A Data-Driven Digital Twin for Student Engagement Prediction in e-Learning Systems

Sandra Kumi[a], Richard K.Lomotey[b],Madhurima Ray[b], Emma Cunningham[b], Stephanie Milovich[b], and Ralph Deters[a]
[a]Department of Computer Science, University of Saskatchewan, SK, Canada
[b]Computer Science, College of Engineering, The Pennsylvania State University, PA, USA
sandra.kumi@usask.ca, rkl5137@psu.edu, mvr6106@psu.edu, erc5617@psu.edu, smm8050@psu.edu, and deters@cs.usask.ca

2025 IEEE World AI IoT Congress (AIIoT) | 979-8-3315-2508-8/25/$31.00 ©2025 IEEE | DOI: 10.1109/AIIoT65859.2025.11105257

***Abstract*— Machine Learning (ML) models are increasingly applied to Learning Management System (LMS) data to predict student engagement and performance. LMS data often contain missing values that can be informative. However, existing modeling approaches in education remove or impute missing values, which can lead to inaccurate or biased models. In this paper, we propose the use of digital twins to model students' engagement based on their learning activities on LMS while preserving the missingness patterns. We leveraged synthetic data generators such as Conditional Tabular Generative Adversarial Network (CTGAN), Tabular Variational Autoencoder (TVAE), and RealTabFormer with reversible data transformations to create a virtual replica of students' data. The CTGAN and TVAE generated balanced synthetic data that accurately captured the meaningful patterns of the real data. Moreover, XGBoost trained on a balanced virtual replica of the students' learning activities data obtained an F1-score of above 80% in predicting the students' engagement levels when evaluated on real data with both complete and incomplete entries. Our findings demonstrate how digital twins can be used to address the complexities of data in the education sector, improve the generalization of models, and reduce bias in real-world performance.**

Keywords—Digital Twin, Education, Learning Activities, Student Engagement Modeling, Learning Management Systems, MOOC, Synthetic Data Generative Models

I. INTRODUCTION

In the era of Education 4.0, innovative technologies and methodologies such as Machine Learning (ML) are being utilized to enhance the quality of education [1]. ML models are often used to analyze Learning Management System (LMS) data to uncover insights for informed decision-making and early interventions on students' achievements [2][3]. However, LMS data often contain missing values due to students not interacting with the system in a way that generates data [4] or device failures [5]. Moreover, the majority of LMS data used in modeling students' behavior is highly imbalanced [2][3]. To handle imbalanced datasets, statistical methods such as the Synthetic Minority Oversampling Technique (SMOTE) are often used. However, many statistical and ML approaches require complete datasets [5]. Missing data is poorly handled in studies utilizing ML for prediction and analysis [6]. Ignoring missingness that depends on observed or unobserved data in predictive modeling can lead to bias and loss of analytical power in prediction when the model is deployed [6][7]. While tree-based models like Extreme Gradient Boosting (XGBoost) can handle missing values, training on imbalanced and less diverse datasets may still lead to inaccurate and biased predictions [8].

In recent research works, data-driven digital twins have been leveraged to address data challenges such as limited data availability, imbalanced data, and so on in areas such as healthcare [9] and smart factories [10]. A digital twin (DT) is a virtual replica of a physical entity that mirrors the state and simulates various scenarios of its physical counterpart to provide insights [11]. Most data-driven DTs employ synthetic data generative models (SDGs) to address the data challenges [9]. However, these SDGs also require a complete dataset before analysis [12].

In this study, we posit that incorporating missingness in the modeling of students' behavior on LMS can improve the predictive power of ML models and mitigate bias. We propose the use of digital twins to model students' engagement based on their learning activities on LMS while retaining the missing values. Specifically, we employed reversible data transformations with synthetic data generative models (SDGs) such as Conditional Tabular Generative Adversarial Network (CTGAN), Tabular Variational Autoencoder (TVAE), and RealTabFormer to create virtual replicas of students' learning activities.

Our experimental analysis shows that CTGAN and TVAE can generate high-quality, balanced synthetic datasets that accurately capture the missing patterns and pairwise correlations of the real data. Additionally, we trained XGBoost on the virtual replicas of the students' learning activities data and tested it on real data containing both complete and incomplete entries. It achieved a score of above 80% in the prediction of students' engagement levels across all classification evaluation metrics used in this study. The proposed work demonstrates the potential of using digital twins to address data complexities in LMS data.

The remainder of the paper is organized as follows. Section II discusses the related work on modeling students' engagement on LMS with ML and the adoption of DT in education. Section III describes the methodology of our proposed work. We present the implementation and evaluation of our proposed approach in Sections IV and V. Section VI outlines our conclusions and future research directions.

979-8-3315-2508-8/25/$31.00 ©2025 IEEE

II. Related Work

This section highlights the related work on using Machine Learning (ML) to detect students' engagement on learning management systems (LMS) and the adoption of digital twins (DTs) in education.

Hussain et al. [13] proposed the use of ML algorithms to detect low-engaged students in a social science course to assess the effect of engagement on student performance. The input variables used in training the ML models include the highest education level, final grade, assessment scores, and the number of clicks on activities performed in the virtual learning environment. Additionally, the authors developed a dashboard based on the ML models to facilitate real-time monitoring of students' engagement during learning. Similarly, Alruwias and Zakariah [5] compared the performance of four ML models in the detection of student engagement in a virtual learning environment. Data preprocessing techniques, such as the removal of missing values, normalization, encoding, and outlier identification, were performed before training the models. The results of the study indicated that CATBoost was the outstanding model with a precision of 94.40%. Moreover, Orji et al. [2] leveraged statistical analysis and ML on a de-identified Canvas Network dataset to predict student engagement based on their learning behaviors. The analysis of the study revealed that the students' self-reported learner types did not represent their learning behaviors recorded on LMS. Sharma et al. [14] employed the Haar-cascade algorithm and Convolutional Neural Network (CNN) to analyze the facial emotions, eye, and head movements of students, as captured by a web-camera in an e-learning environment, to detect their engagement level. The results of the proposed work, tested on fifteen students, show that students with a higher concentration index obtained the best scores in quizzes.

The use of digital twins in education is still in its early stages. According to a literature review by Bachmann et al. [15], most research on using DT in education focuses on the virtualization of campus infrastructures, improving teaching and learning processes, and optimizing costs for equipment and laboratories. Sepasgozar [16] developed a DT and web-based virtual gaming for the delivery of construction lessons. Five mixed-reality modules were developed to enhance students' engagement in construction through online immersive teaching experience. Razzaq et al. [17] proposed a deep learning-based digital twin framework for attendance and course content monitoring for schools. The course contents monitoring module is conducted through a Convolutional Neural Network (CNN), trained with 19,320 unique images to extract text for content matching. The authors proposed that attendance will be monitored through facial recognition and edge devices such as RFID.

Similarly, Leotlela and Coetzee [18] proposed a proof of concept for using digital twins to monitor classes and the performance of students. The proposed approach mainly utilized data analysis and visualization techniques on real-time data to provide personalized feedback on student performance. Further, Huang and Willcox [19] proposed an educational DT (EDT) to understand, model, and analyze educational big data to address how students' behavior after their pathways. The proposed EDT leverages a knowledge graph to organize data to derive insights for decision-making.

Existing traditional methods of detecting students' engagement based on their learning activities on LMS rely on complete data to train ML models [2], [5], [13]. However, students engage with LMS platforms in different ways, which introduces missing data in the logs [4]. Ignoring missing data or not using the right imputation methods to handle missing values can lead to sampling bias if the missingness is informative [12]. In this study, we focus on using synthetic data generators (SDGs) with reversible data transformations to create digital replicas of students' learning activities and training ML models to predict engagement levels while maintaining missingness. The proposed approach will ensure ML models capable of handling missing data, such as XGBoost, to make accurate predictions of students' engagement levels where missing data is unavoidable.

III. Methodology

Fig.1. illustrates the process flow of using a digital twin (DT) to predict the engagement levels of students on Learning Management Systems (LMS). The Canvas Network open courses dataset [4] is used as a data source to create a DT of students' learning activities. We trained Extreme Gradient Boosting (XGBoost) on the virtual replica of the students' learning activities data to predict the engagement levels. The Shapley Additive exPlanations (SHAP) [20] EXplainable Artificial Intelligence (XAI) tool is leveraged to explain the prediction outputs of XGBoost. The details of the proposed work are discussed as follows.

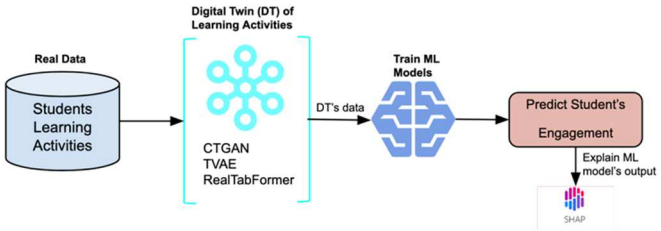


Fig.1. Process Flow of using DT to predict Engagement Levels of Students based on their Learning Activities.

A. Data Collection

We used the de-identified data from Canvas Network open courses [4], which ran from January 2014 to September 2015. The dataset includes over 325,000 records, with each record representing an individual's activity in one of the 238 Canvas Network courses. The dataset has 26 columns and contains information about courses, student learning activities, grades, demographics, and surveys from students about courses. As the focus of this study is the modeling of student engagement based on their learning activities, we extracted only the learning activities' features generated by the system. A description of the five features extracted with their percentage of missing values is shown in Table I. We focus on students who

participated in at least one activity in a course and have a percentage of completion provided by the system. A total of 26,661 records were retained after data cleaning.

In this study, we model the prediction of a student engagement level as a classification task, where the features *nevents, ndays_act, ncontent*, and *nforum_posts* are used as the input features to predict the target feature *completed_%.* We group the *completed_%* into two classes (0,1) of *Engagement_Level,* where *0* represents low-engaged students with a *completed_%* lesser than 50% and *1* represents highly engaged students with a *completed_%* of greater than or equal to 50%. From Table I, *nforum_posts* has a high percentage of missing values, which could be linked to students' lack of participation in forums. The description of the dataset states that the blank values are a result of the student not interacting with the system in a way to generate data, or the course was not designed to produce data [4]. Fig. 2 shows the correlation of missingness in the data. It can be observed that missing *nforum_posts* has a stronger negative correlation with the *Engagement_Level,* indicating a lower engagement. Overall, missingness is informative in this use case. Hence, we retained missing values in the modeling of student engagement.

TABLE I. Description of dataset and proportion of missing values.

Features	Description	Percentage of Missing Values (%)
nevents	Count of distinct interactions with a course	17.08
ndays_act	Count of distinct days with one or more events	10.44
ncontent	Percentage of unique modules viewed in a course	0
nforum_posts	Number of posts in the discussion forums throughout the course	68.05
completed_% (Engagement Level): Target	Percentage of total required content modules completed	

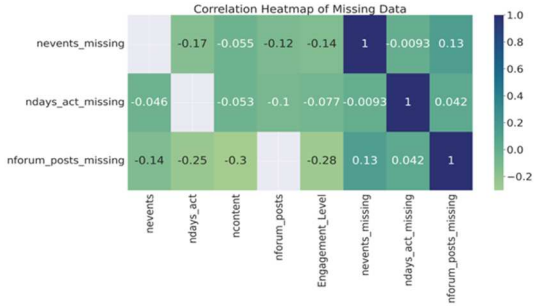


Fig.2. Pearson correlation heatmap of missingness in the cleaned dataset.

B. Digital Twin of Students' Learning Activities

1) Synthetic Data Generation Models

We utilized Synthetic Data Generators (SDGs) to create a virtual replica of the students' learning activities. The performance of three SDGs, namely, Conditional Tabular Generative Adversarial Network (CTGAN) [21], Tabular Variational Autoencoder (TVAE) [21], and Realistic Relational and Tabular Transformer (RealTabFormer) [22] were evaluated on how accurately they can mimic the real learning activities of students. The CTGAN leverages mode-specific normalization, a conditional generator, and training-by-sampling to create synthetic data [23]. The TVAE adapts the Variational Autoencoder (VAE) neural network generative model preprocessing techniques and uses evidence lower-bound (ELBO) loss in training to generate synthetic tabular data. RealTabFormer is a Large Language Model (LLM)-based SDG model that uses sequence-to-sequence (Seq2Seq) and autoregressive GPT-2 model to generate synthetic data for relational and non-relational tabular datasets, respectively.

2) Modeling of Student Learning Activities

The initial implementation of the SDGs used in this study requires complete data for training. We integrate reversible data transformations with each SDG to allow the generation of synthetic data with missing values. The workflow of generating synthetic data to maintain the missing values distribution is shown in Fig.3. Algorithm 1 illustrates the details of generating synthetic data from real data with missing values.

We first transform the real data by learning the missing values patterns, and then impute the missing values as indicated in Steps 1 and 2 of Algorithm 1. We leveraged two approaches in learning the missing values patterns, the first is using missingness indicators (*from_column*), and the second is the proportion of missing values (*random*). To learn the missing values patterns using missingness indicators, a binary column is created to indicate which entries are missing for each column with missing values in the real data. An entry with a missing value is assigned 1, otherwise, 0 is assigned. The missing indicator columns are treated as categorical data types instead of integers. The second learning approach estimates the proportion of missing values in each column with missing values in the real data.

Secondly, complete data is used in training the SDGs to generate synthetic data. As indicated in Step 3 of Algorithm 1, if the missing values pattern learning strategy is missingness indicators (*from_column*), missingness indicator columns are appended to the imputed data and used for training the SDGs. If the missing values pattern learning strategy is based on the proportion of missing values (*random*), only the imputed data is used for training.

Lastly, as indicated in Step 4 of Algorithm 1, missing values are reintroduced into the synthetic data based on the learned real data's missing patterns. If the reverse strategy is *from_column* ($M_{reverse} = from\_column$), the generated missingness indicator column in the synthetic data is used to determine where to apply missing values. If the reverse strategy is *random* ($M_{reverse} = random$), missing values are recreated randomly in approximately the same proportion as the real data. For each feature with missing values, random values between 0 and 1 are generated for each entry in the synthetic data. If a randomly generated value is smaller than the proportion of missing values for that feature, the corresponding entry is replaced with *NaN*.

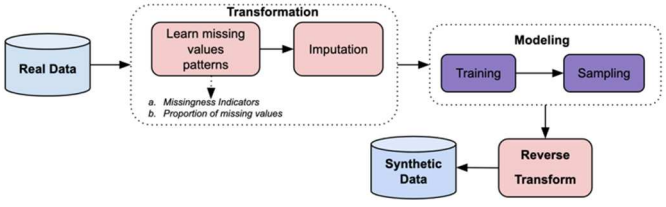


Fig.3. Workflow of Synthetic Data Generation.

Algorithm 1: Generating Synthetic Data with Missing Values

Input: $D_{real} = \{\vec{d}_1, \vec{d}_2, \ldots, \vec{d}_n\}$ // **Real data with missing values**
Missing values learning strategy: $M_{learn} \in \{random, from\_column\}$
Missing values reverse strategy: $M_{reverse} \in \{random, from\_column\}$
SDG: Synthetic Data Generation Model
Output: $D_{syn} = \{\vec{d}_1, \vec{d}_2, \ldots, \vec{d}_n\}$ // **Synthetic data with missing values**
1: $M_j = \{i \mid D_{real}[i,j] = NaN\}$ //**track missing value indices for each feature** f_j.
2: $D_{real}^{imputed} = impute(D_{real})$ //**Impute missing values**.
3: $D_{syn} = Train_{SDG}(D_{real}^{imputed})$ // **Train SDG models**
  if $M_{learn} = from\_column$:
    for each f_j create missingness indicator column, $f_j^{missing}$ and append to $D_{real}^{imputed}$:
      $f_j^{missing}[i] = \begin{cases} 1, & if\ D_{real}[i,j]\ is\ missing \\ 0, & otherwise \end{cases}$
      $Train_{SDG}\left(D_{real}^{imputed}, f_j^{missing}[i]\right)$ //**include missing indicators in training**
  if $M_{learn} = random$:
    $Train_{SDG}(D_{real}^{imputed})$ //**train models without missing indicators**
4: Return D_{syn} //**Recreate missing values in synthetic data**
  if $M_{reverse} = from\_column$:
    $D_{syn}[i,j] = \begin{cases} NaN, & if\ f_j^{missing}[i] = 1 \\ D_{syn}[i,j], & otherwise \end{cases}$ // **Reverse missingness using missing indicators.**
  if $M_{reverse} = random$:
    $D_{syn}[i,j] = \begin{cases} NaN, & if\ random\ value < f_j^{missing}\% \\ D_{syn}[i,j], & otherwise \end{cases}$ // **Randomly assign missing values.**

C. Students Engagement Level Prediction

Machine Learning (ML) models are trained on the generated synthetic data to predict the engagement level of students. In this study, we leveraged the Extreme Gradient Boosting (XGBoost) [24] for prediction due to its ability to handle missing values in training data without imputation. XGBoost uses a gradient boosting framework based on decision tree learning algorithms. It leverages level-wise growth for faster training. The XGBoost uses a sparsity -aware split finding algorithm to learn the optimal branch direction for missing values during training [24]. The Shapley Additive exPlanations (SHAP) [20] EXplainable Artificial Intelligence (XAI) tool is used to explain the prediction outputs of XGBoost.

IV. Experiments

All experiments in this study were implemented in Python and ran on Google Colab notebooks.

A. Modeling of Students' Learning Activities Digital Twin

We used 80% of the cleaned dataset, i.e., 21,328 records, as training data for the SDGs to create a digital twin of students' learning activities. We performed Little's test for missing completely at random (MCAR) [25] to confirm the type of missingness. The test resulted in a *p-value* of 0, which implies that the training dataset is either missing at random (MAR) or missing not at random (MNAR). We used the Reversible Data Transforms (RDT)[1] Python package to learn the missing values pattern in the real data and to recreate the missing values after the generation of synthetic data. We applied two reverse strategies from RDT: *random* and *from_column*. The *random* approach learns the missing patterns by calculating the proportion of missing values for each column with missingness in the real data. In the *random* strategy, missing values are generated in the synthetic data by randomly assigning missing values from the learned distribution of missing values in the real data. In the case of the *from_column* strategy, missingness indicators are used to learn and reverse the missingness patterns. The missingness indicators are new columns created to store whether an entry in the real data should be missing. An entry with a missing value is assigned 1, otherwise, 0 is assigned.

We used the K-nearest Neighbors (KNN) imputation to create a complete dataset for all the SDGs. In the *random* strategy, only the imputed data is used in training the SDGs. In the from_*column* strategy, the SDGs are trained on a combination of the imputed data and the missingness indicators columns.

The CTGAN and TVAE models use the Synthetic Data Vault's (SDV) [2] Python package for training. The RealTabFormer uses the *realtabformer*[3] library for training. For both transformation strategies, CTGAN and TVAE were trained with 300 epochs and RealTabFormer was trained with 10 epochs and maximum steps of 6600.

The training graphs of CTGAN, TVAE, and RealTabFormer when using the *random* reversible data transformation strategy is shown in Fig.4. The CTGAN obtained a generator loss of approximately -1.05 and discriminator loss of -0.08 after training. The TVAE and RealTabFormer finished with a loss of -14.05 and 0.45 respectively. Fig.5 illustrates the training graphs of the SDGs when using the *from_column* (missingness indicator) reversible data transformation strategy. The CTGAN finished with a generator loss of approximately -1.22 and discriminator loss of -0.12. A negative value for generator loss indicates the model is generating a better synthetic data while a discriminator loss with value closer to 0 implies the discriminator is unable to differentiate between the real and synthetic data. In the case of TVAE and RealTabFormer, a lower loss value means the model is improving in the generation of synthetic data.

B. Training of XGBoost for Student Engagement Level Prediction

We employed the Python implementation of XGBoost for prediction. We sampled synthetic dataset of 21,328 records from each trained SDG to train the predictive model with its default parameters. The remaining 20% of the real data, i.e., 5333, was used to evaluate the performance of XGBoost in predicting the student engagement levels. The *TreeExplainer* from the SHAP framework was utilized to explain the predictions of XGBoost on the test data.

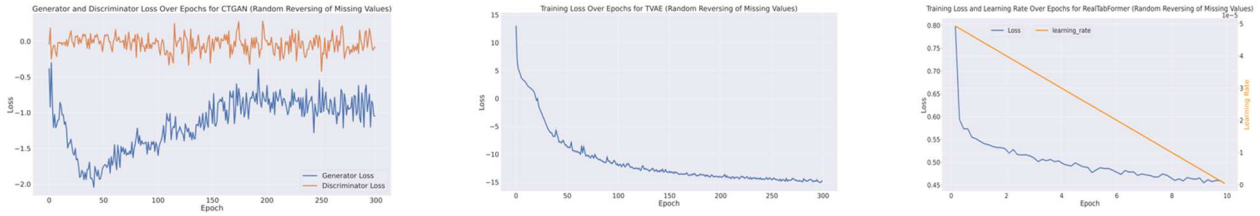
C. Evaluation Metrics

1) Evaluation Metrics for SDGs.

We used the Synthetic Data Metric [4] (SDMetrics) to evaluate the quality of the synthetic data generated by each generative model.

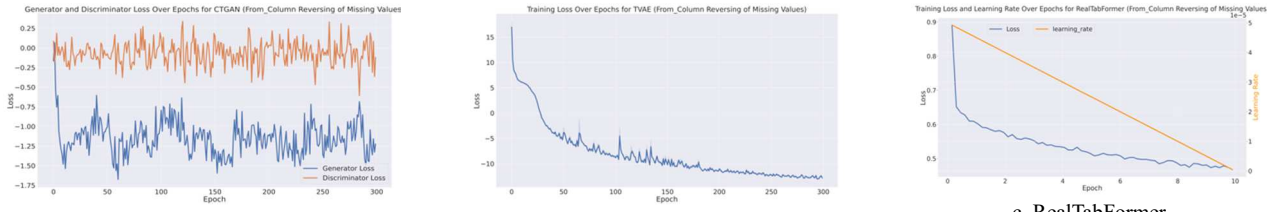
a) *The Data Quality Report* evaluates how well the synthetic data mimics the real data based on the average of the column shape and pair trend scores. The column shape describes the overall distribution of the columns. The column pair trends quantify the correlation of all pairs of

[1] https://docs.sdv.dev/rdt
[2] https://docs.sdv.dev/sdv
[3] https://pypi.org/project/realtabformer/
[4] https://docs.sdv.dev/sdmetrics



a. CTGAN b. TVAE c. RealTabFormer

Fig. 4. Training graphs of SDG when using random reversible data transformation strategy of missing values.



a. CTGAN b. TVAE c. RealTabFormer

Fig. 5. Training graphs of SDG when using missing indicators (*from_column*) reversible data transformation strategy to reverse of missing values.

columns. A higher score indicates that the synthetic data is close to the real data.

b) Missing Value Similarity metric measures whether the synthetic data has the same distribution of missing values as the real data for a given column. The score for this metric ranges between 0 and 1, where a value closer to 1 implies the synthetic data accurately captures the distribution of missing values.

2) Evaluation Metrics for XGBoost.

We used weighted average precision, recall, and F1-score as evaluation metrics to evaluate the performance of the XGBoost in the prediction of student engagement levels.

V. Results

A. Students' Learning Activities Digital Twin Evaluation

We sampled synthetic data of the same size as the training data (21,328) from each SDG for evaluation. The data quality report of the synthetic data for both reversible data transformation strategies is shown in Table II. The RealTabFormer outperformed CTGAN and TVAE in terms of column pair trends, shapes, and overall data quality score in both reversible data transformation strategies. It obtained a score of 95.32%, 94.15%, and 94.73% for column pair trends, shapes, and overall data quality, respectively, for the *random* strategy. In the case of the *from_column* (missingness indicator) strategy, the RealTabFormer achieved approximately 98% across all data quality report metrics. This implies the RealTabFormer perfectly captures the distribution and correlations of the real data. Overall, it can be observed that all SDGs performed better when using the *from_column* (missingness indicator) reversible data transformation strategy to model the DT of the students' learning activities.

The missing value similarity of the synthetic data generated by each SDG is shown in Table III. In the *random* reversible data transformation strategy, all the SDGs achieved the same missing value similarity score of 0.9983 for *nevents* and *ndays_act*, and a score of 0.9954 for *nforum_posts*.

TABLE II. Data Quality Report of Synthetic Data.

Reversible Data Transformation Strategy	Model	Quality Report		
		Column Pair Trends (%)	Column Shapes (%)	Overall Quality Score (%)
Random (Proportion of missing values)	CTGAN	83.28	83.82	83.55
	TVAE	89.94	89.91	89.92
	RealTabFormer	**95.32**	**94.15**	**94.73**
From_Column (Missingness Indicator)	CTGAN	91.12	92.75	91.94
	TVAE	93.1	91.38	92.24
	RealTabFormer	**97.52**	**97.58**	**97.55**

The SDGs have the same scores because the *random* reversible data transformation strategy ensures that the missingness recreation follows approximately the same proportion as the real data. However, the specific entries labeled as missing may differ due to the randomness. In the case of the *from_column* (missingness indicator) reversible data transformation strategy, the scores for each SDG for the columns with missing values differ due to how each model generates the missingness indicator columns. It can be observed from Table III that the recreation of missing values in synthetic data through the *random* reversible data transformation strategy accurately captures the proportion of missing values better than using missingness indicators.

Fig.6 shows the distribution of the categorical column (*Engagement_Level*) generated by each SDG on both reversible data transformation strategies. It can be observed that the CTGAN attempts to sample balanced data better than TVAE and RealTabFormer in both reversible data transformation strategies. This can be attributed to the conditional generator and training-by-sampling methods used in the CTGAN architecture.

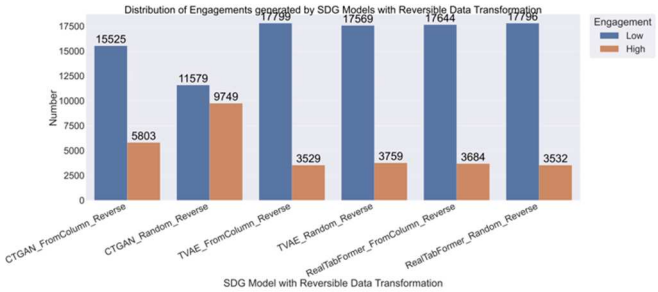


Fig.6. Distribution of Engagement levels generated by SDGs with reversible data transformation of missing values.

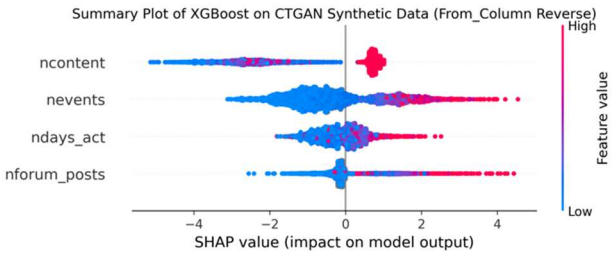


Fig.7. Impact of learning activities features on XGBoost prediction (CTGAN using missingness indicator reversible data transformation strategy of missing values).

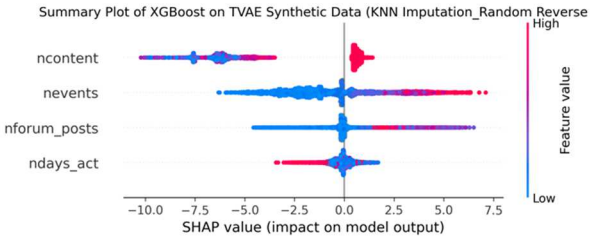


Fig.8. Impact of learning activities features on XGBoost prediction (TVAE using random reversible data transformation strategy of missing values).

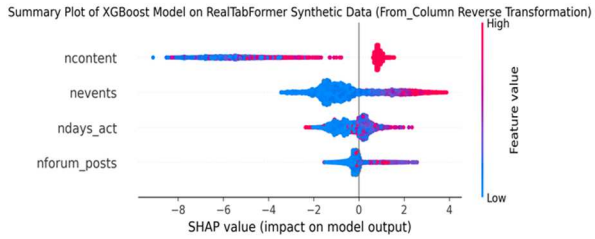


Fig.9. Impact of learning activities features on XGBoost prediction (RealTabFormer using missingness indicator reversible data transformation strategy of missing values).

TABLE III. Missing Value Similarity of Synthetic Data.

Reversible Data Transformation Strategy	Model	Missing Value Similarity		
		nevents	ndays_act	nforum_posts
Random (Proportion of missing values)	All SDGs	**0.9983**	**0.9983**	**0.9954**
From_Column (Missingness Indicator)	CTGAN	0.9927	0.9621	0.9584
	TVAE	0.9925	0.9533	0.9821
	RealTabFormer	0.9628	0.9921	0.9809

B. *Performance of XGBoost on the Prediction of Student Engagement Level*

We trained the XGBoost model on synthetic data sampled from each trained SDG and tested using 20% of real data not involved in the creation of synthetic data. This is to ensure unbiased utility evaluation of the generated synthetic data. The performance of XGBoost in predicting the student engagement level is shown in Table IV. In the case of CTGAN, XGBoost performed better with a score of 89% for precision and 88% for both recall and f1-score when trained on data generated by the CTGAN through the missingness indicator reversible data transformation strategy. For TVAE, XGBoost obtained an outstanding performance when trained on data sampled through the random reversible data transformation strategy. It obtained a score of 90% across all metrics. In the case of RealTabFormer, the XGBoost exhibits identical performance on trained data from both reversible data transformation strategies.

Fig. 7 to 9 illustrates the significance of the learning activities features on XGBoost in predicting the engagement level of students when trained using the synthetic data from each SDG with high data quality. The SHAP values generated for XGBoost indicate that *ncontent* and *nevents* were the most important features across all SDGs.

TABLE IV. Performance of XGBoost trained on Synthetic Data.

Reversible Data Transformation Strategy	SDG Model	Precision (%)	Recall (%)	F1-Score (%)
Random (Proportion of missing values)	CTGAN	88	84	85
	TVAE	**90**	**90**	**90**
	RealTabFormer	89	90	90
From_Column (Missingness Indicator)	CTGAN	89	88	88
	TVAE	89	90	89
	RealTabFormer	89	90	89

We applied a conditional sampling on the trained SDGs to generate balanced samples of the *Engagement_Levels*. Unfortunately, the RealTabFormer does not allow conditional sampling, hence, we performed this experiment on only TVAE and CTGAN. Table V shows the data quality of CTGAN and TVAE after a balanced data sampling. The data quality of both SDGs reduced after the conditional sampling, however, the scores obtained are above 70%. This indicates that the trained CTGAN and TVAE can generate a balanced synthetic dataset of student learning activities while still capturing the meaningful patterns of the real data. The XGBoost trained on the balanced data of CTGAN and TVAE achieved a score of above 80% across all metrics when tested on the real data. It

obtained the best scores of 89%, 84%, and 86% for precision, recall, and F1-score, respectively, when trained on balanced data sampled from the TVAE on a random reversible data transformation strategy.

TABLE V. Data Quality of SDGs and Performance of XGBoost trained on Balanced Dataset.

Reversible Data Transformation Strategy	SDG Model	Data Quality (%)	Precision (%)	Recall (%)	F1-Score (%)
Random (Proportion of missing values)	CTGAN	81.91	88	82	84
	TVAE	78.92	89	84	86
From_Column (Missingness Indicator)	CTGAN	81.91	89	81	83
	TVAE	80.24	88	83	85

VI. CONCLUSION

ML models are integrated into Learning Management Systems (LMS) to analyze students' learning behavior to provide hidden insights. However, LMS data is often characterized by missing values and imbalanced data. To handle imbalance, existing modeling approaches in education remove or impute missing values to sample balanced data. Imputing or ignoring missingness that depends on observed or unobserved data in predictive modeling can lead to bias.

To address these shortcomings, we propose the use of digital twins to create a balanced sample of students' learning activities to model their engagement while preserving missingness. We leveraged synthetic data generators such as Conditional Tabular Generative Adversarial Network (CTGAN), Tabular Variational Autoencoder (TVAE), and RealTabFormer with reversible data transformations to create a virtual replica of students' data. The CTGAN and TVAE generated balanced synthetic data that accurately captured the meaningful patterns of the real data. Furthermore, XGBoost trained on a balanced virtual replica of the students' learning activities data obtained an F1-score of above 80% in predicting the students' engagement levels when evaluated on real data with both complete and incomplete entries. Our findings demonstrate how digital twins can be used to address the complexities of data in the education sector.

In our future research directions, we will include personalized feedback for students based on their learning activities and investigate the impact of students' demographics on their engagement levels. Additionally, we will develop a dashboard for real-time analysis of the student engagement levels to enhance the monitoring of student learning behavior. This will also help the digital twin of the students' learning activities to evolve dynamically along with real-time data.

REFERENCES

[1] A. S. C. de Souza and L. Debs, "Concepts, innovative technologies, learning approaches and trend topics in education 4.0: A scoping literature review," *Social Sciences & Humanities Open*, vol. 9, p. 100902, Jan. 2024, doi: 10.1016/J.SSAHO.2024.100902.

[2] F. A. Orji, S. Fatahi, and J. Vassileva, "Data-Driven Approach for Student Engagement Modelling Based on Learning Behaviour," *Communications in Computer and Information Science*, vol. 1834 CCIS, pp. 334–342, 2023, doi: 10.1007/978-3-031-35998-9_46/FIGURES/3.

[3] G. Ben Brahim, "Predicting Student Performance from Online Engagement Activities Using Novel Statistical Features," *Arab J Sci Eng*, vol. 47, no. 8, pp. 10225–10243, Aug. 2022, doi: 10.1007/S13369-021-06548-W/FIGURES/8.

[4] "Canvas Network Person-Course (1/2014 - 9/2015) De-Identified Open Dataset - Canvas Network Dataverse." Accessed: Jan. 26, 2025. [Online]. Available: https://dataverse.harvard.edu/dataset.xhtml?persistentId=doi:10.7910/DVN/1XORAL

[5] N. Alruwais and M. Zakariah, "Student-Engagement Detection in Classroom Using Machine Learning Algorithm," *Electronics 2023, Vol. 12, Page 731*, vol. 12, no. 3, p. 731, Feb. 2023, doi: 10.3390/ELECTRONICS12030731.

[6] S. W. J. Nijman *et al.*, "Missing data is poorly handled and reported in prediction model studies using machine learning: a literature review," *J Clin Epidemiol*, vol. 142, pp. 218–229, Feb. 2022, doi: 10.1016/J.JCLINEPI.2021.11.023.

[7] N. Goel *et al.*, "The Importance of Modeling Data Missingness in Algorithmic Fairness: A Causal Perspective," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 9, pp. 7564–7573, May 2021, doi: 10.1609/AAAI.V35I9.16926.

[8] H. P. Das *et al.*, "Conditional synthetic data generation for robust machine learning applications with limited pandemic data," *ojs.aaai.org*, 2022, Accessed: Mar. 06, 2023. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/21435

[9] T. Das, Z. Wang, and J. Sun, "TWIN: Personalized Clinical Trial Digital Twin Generation," *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 402–413, Aug. 2023, doi: 10.1145/3580305.3599534.

[10] W. Booyse, D. N. Wilke, and S. Heyns, "Deep digital twins for detection, diagnostics and prognostics," *Mech Syst Signal Process*, vol. 140, p. 106612, Jun. 2020, doi: 10.1016/J.YMSSP.2019.106612.

[11] E. H. Glaessgen and D. S. Stargel, "The digital twin paradigm for future NASA and U.S. Air force vehicles," *Collection of Technical Papers - AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference*, 2012, doi: 10.2514/6.2012-1818.

[12] X. Wang, H. Asif, and J. Vaidya, "Preserving Missing Data Distribution in Synthetic Data," *ACM Web Conference 2023 - Proceedings of the World Wide Web Conference, WWW 2023*, pp. 2110–2121, Apr. 2023, doi: 10.1145/3543507.3583297.

[13] M. Hussain, W. Zhu, W. Zhang, and S. M. R. Abidi, "Student Engagement Predictions in an e-Learning System and Their Impact on Student Course Assessment Scores," *Comput Intell Neurosci*, vol. 2018, no. 1, p. 6347186, Jan. 2018, doi: 10.1155/2018/6347186.

[14] P. Sharma *et al.*, "Student Engagement Detection Using Emotion Analysis, Eye Tracking and Head Movement with Machine Learning," *Communications in Computer and Information Science*, vol. 1720 CCIS, pp. 52–68, 2022, doi: 10.1007/978-3-031-22918-3_5/FIGURES/9.

[15] J. E. C. Bachmann, I. F. Silveira, and V. F. Martins, "Digital Twins for Education: A Literature Review," *Simpósio Brasileiro de Informática na Educação (SBIE)*, pp. 722–736, Nov. 2024, doi: 10.5753/SBIE.2024.242288.

[16] S. M. E. Sepasgozar, "Digital Twin and Web-Based Virtual Gaming Technologies for Online Education: A Case of Construction Management and Engineering," *Applied Sciences 2020, Vol. 10, Page 4678*, vol. 10, no. 13, p. 4678, Jul. 2020, doi: 10.3390/APP10134678.

[17] S. Razzaq, B. Shah, F. Iqbal, M. Ilyas, F. Maqbool, and A. Rocha, "DeepClassRooms: a deep learning based digital twin framework for on-campus class rooms," *Neural Comput Appl*, vol. 35, no. 11, pp. 8017–8026, Apr. 2022, doi: 10.1007/S00521-021-06754-5/TABLES/4.

[18] B. Leotlela and M. Coetzee, "Digital Twin Monitoring of Classes and Students," *2024 IST-Africa Conference, IST-Africa 2024*, 2024, doi: 10.23919/IST-AFRICA63983.2024.10569389.

[19] L. Huang and K. E. Willcox, "Educational Digital Twin: Tackling Complexity in Educational Big Data," *Proceedings - 2024 IEEE International Conference on Big Data, BigData 2024*, pp. 1978–1985, 2024, doi: 10.1109/BIGDATA62323.2024.10825338.

[20] S. M. Lundberg, P. G. Allen, and S.-I. Lee, "A Unified Approach to Interpreting Model Predictions", doi: 10.5555/3295222.3295230.

[21] L. Xu, M. Skoularidou, A. Cuesta-Infante, and K. Veeramachaneni, "Modeling Tabular data using Conditional GAN," *Adv Neural Inf Process Syst*, vol. 32, 2019, Accessed: Feb. 19, 2023. [Online]. Available: https://github.com/DAI-Lab/CTGAN

[22] A. V. Solatorio and O. Dupriez, "REaLTabFormer: Generating Realistic Relational and Tabular Data using Transformers," Feb. 2023, Accessed: Jan. 15, 2024. [Online]. Available: https://arxiv.org/abs/2302.02041v1

[23] S. Kumi, M. Ray, S. Walia, R. K. Lomotey, and R. Deters, "Digital Twins for Stress Management Utilizing Synthetic Data," *2024 IEEE 5th World AI IoT Congress, AIIoT 2024*, pp. 329–335, 2024, doi: 10.1109/AIIOT61789.2024.10579038.

[24] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, vol. 13-17-August-2016, pp. 785–794, Aug. 2016, doi: 10.1145/2939672.2939785/SUPPL_FILE/KDD2016_CHEN_BOOSTING_SYSTEM_01-ACM.MP4.

[25] R. J. A. Little, "A test of missing completely at random for multivariate data with missing values," *J Am Stat Assoc*, vol. 83, no. 404, pp. 1198–1202, 1988, doi: 10.1080/01621459.1988.10478722.