and physics. This means that each student's student record is 7 Courses × 7 Features = 49, which is 35, 000 × 49 = 1, 715, 000 for the entire dataset.

2.2 Data- preprocessing:

An essential phase in the data mining process is data preparation. It describes the steps taken to prepare data for analysis, such as cleaning, converting, and integrating it. Enhancing the quality of the data and tailoring it to the particular data mining task are the objectives of data preprocessing.

1. **Data cleaning:** It is the process of finding and fixing mistakes or inconsistencies in the data, such as duplicates, outliers, and missing numbers. Data cleaning can be accomplished with a variety of methods, including imputation, removal, and transformation.
2. **Data integration**: It is the process of merging information from several sources to produce a single, cohesive dataset. Because it involves handling data with various formats, structures, and semantics, data integration can be difficult. Data integration can be accomplished by using methods like record linkage and data fusion.
3. **Data transformation**: It is the process of transforming the data into a format that is appropriate for analysis. Normalization, standardization, and discretization are common methods used in data transformation. While standardization is used to change the data to have a zero mean and unit variance, normalization is used to scale the data to a common range. Continuous data can be discretized using the discretization process.
4. **Data reduction:** It is the process of cutting down on the dataset's size without sacrificing any of its crucial information. Techniques like feature selection and feature extraction can be used to reduce data. While feature extraction entails converting the data into a lower-dimensional space while maintaining the crucial information, feature selection entails choosing a subset of pertinent characteristics from the dataset

2.3 Feature Selection:

Feature extraction pulls important features to store in a feature vector by importing student performance data from the Blackboard system. Table 3 gives an explanation of the features.

In this work, the KPIs data is a 1D vector of student features. The prediction CNN-LSTM issue takes into account the chosen student features as an input and sequence labeling model, as illustrated in Fig. 2. A feature vector is created and saved with the output labels as well as the retrieved student features built as sequences. An input sequence Vc[i] = [Vc[i],f1, Vc[i],f2, Vc[i],f3, Vc[i],f4, Vc[i],f5, Vc[i],f6 Vc[i],f7] with i = 1 to n = 7 is used to represent this. During the first and second semesters, these characteristics were looked at for a variety of preparation year courses.

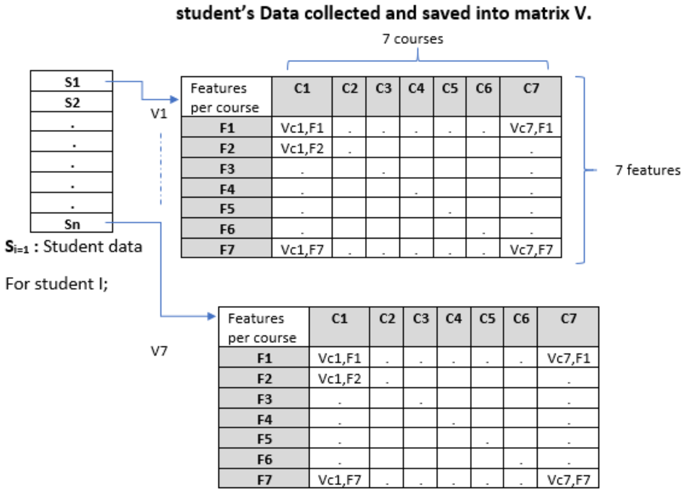

Fig 2. Data structure of collected student's data.

2.4 Feature Extraction Using CNN Model.

The performance of each course's students was predicted by a novel deep learning model that combined CNN and LSTM. LSTM was utilized for performance prediction in the suggested prediction model, whereas CNN was used to extract the student feature time series. This obtained more dependable forecasts by fully utilizing the student data time sequence. Furthermore, our system demonstrated strong prediction accuracy and was more adept at forecasting student performance within our higher education institution when compared to CNN, LSTM, RNN, and CNN-RNN assessment indexes.

The CNN model is a type of feedforward neural network and CNN is a popular deep learning model that performs well in various applications, including image identification, healthcare analysis, and predictive analytics. CNN is mostly made up of two layers: the max pooling layer and the convolution layer, and it may be used to predict time series data with effectiveness. Our approach included one pooling layer and four convolution layers. All convolution layers have many convolution kernels, and their final equation is lt = tan h (xt ∗ kt + bt) after the convolution process.

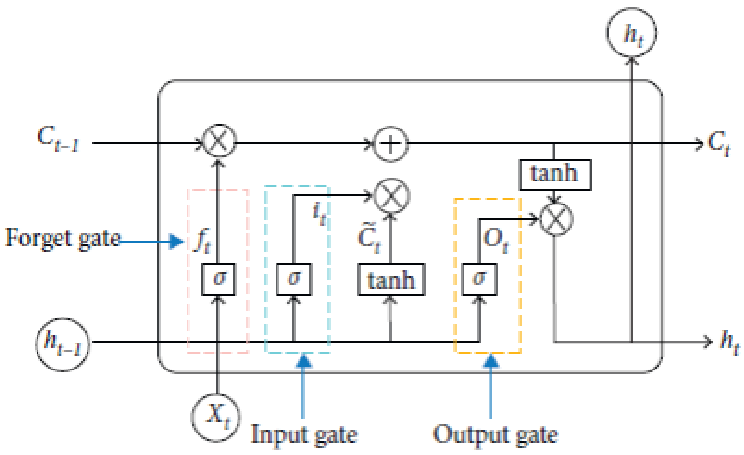

Fig 3. LSTM model Architecture.

The following is the LSTM calculating procedure:

a) The forget gate uses the current time's input parameter and the output value of the last cell as its input. The forget gate's output value is computed as follows:

$ft = \sigma\, Wf \cdot [ht{-}1, xt] + bf$

where the forget gate bias function, denoted by bf, is defined and the output is within the range of ft ~ [0, 1]. Wf is the forget gate's weighted valued. The current time's input value is denoted by xt, and its output value is ht−1.

b) The input gate takes xt and ht−1 as input values. The input gate's output and memory cell states are computed as follows:

$it = \sigma\, (Wi \cdot [ht{-}1, xt] + bi),$

$t = tanhtanh\, (Wc \cdot [ht{-}1, xt] + bc),$

Where (0, 1)} is the weighted coefficient of the input gate, wi is the candidate input gate's weight, bc is its bias, and bi is the input gate bias function.

c) Change the current cell state as follows:

$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t,$

where Ct's value range is between 0 and 1.

d) At time t, the output gate receives the input values of output ht−1 and input xt, and the output Ot of the output gate has the following definition:

$ot = \sigma\, (Wo\, [ht{-}1, xt] + bo),$

In this case, ot (0, 1), wo denotes the output gate's weight, and bo denotes the output gate's bias function.

e) The cell's state and the output gate output are used by the LSTM output value in the following manner:

$ht = ot * tanhtanh\, (Ct),$

The following was the introduction of the training and prediction steps:

A. Set the training parameter for the CNN model: The weight coefficients are represented by Wi, and the deviation, bi, is evident in Table.
B. Get the students' data in time series t ready for the training set STrain. For 70% of the dataset, use the S train.
C. In accordance with Fig. 1, fed to the STrain as input layer, transmitted to the CN layer for the purpose of calculating S conv, and finally transferred to the max pooling layer as S maxpool.
D. After extracting the effective features S maxpool, the LSTM model is fed the data, producing the output result ht that is displayed in Fig. 3.
E. The completely connected layer is then used to estimate the expected values, xˆi.
F. Data xi is standardized to enhance training in the model because there is a reasonable gap between student data in the input gate. The input data is normalized using the z-score approach in the following ways:

$$y_i = \frac{x_i - \bar{x}}{s},$$
$$x_i = y_i * s + \bar{x},$$

wherein xi is the input data for each student, x is the average student performance, and yi is defined as the normalized value. s represents xi's standard deviation.

G. Error estimation: by comparing yˆi with the observed value of this data group yi, the associated error is estimated. The estimated value is determined by the output gate.
H. Determine whether the weights Wi satisfy a certain criterion. A predefined number of epochs is chosen so that the training model can be finished with the lowest failure rate. Proceed to step 10 after updating the CNN-LSTM model; alternatively, move to step 9.
I. Proceed to step d to continue training the model by propagating the determined error backward and adjusting the weight and bias function for each layer.
J. Preserve the learned model for future use.
K. To anticipate their values, set up an input testing S test with a dataset size of 30%.

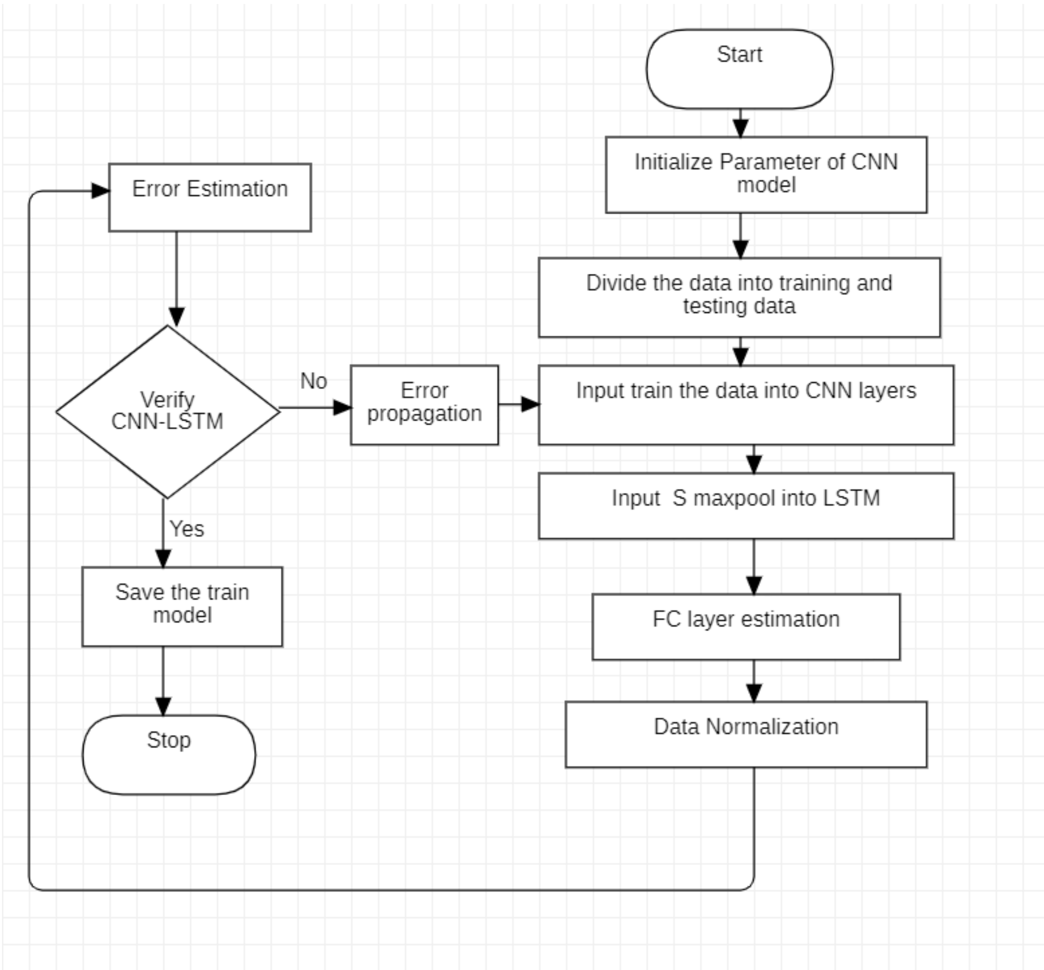

Fig 4. Flowchart of prediction process

3. Result:

In our experiment, each course Ci has seven features [F1, F2, F3, F4, F5, F6, F7], as shown in Table.1. Results of the CNN-LSTM prediction method's F1 score. Seven courses (Ci. =1,2,3,...,7) were chosen from each Blackboard student record. The pupils use these noteworthy attributes as KPI. This assisted in predicting the study habits of the students through the use of a CNN-LSTM-based deep learning model. These characteristics were looked at for a variety of first- and second-semester preparatory year courses. The results of the experiment, which are displayed in Fig. 5, indicate that the suggested CNN-LSTM technique used seven features in total to reach a precision score of 94.2%, whereas the proposed method used just three features (F1, F2, F4) to achieve a precision score of 90.94%. These attributes stand for login, reading course duration, and download volume. It displays the number of students who are considering the chosen courses.

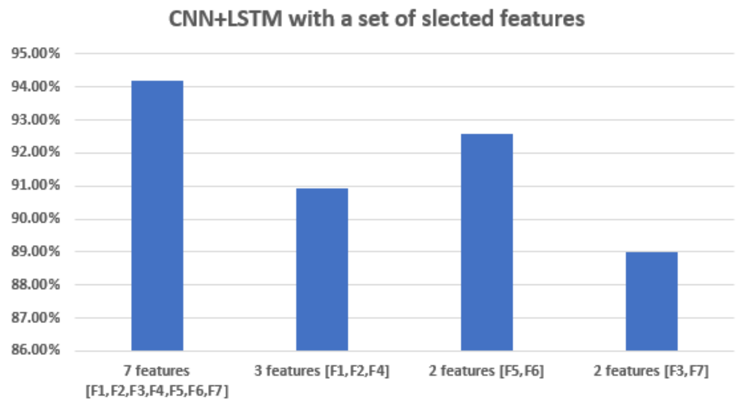

Fig 5: Performance evaluation of the proposed method based on selected features (Fi)

As illustrated in Fig. 6, TP + FN has an F1-score of almost 0.9359, which is superior to either CNN or LSTM alone.

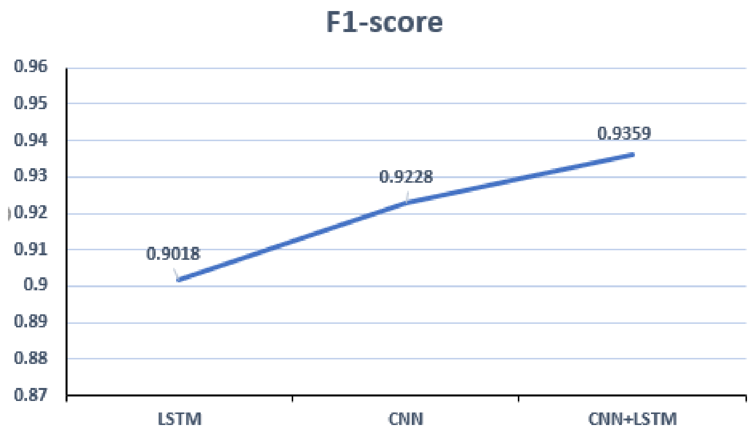

Fig 6: F1-score results of CNN-LSTM prediction method.

4. Conclusion:

The data used in this study was gathered through student interactions with an LMS, such as Blackboard. Prediction accuracy and prediction error were used to gauge how well our deep learning CNN-LSTM model predicted student performance. For every student, seven by seven features were chosen to be entered into the CNN layers. The deep learning model included three factors: the size of the CNN convolution filter, the number of neurons in the LSTM, and the batch size of the LSTM, to influence prediction accuracy and prediction error.

The CNN-LSTM model's drawback is that it takes a lot of time to increase the LSTM batch size, CNN layers, and filters' sizes. It's also important to remember that various feature selection techniques can be applied to display student performance. Additionally, the CNN-LSTM deep learning model requires more time to train than other models, but it can learn more efficiently and has higher processing power. Therefore, a shallow, light-weight deep learning model with a short training time and adequate processing capacity could be used in future research.

5: References:

1. J. Uziak, M. T. Oladiran, E. Lorencowicz, and K. Becker, ''Students' and instructor's perspective on the use of blackboard platform for delivering an engineering course,'' *Electron. J. e-Learn.*, vol. 16, no. 1, pp. pp1–15, Feb. 2018.
2. J. A. Alokluk, ''The effectiveness of blackboard system, uses and limi tations in information management,'' *Intell. Inf. Manage.*, vol. 10, no. 6, p. 133, 2018.
3. I. Osman, M. Nasir, and R. Alzoubi, ''Blackboard usage: An investigative study among CCSE female faculty staff and students at University of Hail,'' *J. Econ. Manage. Perspect.*, vol. 11, no. 2, pp. 508–515, 2017.
4. D. Carmichael and C. MacEachen, ''Heuristic evaluation of the use of Blackboard & Facebook groups in computing higher education,'' *Int. J. Modern Educ. Comput. Sci.*, vol. 9, no. 6, pp. 1–8, Jun. 2017
5. M. J. Verdu, J.-P. De Castro, L. M. Regueras, and A. Corell, ''MSocial: Practical integration of social learning analytics into Moodle,'' *IEEE Access*, vol. 9, pp. 23705–23716, 2021.
6. C. Greenhow and E. Askari, ''Learning and teaching with social network sites: A decade of research in K-12 related education,'' *Educ. Inf. Technol.* vol. 22, no. 2, pp. 623–645, Mar. 2017.
7. M. D. Lytras, N. R. Aljohani, A. Visvizi, P. Ordonez De Pablos, and D. Gasevic, ''Advanced decision-making in higher education: Learning analytics research and key performance indicators,'' *Behav. Inf. Technol.*, vol. 37, nos. 10–11, pp. 937–940, Nov. 2018.
8. X. Zhang, Y. Meng, P. O. de Pablos, and Y. Sun, ''Learning analytics in collaborative learning supported by slack: From the perspective of engagement,'' *Comput. Hum. Behav.*, vol. 92, pp. 625–633, Mar. 2019.
9. A. Van Barneveld, K. E. Arnold, and J. P. Campbell, ''Analytics in higher education: Establishing a common language,'' *Educause Learn. Initiative*, vol. 1, no. 1, pp. 1–2, 2012.
10. R. Ferguson, ''Learning analytics: Drivers, developments and challenges,'' *Int. J. Technol. Enhanced Learn.*, vol. 4, nos. 5–6, pp. 304–317, 2012.