International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
IJISAE
ISSN:2147-6799 www.ijisae.org Original Research Paper

Predicting Student Performance in Higher Education Using A Cluster-Based Distributed Architecture (CDA)

[1]Manju Bargavi Sankar Krishnamoorthy, [2]Ajay Chakravarty, [3]Kapilesh Jadhav, [4]Rajeev Gupta

Submitted:23/04/2023 **Revised:**12/06/2023 **Accepted:**22/06/2023

Abstract: In recent years, academics from many related study fields worldwide have begun to focus on educational data mining (EDM). To help academic planners in higher education institutions make better decisions, suggestions can be made using the information gained from the EDM. Various prediction models have been put out in the literature to forecast student performance. This paper suggests a distributed cluster-based architecture (CDA) for predicting student performance. The proposed CDA indicates clustering via water wave optimization based on K-means cluster and deep neural network (WWO-KMC-DNN), feature extraction using Multi-Linear Discriminant Analysis (M-LDA), and feature fusion using a Bayesian network. In the suggested design, the WWO algorithm is used to determine the DNN ideal weights. Accuracy, prediction rate, mean square error, and root mean square error is monitored in a real-time database to evaluate the proposed task. Using the MSE and RMSE values from the data, the proposed WWO-KMC-DNN model outperforms other models.

Keywords:*Student performance, higher education Prediction, educational data mining (EDM), Water wave optimization based K-means cluster and deep neural network (WWO-KMC-DNN), cluster-based distributed architecture (CDA)*

1. Introduction

Higher education institutions, their specific schools, departments, and individual academics have long been interested in finding ways to ensure that students, once enrolled, stay in school, complete their degrees, and get the most out of their education as possible. The terms student engagement and student retention encompass these two interrelated issues. In terms of research, student retention is the more established issue. It was traditionally referred to by other, less favorable labels such as student withdrawal, attrition, and dropout. Although it is a more recent issue, student engagement, which involves the student participating as fully as possible in their higher education experience, clearly represents a solution to the retention issue. In other words, a student is less likely to leave higher education voluntarily before finishing their studies the more involved they are with their education and the institution from which they obtain information [1]. In particular, classroom behaviors and interactions between teachers and students may moderate the impact of teachers' traits on student outcomes. Conversely, teacher competency might be a significant lever that can be used, for example, through professional development programs, to improve the standard of instruction and student results.

However, studies have not systematically examined the relationships between teacher competence, teaching quality, and student outcomes, particularly in elementary science education. It is difficult to determine how instructional contexts and teacher competence in scientific education relate. In many countries, reform initiatives are centered on inquiry-based learning since it is crucial for fostering students' conceptual understanding of scientific phenomena. Scholars are concluding that scientific education should be viewed as a sense-making process that can be aided by techniques like developing inquiries, conducting experiments, and discussing the outcomes. There is a lot of demand for science instructors to advance science literacy, and it can be challenging to integrate inquiry-based learning in the classroom. However, elementary school teachers in Germany, like in many other nations, are typically generalists who lack formal training in a science-related field. Elementary science teachers frequently show reluctance to teach science due to their poor comprehension of pedagogical topics and low self-efficacy. Due to their lack of specialized university training or professional development in science education, early childhood educators may have comparable expectations for failure while teaching science to young children as elementary school teachers. Primary school teachers are

1Professor, Department of Computer Science and IT, Jain(Deemed-to-be University), Bangalore-27, India, Email Id: b.manju@jainuniversity.ac.in

2Assistant Professor, College of Computing Science and Information Technology, Teerthanker Mahaveer University, Moradabad, Uttar Pradesh, India, Email id: ajay.chakravarty1@gmail.com

3Professor, School of Engineering & Technology, Jaipur National University, Jaipur, india, Email Id: kapilesh@jnujaipur.ac.in

4Assistant Professor, School of Management & Commerce, Dev Bhoomi Uttarakhand University, Uttarakhand, India, Email Id: somc.rajeev@dbuu.ac.in

challenged to support young children's learning while teaching science. Therefore, studying more about the causes of excellent science teaching will help us improve educational quality and, in turn, student learning [2]. One of the most significant issues in education is addressing the psychological requirements of teenagers who have lost interest in school. The significance of satisfying the need to belong in educational situations has received more emphasis in recent decades. Researchers stress the importance of a supportive learning environment in the classroom that fosters a sense of belonging and community among students. This sense of connection is often called a "school spirit." Students feel a sense of belonging when they experience high social acceptance, respect, inclusion, and support from their peers at school. Students with a strong sense of school community have higher levels of motivation, social and emotional well-being (e.g., increased self-esteem), and academic achievement.

In contrast, those who don't have a strong sense of school community are more likely to drop out of school [3]. As a result, we suggested the CDA for predicting student performance in higher education. This initiative aimed to CDA based on clusters for predicting student performance. Second, the WWO-KMC-DNN algorithm is created by merging WWO-KMC-DNN and is used to forecast student achievement based on previously recorded academic performance.

2. Related Work

The research [4] evaluated the effects of students' emotional, cognitive, and behavioral engagement with TEL on their grades and how motivation levels differently predict involvement across various types of TEL. To forecast student performance, the study [5] developed a classifier based on an ensemble meta-based tree model (EMT) and conducted in-depth statistical evaluations of the selected characteristics and their impact on the performance value. Study [6] demonstrated decision-making structures at HEIs and their efficiency in assisting institutional governance. New findings on how academic pressure affects students' physical and mental health, including depression, anxiety, sleep problems, and substance addiction, were published in a recent study [7]. Academic performance, self-efficacy, accomplishment orientation goals, and the parallel and serial mediation models were all investigated in the study [8]. To directly accomplish the primary purpose of performance prediction, the study's authors [9] first create a general Exercise-Enhanced Recurrent Neural Network (EERNN) framework based on student exercise logs and the text content of exercises. Study [10] looked at how students' enthusiasm to learn, readiness to learn, and confidence in their ability to learn to live online during the coronavirus pandemic. This article presents results from a significant study, allowing for some correction. Factors such as instructor type, class size, the percentage of students who need special education, teachers' attitudes toward inclusive education, and the availability of support were among those studied in the study [11], along with teacher burnout and its three sub-domains. According to the research [12], a brand-new adaptive prediction model that has been trained individually for each course and exclusively uses student grades is shown the entire institution is subjected to a thorough analysis to see how accurate its performance is. The research [13] described a two-level classification technique for estimating when kids will graduate from high school. There are two critical components to the suggested algorithm. Several factors influence whether or not college students use mobile learning apps, and a study [14] applied the Unified Theory of Adoption and Use of Technology (UTAUT) model to examine this phenomenon. This study [15] used six data mining techniques implemented in the KNIME and Orange platforms to analyze first-year students' academic success as assessed by cumulative grade point average and degree class. To pinpoint areas of the student's learning experience that could be improved in subsequent versions of the learning design, the study [16] looked at how the students responded to this redesign. Study [20] present a Deep Neural Network-based approach for predicting students' future academic achievement. Predicting student success is crucial in the education sector, and analyzing their current situation is essential for making gains in that regard.

3. Methodology

Face We detail the suggested dispersed construction, built on clusters, and can forecast a student's future performance. The proposed cluster-based distributed architecture is depicted in Fig.1. Two distributed layers, labeled Distributed Layer 1 and Distributed Layer 2, and two merging layers, labeled merging Layer 1 and Merger Layer 2, make up the four-tiered prediction approach proposed. Additional information sources, such as database D, are made available to the proposed architecture, and information for the prediction method is acquired from institutional websites. The distributed layer 1 data sources all include information about the students enrolled at each school. In distributed layer 1, data sources are grouped, and any gaps in knowledge are filled in by making assumptions. Each merger combines the clusters in the first layer (the unions) and then is sent to the second layer (the dispersion). For the WWO-KMC-DNN system, distributed layer 2 is responsible for retrieving the merged data set's features and delivering them for training. The LWDBN predicts the student's success over the following two semesters based on their attributes. Then, as expected, the data about the students' forecasted semester grades are transmitted over merger layer 2.

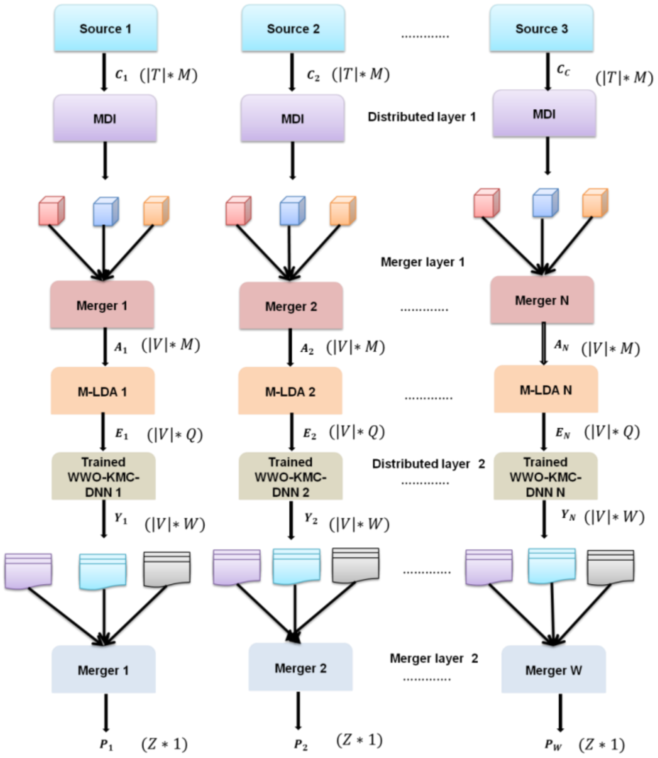

Fig.1. Methodology of CDA

Data gathering is one of the biggest obstacles in creating a prediction model, and the data must be accurate and true. The student's academic records from the college websites and records were employed in the suggested cluster-based design. The student information from each college is reflected in the compiled database. Let's assume that each college has |S| students available, and the data sources that were gathered are represented as

$$C = \{C_1, C_2, \ldots., C_j, \ldots, C_C\} \quad (1)$$

Where C_j denotes the data source gathered from the institution or college in question, and $|C|$ N represents the range in size of each unique data source. The term $|C|$ means the total number of pupils in the database, while N refers to all of the characteristics included in the dataset. The data repository contains personal information, academic records, extracurricular activity records, and schooling records. Since there are C data sources, N nodes are needed for the first merger layer, N nodes for the second, and Y nodes for the third in the suggested cluster-based distributed architecture.

$$B = C + N + N + Y \quad (2)$$

Where B stands for the suggested design's total number of cluster nodes.

3.1. Developing distributive layer-1

Layer 1 of the proposed prediction technique is a distributed layer in which missing data blocks are clustered and imputed. Due to the large amount of data included within the C data sources, the recommended design takes advantage of parallel processing when possible. The clustering procedure makes use of C processors in distributed layer one architecture.

3.1.1 Imputation of missing data

Data from information sources may need to be completed, so this issue must be explored adequately before conclusions regarding the student's performance can be drawn. The suggested cluster-based distributive system relies heavily on the MDI block, which requires the clustering-based neural network described in. In this scenario, we use the clustering model, which calculates the centroid by arithmetic mean, to cluster the data from various sources. The hybrid neural network receives the clustered data, and the WWO participates in NN weights training. The hybrid NN details the missing data from the data source. The MDI block transfers the data source and any missing attribute information to the following layer.

3.2. Developing the merging layer one

The output of the first dispersal layer is aggregated in the first merging layer. N mergers, given by the notation $N_n^1, N_n^2, \ldots, N_n^j, \ldots, N_n^C$, collect N clusters from each cluster in the first merger layer. Every set gives rise to a new cluster group, the responsibility for integrating which falls on the mth merger in the initial merger layer. In this case, the mth cluster group from the j^{th} clustering is denoted by N_n^j. $A_n = N_n^1, N_n^2, \ldots, N_n^j, \ldots, N_n^C$ (3) results from the mth merging. The information gained from each merger can be expressed as |U| N.

$$A_n = \{N_n^1, N_n^2, \ldots, N_n^j, \ldots, N_n^C\} \quad (3)$$

Where N_n^j represents the n^{th} cluster subset within the j^{th} clusters, each merged dataset has a size of |U| N.

3.3. Developing distributive layer second

The combined information is passed on from the first merging layer to the second distributed layer for feature selection and prediction. The distributed layer has N feature selector blocks and N predictor blocks. The suggested architecture uses the WWO-KMC-DNN algorithm for prophecy and the M-LDA model for feature selection.

3.3.1. Feature selection using M-LDA

The feature extraction method known as multi-linear discriminant analysis (M-LDA) is widely utilized during information processing. Controlled complexity reduction is accomplished through the use of M-LDA. The foundation of M-LDA is maximizing variation between categories while minimizing variation inside such a class. The same category values are as close as they can be after the data is shown in a near-bottom area, but the two independent data remain in distinct practical places. The graphical angle value of M-LDA can be calculated by doing a linear analysis on a diverging matrix containing the provided data. For a range of n values, the Equation yields a well-known and controllable LDA.

$$c^* = \underset{c}{argmax} \frac{c^P R_d c}{c^P R_z c}$$
(4)

$$R_d = \sum_{l-1}^{a} n_l \, (\mu^{(l)} - \mu)(\mu^{(l)} - \mu)^P$$
(5)

$$R_z = \sum_{l=1}^{a} \Big(\sum_{j=1}^{n_l} (y_j{}^{(l)} - \mu^{(l)}) \, (y_j{}^{(l)} - \mu^{(l)})^P \Big)$$
(6)

Where c represents all possible outcomes and is the average vector over all datasets, n_l represents the number of records in the lth category, l^{th} represents the mean vector of the l^{th} class, $y(l)$ represents the j^{th} sample in the lth class, R_z is the intra-class convergence matrix, and R_d is the inter-class divergent matrix. For a diverging matrix (R_t = R_d+ R_z) to be complete, its LDA evaluation value in Equation (8) must match its evaluation value in Equation (7).

$$c^* = \underset{c}{argmax} \frac{c^P R_d c}{c^P R_p c}$$
(7)

The previously extended matrix equation has a solution equal to (8), which can be considered an approximation.

$$R_d c = \lambda R_p$$
(8)

Therefore, we can change Eq. (8) to Eq. (9):

$$R_d = \sum_{l-1}^{a} n_l \, (\mu^{(l)} - \mu)(\mu^{(l)} - \mu)^P = \sum_{l-1}^{a} n_l \left(\frac{1}{n_l} \sum_{j=1}^{n_l} (y_j{}^{(l)} - \mu) \right) \left(\frac{1}{n_l} \sum_{j=1}^{n_l} (y_j{}^{(l)} - \mu) \right)^P = \sum_{l-1}^{a} \frac{1}{n_l} \Big(\sum_{j=1}^{n_l} (\bar{y}_j{}^{(l)} \sum_{j=1}^{n_l} (\bar{y}_j{}^{(l)})^P \Big) = \sum_{l=1}^{a} \bar{Y}^{(l)} \, Z^{(l)} (\bar{Y}^{(l)})^P = \bar{Y} Z \bar{Y}^P \quad (9)$$

$Z^{(l)}$is a$n_l * n_l$ is a square matrix of size n whose elements are all $1/n_l$. W is a matrix with the following dimensions in mm:

$$Z = \begin{bmatrix} Z^{(1)} & 0 & \dots & 0 \\ 0 & Z^{(2)} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & Z^{(a)} \end{bmatrix}$$
(10)

Where xi = xi represents a centralized data point, and $\bar{Y}^{(l)}$=$\overline{Y1}^{(l)}$,…., $\bar{Y}n_l{}^{(l)}$ represents the centralized data matrix of the l^{th} class. Moreover, $\bar{Y}^{(l)}$=$\overline{Y1}^{l}$,…., $\bar{Y}n_l{}^{l}$ represents the cth class's centralized data matrix. Given that R_d=$\bar{Y}Z\bar{Y}^P$, the following transformation applies to the extended value solution in Equation (11).

$$\bar{Y} Z \bar{Y}^P c = \lambda \overline{YY}^P$$
(11)

To more efficiently manage the eigenvalues challenge of M-LDA, we provide an M-LDA. Then, Equation (12) can be rewritten as if $\bar{Y}^P$ c=$\bar{x}$.

$$Z\bar{x} = \lambda \bar{x}$$
(12)

The eigenvalues in equations (12) and (13) are equivalent to those created and modified from $\bar{Y}^P$ c =$\bar{x}$. Using the observed data strategy, as shown in Equation (13), is another viable option for $\bar{Y}^P$ c =$\bar{x}$.

$$c = \underset{c}{argmin} \sum_{j=1}^{n} (c^P \bar{y_J} - \bar{x_J})^2$$
(13)

By solving (13), we can obtain the mappings matrix c= [$c1$, $c2$,:::, c^P]. Once features are extracted, they can be calculated using the formula c=$c^P\bar{y_J}$, which $\bar{y_J}$ produces a C_1 linear transformation.

3.3.2. Predicting Student Success with the suggested WWO-KMC-DNN

The M-LDA feature set feeds into the WWO-KMC-DNN's training process. M different WWO-KMC-DNNs are proposed for use in the distribution layer to forecast student performance. Each M-LDA's feature set is used to train the planned WWO-KMC-DNN network.

In this piece, we'll go through WWO fundamentals. The WWO method is a metaheuristic solution to global optimization problems. Each key has a height (h) and a wavelength (λ), and together, they form a "wave" on the ocean floor, as described by WWO. Using the distance from the seafloor to where the water is entirely motionless, we may determine the fitness of each wave. The WWO algorithm uses a population where each wave has a max of 0.5. With WWO, you may think of each iteration as a progression through these three operations propagation, refraction, and breaking—toward a global optimum. If we follow the logic of (1), we can create a new wave (Y) by adding the displacement along each axis (c).

$$Y, = Y + rand(-1,1) \times \lambda \times K_c$$
(14)

In this context, K_c is the search space size in the $c$ity dimension, where d is the duration of a unique function for producing random numbers within a specific range.

$$\lambda = \lambda \times \alpha^{\frac{-(e(Y)-emin+\varepsilon)}{(emax-emin+\varepsilon)}}$$
(15)

This is because deep-water waves have highly long wavelengths and low amplitudes. Shallow water waves have similarly modest wave heights and wavelengths. A wave's wavelength will decrease from deep to shallow water. Solving for Y in (15) yields each wave's wavelength (Y).

$$Y, = Gaussian(\mu, \sigma)$$
(16)

To prevent division by zero, we designate the fitness of a wave Y as f(Y), where fmax and fmin are the maximum and minimum fitness values in the current population, is the parameter for the wavelength reduction coefficient, and is a minor constant. Fewer, more powerful waves can travel more irregular distances with this aid. The refraction operator is used to achieve a vertical wave height of zero. The resulting wave (Y') is based on a Gaussian function, a statistical distribution with a known mean and standard deviation.

Mean (μ) is defined in Eq. 3, whereas standard deviation (σ) is calculated using Eqs. (17) and (18).

$$\mu = \frac{Ybestd+Yc}{2}$$
(17)

$$\sigma = \frac{Ybestd+Yc}{2}$$
(18)

The average (μ) is calculated using the current wave (Y) and the optimal wave ($Ybestd$). The difference between the best wave ($Ybestd$) and the recent wave (Y) is the standard deviation (σ). Additionally, the wavelength is determined by (19), and the wave height is reestablished to the max.

$$\lambda' = \frac{e(Y)}{e(Y,)}$$
(19)

Where is the frequency of the initial wave, Y' is the frequency of the current wave, and f(Y) is the fitness of both the recent surge and the previous wave. To eliminate the wave (Y), the splitting operator (e) must eventually outperform the best-known solution at the time ($Ybestd$). To calculate the solitary wave (Y'), we use (20)

$$Y, = Y + Gaussian(0,1) \times \beta \times Kc$$
(20)

The random integer between 0 and 1 is generated using the 0–1 Gaussian function, denoting the rupture frequency. If wave Y ' is superior to wave y, y' will take its place.

K-Means clustering is based on a straightforward idea. The dataset $\bar{y}_j$ (j = 1, 2, 3,...) has m observations. Cluster centers and the L value are known. The K-Means clustering algorithm uses the sum of squared errors (SSE) as its goal function.

$$TTF = \sum_{j=1}^{L} \sum_{y \in D_j} |y - \bar{y}_j|^2$$
(21)

One cluster result class is denoted by D, the number of categories by L, and the mean of a cluster, $\bar{y}_j$. In the minimal value of the goal function, the clustering effect is at its best. The K-Means clustering algorithm can be broken down into three distinct phases: First, we scale back the objective function by assigning samples to the vector centers closest to them.

$$\sum_{j=1}^{m} \underset{i \in |j = 1,2,3 \ldots l|}{min} ||y_j - o_i||^2$$
(22)

Euclidean distance is used in the following formula for finding the distance between two points:

$$c(y_j, y_j) = \sqrt{\sum_{l=1}^{c} (y_{jl} - y_{il})^2}$$
(23)

Centers of K-clusters are represented by o. The value of $the\ y_j$ attribute is denoted by c.

step 2: Modify the cluster-wide mean

$$\bar{y}_j = \frac{1}{|D_j|} \sum_{y \in D_j} Y$$
(24)

Step 3: Determine the value of the objective. The cluster effect performs best when the objective function's value is minimized.

The result is a neural network when the perceptron model is repeated and superimposed. A DNN can be thought of as an extensive network structure that is comprised of neurons that have nonlinear units attached to them. As shown in Fig. 2, the DNN is superior to the NN in obtaining the complex multivariate multi-function model and solving issues with multiple variables.

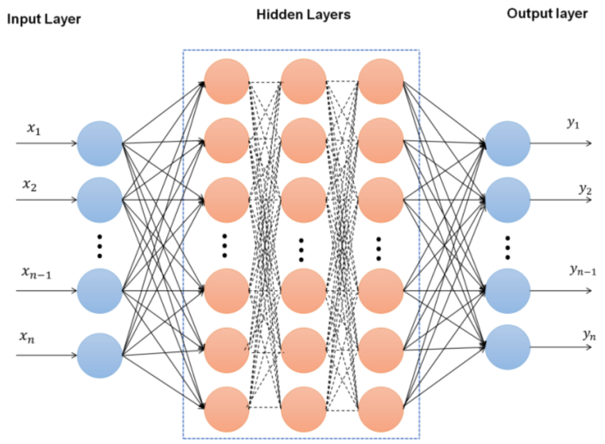

Fig.2. Structure of DNN

This neural network is multi-layered, with each layer receiving input characteristics from the layer above it based on the layer above its output characteristics. The multi-layer nonlinear transformation is used to train the original parts, which is then used to evaluate the features of the current input. The features of the first sample are then gradually changed into a different feature space, depending on the application's needs. The DNN can train on big datasets to increase their information value and generate multi-level mathematical models, allowing it to sift through massive amounts of data more precisely and efficiently.

The DNN is segmented by where the various network layers are located. The top layer receives data, the bottom layer displays it, and the layers in between are kept secret. The entire connectivity of any neuron in any connected layer is shown in Fig. 2. During feed-forward propagation, the input vector is transformed in a series of linear and activation operations using the weighted coefficient matrices, the B-biased wand. After running the model, the results are collected. To complete the DNN model training, however, back-propagation is employed. To put it another way, the loss function puts a number on the gap between the forecasted and actual outcomes. The bias B and weight coefficient W are updated continuously throughout the training process.

$$I(V,a,y,x) = \frac{1}{2}\|x^K - x\|_2^2 = \frac{1}{2}\sigma(V^K x^{K-1} + a^K) - x\|_2^2$$
(25)

The buried layer's output is denoted by x^K.

3.4. Second Step in Building the Merge Layer

In the suggested design, the second layer of mergers predicts how well students will do in future semesters based on their current performance. P is a $Y1$-sized result from the second layer of merging. Each merge in the second tier of the merger tree offers the semester one grade for students denoted by the Y.

Therefore, the Y merger in the second layer will reflect the Y students' semester grades. The following Equation can describe the outputs of Merger Layer 2 Y merges.

$$P = \{P_1, P_2, \dots, P_P, \dots., P_Y\}$$
(26)

Where P_P describes the outcomes of the P^{th}-level merger, the WWO-KMC-DNN's m-th layer's outputs are combined at the o-th layer's dispersed layer two mergings. Since Y students are represented in database C, their grades from the first semester are stored in the first layer of the merger, labeled 1. To specify the output of the second merging layer, we use the phrase,

$$P_P = [U_d(1), U_d(2), \dots, U_d(n), \dots, U_d(M)\}$$
(27)

$U_d(M)$ is the size of the output layer in the d^{th} WWO-KMC-DNN network. Finally, the final merge delivers all Y students' (t + 1)th semester grades. The suggested CDA for forecasting student performance is presented in the pseudo-code form in Algorithm 1.

Algorithm: pseudo code of CDA algorithm

Input: Data source $=\{C_1, \dots.. C_j, \dots, C_C\}$

Output: Prediction $=\{P_1, P_2, \dots.., P_p, \dots., P_W\}$

Begin

For each data source, j=1 to C

Layer 1 Spread

Compute the Missing data from NCJ

Cluster the data using the AED

For (j=1 to C)

Cluster the data sources into N clusters

Output of AED N_j=$\{N_1^j, N_2^j, \dots \dots N_n^j, \dots.. N_N^j\}$

End for

Merger layer 1

For each cluster (n=1 to N)

Merge the data A_n $=\{N_n^1, N_n^2, \dots \dots N_n^j, \dots.. N_n^C\}$

End for

Distributed layer 2

For each merger (n=1 to N)

Find the features E_n $=\{e_n^1, e_n^2, \dots \dots e_n^q, \dots.. e_n^Q\}$

Provide the features to LW-DBN for training

Find the output of the LW-DBN based on the Equation

End for

Merger layer 2

For each output layer of LW-DBN (d=1 to W)

Merge the data

$P_p = \{Y_d(1), Y_d(2), \dots \dots Y_d((n), \dots \dots Y_d(N)\}$

Find the predicted output P=$\{P_1, P_2, \dots \dots, P_p, \dots, P_W\}$ for all W

End for

Return the predicted output$P$=$\{P_1, P_2, \dots \dots, P_p, \dots, P_W\}$

End

4. Result and Discussion

In this study, the proposed method its efficacy is compared to that of previously used approaches such as deep artificial neural networks (deep ANN), deep neural networks (DNN), naive Bayes [NB], generative adversarial network-based support vector machine [GAN-SVM]. Accuracy, prediction rate, MSE, and RMSE were important metrics studied using proposed and current approaches.

The accuracy of a statement can be determined by dividing the number of words by the corresponding number of accurate classifications. The system's accuracy depends on the classifier's ability to categorize students' results correctly. In mathematics, precision means,

$$Accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)} \quad (28)$$

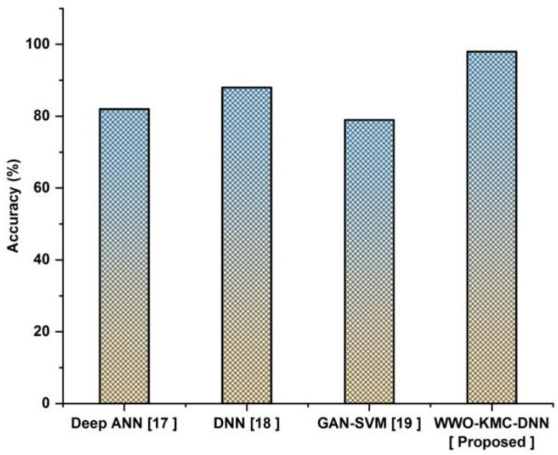

Fig.3. Comparison of the accuracy

Fig.3 displays the results of the comparison of accuracy. Compared to established approaches like deepANN, DNN, NB, and GAN-SVM, the suggested method WWO-KMC-DNN has higher significance accuracy.

One of the most often used methods for forecasting exchange rates, students' performance compares the prices of goods in different countries. The relative economic strength method analyzes student growth rates across countries to forecast exchange rates. In statistics and diagnostics, a test's positive and negative predictive values indicate the proportion of correct conclusions.

$$PPR = \frac{T_{positive}}{T_{positive} + T_{Negative}} \quad (29)$$

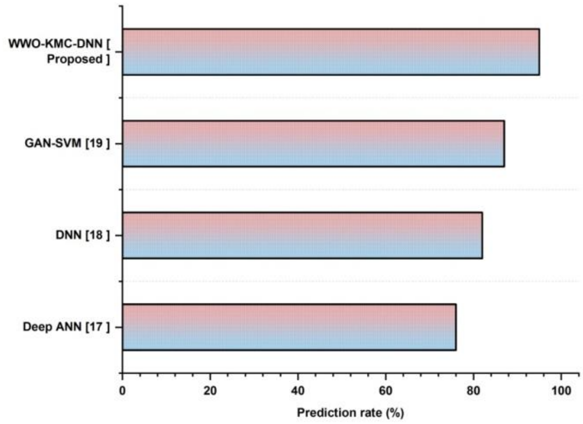

Fig.4. Comparison of the prediction rate

Prediction rate is compared in Fig.4. Therefore, WWO-KMC-DNN has advantages over preexisting approaches such as deep ANN, DNN, and GAN-SVM.

The root-mean-square error (RMSE) is calculated by squaring the most significant network measurement point discrepancy between a dataset's state vectors and coordinate values from a highly independent source. Although the totals are tallied differently, there is a substantial correlation between the discrepancy in predicted and observed student reviews for each connection and the objective measure of accessibility.

$$\text{RMSE} = \sqrt{\sum_{l}^{H} \frac{[count_l - model_l]^2}{N}} \quad (30)$$

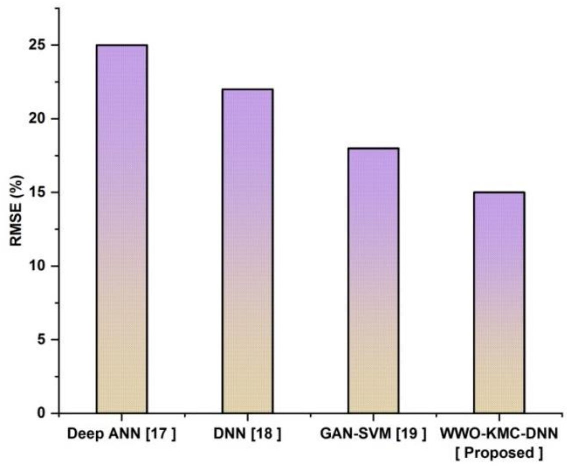

Fig.5.RMSE of the proposed and existing method

The RMSE contrast is shown in Fig.5. Student performance review prediction efficiency is measured by the root-mean-squared error (RMSE), which is the average difference between a variable's actual and predicted values. The root mean square error of the proposed method WWO-KMC-DNN is smaller than that of state-of-the-art approaches like deepANN, DNN, NB, and GAN-SVM.

The MSE of a method for a given test data set is the mean of the individual PEs for each instance in the validation set. Generally speaking, higher prediction quality is associated with lower metrics values. To finish the efficiency evaluation metrics, it may be essential to examine the forecast inaccuracy visually.

$$MSE = \frac{\sum_{k=1}^{l} |h_i|}{N}$$
(31)

Where

$|h_i|$ = total of all errors.

N = the level of the prediction errors.

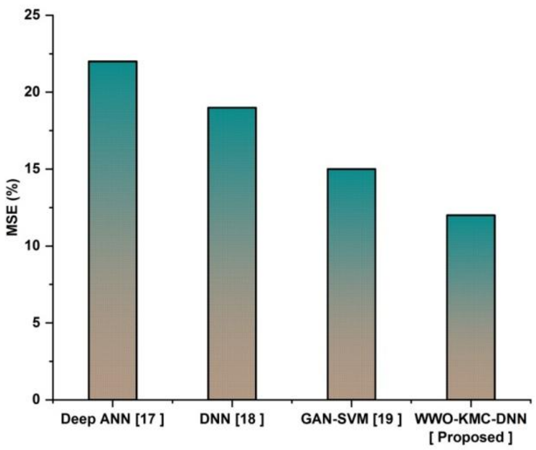

Fig.6.MSE of the proposed and existing methodology

Average errors are displayed in Fig.6. The proposed WWO-KMC-DNN has low MSE compared to state-of-the-art algorithms like deep ANN, DNN, and GAN-SVM.

5. Conclusion

This study's main contribution to the EDM is the creation of the CDA, a tool for predicting academic performance in the future. The proposed distributed architecture is based on clusters, and it uses the kids' prior versions in school to make predictions about their future performance. The suggested architecture uses two merging levels and two distributive layers to carry out operations like clustering, feature extraction, and forecast. Features for the architecture are picked using the M-LDA model from among the clusters. During training, the parts are sent into the suggested WWO-KMC-DNN algorithm, where the WWO algorithm selects the most appropriate weights for the prediction. We tested the proposed work in the lab by playing around with the parameters of the KMC-DNN network's training, dropout rate, and step-out rate. We evaluate the suggested WWO-KMC-DNN's simulation outcomes in terms of accuracy, prediction rate, root-mean-squared error, and mean-squared error, and we contrast them with those from prior works. The results were contrasted with several baseline techniques to determine performance levels. It was advised to use WWO-KMC-DNN, and the results showed 98% accuracy, 95% prediction rate, 12% of RMSE, and 15% MSE. Utilizing cutting-edge clustering and optimization techniques will be a future improvement of this work's performance.

References.

[1] Tight M. Student retention and engagement in higher education. Journal of further and Higher Education. 2020 May 27;44(5):689-704.

[2] Korpershoek H, Canrinus ET, Fokkens-Bruinsma M, de Boer H. The relationships between school belonging and students' motivational, social-emotional, behavioural, and academic outcomes in secondary education: A meta-analytic review. Research papers in education. 2020 Nov 1;35(6):641-80.

[3] Fauth B, Decristan J, Decker AT, Büttner G, Hardy I, Klieme E, Kunter M. The effects of teacher competence on student outcomes in elementary science education: The mediating role of teaching quality. Teaching and Teacher Education. 2019 Nov 1;86:102882.

[4] Dunn TJ, Kennedy M. Technology Enhanced Learning in higher education; motivations, engagement and academic achievement. Computers & Education. 2019 Aug 1;137:104-13.

[5] Almasri A, Celebi E, Alkhawaldeh RS. EMT: Ensemble meta-based tree model for predicting student performance. Scientific Programming. 2019 Feb 24;2019.

[6] Nieto Y, Gacía-Díaz V, Montenegro C, González CC, Crespo RG. Usage of machine learning for strategic decision making at higher educational institutions. IEEE Access. 2019 May 27;7:75007-17.

[7] Pascoe MC, Hetrick SE, Parker AG. The impact of stress on students in secondary school and higher education. International Journal of Adolescence and Youth. 2020 Dec 31;25(1):104-12.

[8] Alhadabi A, Karpinski AC. Grit, self-efficacy,

achievement orientation goals, and academic performance in University students. International Journal of Adolescence and Youth. 2020 Dec 31;25(1):519-35.

[9] Gangula, R. ., Vutukuru, M. M. ., & Kumar M., R. . (2023). Network Intrusion Detection Method Using Stacked BILSTM Elastic Regression Classifier with Aquila Optimizer Algorithm for Internet of Things (IoT). International Journal on Recent and Innovation Trends in Computing and Communication, 11(2s), 118–131. https://doi.org/10.17762/ijritcc.v11i2s.6035

[10] Liu Q, Huang Z, Yin Y, Chen E, Xiong H, Su Y, Hu G. Ekt: Exercise-aware knowledge tracing for student performance prediction. IEEE Transactions on Knowledge and Data Engineering. 2019 Jun 24;33(1):100-15.

[11] Tang YM, Chen PC, Law KM, Wu CH, Lau YY, Guan J, He D, Ho GT. Comparative analysis of Student's live online learning readiness during the coronavirus (COVID-19) pandemic in the higher education sector. Computers & education. 2021 Jul 1;168:104211.

[12] Saloviita T, Pakarinen E. Teacher burnout explained: Teacher-, student-, and organisation-level variables. Teaching and Teacher Education. 2021 Jan 1;97:103221.

[13] Baneres D, Rodríguez-Gonzalez ME, Serra M. An early feedback prediction system for learners at-risk within a first-year higher education course. IEEE Transactions on Learning Technologies. 2019 Apr 23;12(2):249-63.

[14] Tampakas V, Livieris IE, Pintelas E, Karacapilidis N, Pintelas P. Prediction of students' graduation time using a two-level classification algorithm. InTechnology and Innovation in Learning, Teaching and Education: First International Conference, TECH-EDU 2018, Thessaloniki, Greece, June 20–22, 2018, Revised Selected Papers 1 2019 (pp. 553-565). Springer International Publishing.

[15] Almaiah MA, Alamri MM, Al-Rahmi W. Applying the UTAUT model to explain the students' acceptance of mobile learning system in higher education. IEEE Access. 2019 Dec 2;7:174673-86.

[16] Adekitan AI, Noma-Osaghae E. Data mining approach to predicting the performance of first year student in a university using the admission requirements. Education and Information Technologies. 2019 Mar 16;24:1527-43.

[17] Awidi IT, Paynter M, Vujosevic T. Facebook group in the learning design of a higher education course: An analysis of factors influencing positive learning experience for students. Computers & Education. 2019 Feb 1;129:106-21.

[18] Waheed H, Hassan SU, Aljohani NR, Hardman J, Alelyani S, Nawaz R. Predicting academic performance of students from VLE big data using deep learning models. Computers in Human behavior. 2020, 104:106189.

[19] Pande, S. D., Kanna, R. K., & Qureshi, I. (2022). Natural Language Processing Based on Name Entity With N-Gram Classifier Machine Learning Process Through GE-Based Hidden Markov Model. Machine Learning Applications in Engineering Education and Management, 2(1), 30–39. Retrieved from http://yashikajournals.com/index.php/mlaeem/article/view/22

[20] Li S, Liu T. Performance prediction for higher education students using deep learning. Complexity. 2021, 2021:1-0.

[21] Chui KT, Liu RW, Zhao M, De Pablos PO. Predicting students' performance with school and family tutoring using generative adversarial network-based deep support vector machine. IEEE Access. 2020 May 6;8:86745-52.

[22] Vijayalakshmi V, Venkatachalapathy K. Comparison of predicting student's performance using machine learning algorithms. International Journal of Intelligent Systems and Applications. 2019, 11(12):34.