Explainable Data-Driven Digital Twin for Stress Management

Sandra Kumi[1], Richard K. Lomotey[2], Madhurima Ray[2], Emma Cunningham[2], and Ralph Deters[1]

[1] Department of Computer Science, University of Saskatchewan, SK, Canada
[2] Computer Science, College of Engineering, The Pennsylvania State University, PA, USA
sandra.kumi@usask.ca,rkl5137@psu.edu, mvr6106@psu.edu, erc5617@psu.edu,and deters@cs.usask.ca

Abstract. Health Digital Twins (HDTs) rely on Machine Learning (ML) capabilities to provide insights and decision support for healthcare stakeholders. However, the complexity of ML models makes it challenging for non-technical stakeholders to understand the reasoning behind predictions, raising concerns about lack of transparency and trust in HDTs. To address these shortcomings, explainable AI (XAI) methods have been proposed to describe the predictions of ML models. Despite the efforts, the technical outputs of the XAI methods can be difficult for both expert and non-expert healthcare stakeholders to comprehend. In this paper, we propose a framework called Stress Management Digital Twin (SMDT) that integrates XAI methods with Large Language Models (LLMs) to generate natural language explanations for predictions of ML models. This enhances transparency and trustworthiness in HDTs. Specifically, we leveraged the Google Gemini and Mistral 7B to transform Shapley Additive exPlanations (SHAP) local explanation of stress management score prediction by Random Forest model into natural language narratives. From our experiments, the Google Gemini generated clear and concise narratives of the model's decision while retaining the accuracy of the given SHAP values. The findings from this study demonstrate that the proposed digital twin can be used for what-if-analysis of stress management scores while providing user-friendly explanations to enhance transparency and trust in HDTs.

Keywords: Digital Twin · Wearable Data · Synthetic Data Generative Models · Explainable AI · Large Language Model

1 Introduction

The concept of Digital Twins (DTs) was initially proposed to simulate and monitor the conditions of complex systems in the manufacturing and engineering sectors [1]. With the rapid evolution of technologies such as the Internet-of-Things (IoT), big data, communication protocols, artificial intelligence (AI), and so on, the adoption of DTs has expanded to streamline operations in domains such as smart cities, healthcare, farming, transportation, etc. [2]. In the healthcare domain, DTs are virtual representations of patients' data, medical workflows, and hospital architectures [3]. The integration of ML in Health Digital Twins (HDTs) offers tremendous benefits such as continuous monitoring of patients, precision diagnosis, personalized treatment, and clinical decision support [4]. Despite the benefits of making HDTs intelligent, the black-box nature of ML models makes it a challenge for stakeholders and clinicians to understand the reasoning behind its predictions [5][6][7].

To address this shortcoming, eXplainable AI (XAI) techniques have been leveraged into HDTs to interpret AI-driven decisions [6]. XAI methods such as Shapley Additive exPlanations (SHAP) use a game theory approach to explain the output of ML models [8]. However, these XAI methods are based on statistical approaches and visualizations, which may be difficult to interpret by non-technical users [9]. Without clear and human-centered explanations, end-users may struggle to understand the decisions of the model [10]. This may hinder the trust and transparency in AI-driven DTs. Recent developments on XAI have explored the in-context learning (ICL) capabilities of Large Language Models (LLMs) to explain the predictions made by ML models [11]. As a result, LLMs have emerged as powerful technologies for text summarization, coding tasks, and commonsense reasoning [11].

In this research, we posit that XAI methods can be combined with Large Language Models (LLMs) to provide user-friendly explanations of a model's prediction in HDTs. We propose a Stress Management Digital Twin (SMDT) framework that integrates XAI methods with LLMs to generate natural language explanations for predictions of ML models to enhance transparency and trustworthiness in HDTs. The proposed work leverages synthetic generative models to construct a digital replica of patients' wearable data. We then trained ML models on the digital twin data for stress management score predictions. The Random Forest model achieved the best mean absolute error (MAE) of approximately 3.89% when evaluated on real data. The SHAP framework is used to measure the contribution of each feature on the predicted stress management score to provide insights and interpretations of the model's decision. In this study, we focus on employing LLMs to transform the SHAP local explanation to natural language explanations. Our experimental analysis shows that the Google Gemini can generate clear and concise user-friendly explanations of the model's decision while retaining the accuracy of the given SHAP values. The main contributions of this work are summarized below:

- The use of synthetic data generative models to construct a high fidelity and utility digital twin (DT) from limited wearable data for stress management score prediction.
- Integration of Explainable Artificial Intelligence such as SHAP in DT to interpret predictions of stress management score by ML models.
- LLMs are introduced to transform SHAP explanations into natural language narratives to enable transparency and trustworthiness in DT.
- Conducted comprehensive experiments to demonstrate the feasibility of the proposed work.

The remainder of the paper is structured as follows. Section 2 highlights the related work on explainable artificial intelligence (XAI) in digital twins. The methodology of our proposed work is described in Section 3. The implementation and evaluation of our proposed approach are presented in Sections 4 and 5. Finally, our conclusions and future research directions are outlined in Section 6.

2 Related Work

This section discusses prior research on explainable artificial intelligence (XAI) in digital twins (DTs). According to Bertalanič et al. [12], DTs can be used to detect unusual changes in network signals that could indicate issues. Similarly, Naser et al. [13] developed a digital twin and k-means clustering algorithms for pattern and data outlier detection. Also, Ferdousi et al. [7] explored digital twins and how they are used in the healthcare industry

for well-being (WDTs). Together with AI, the issues of misdiagnosis can be minimized. The combination of DTs and AI can pinpoint exactly where the issue occurs and how long it lasts. Even though deep learning (DL) algorithms are becoming more and more used in different domains, a problem with DL algorithms is their "black-boxed" nature. To solve these problems, Gupta et al. [14] proposed eXplainable Digital Twins (XDT), which combines the transparency of explainable artificial intelligence (XAI) and the real-time simulation from digital twins (DT), allowing for clear explanations for their predictions. This makes it easier for users to understand the reasoning behind the predictions.

Moreover, Bhattacharya et al. [15] employed the term "Internet of Explainable Digital Twins", to refer to a physical implementation of explainable AI and digital twins in IoT systems. By pairing explainable AI with digital twins, the authors can explain the motivations behind the decisions being made, such as why they believe an input was faulty. Kobayashi and Alam [16] explore the integration of explainable, interpretable, and trustworthy AI within an Intelligent Digital Twin (IDT) framework, focusing on predicting the remaining useful life (RUL) of industrial systems. The study addresses the critical challenge of enhancing AI-driven decision-making transparency and reliability in predictive maintenance applications. To tackle this, the authors propose an explainable AI (XAI) approach that improves interpretability while ensuring the trustworthiness of RUL predictions. Their methodology combines advanced machine learning techniques with feature attribution methods to provide insights into model decisions. The results demonstrate improved accuracy and interpretability compared to traditional black-box models. However, the study acknowledges certain limitations, including the need for domain-specific customization and the challenge of balancing interpretability with predictive performance. Jox et al. [17] delve into the guidelines for using digital twins in the dairy industry while focusing on the model's validity. This calls for combining black-box models and white-box models, integrating machine algorithms along with simulations of physical and biochemical food properties. This implementation focuses on the use of XAI, which concentrates on the transparency of the AI model, helping people understand the model better.

The authors in [18] also propose a solution that combines XAI with digital twin technology, using real-time sensor data to track and predict soil carbon levels. XAI makes the model easier to understand, which helps farmers and policymakers feel more confident in the system's advice. The combination of XAI and DTs has also proven to be useful in cybersecurity [19]. According to Krishnaveni et al. [20], propose a multi-layered defense approach to detect and mitigate cyber threats by using techniques such as explainable AI (XAI), ensemble-based feature selection (EFFS), and deep learning models such as a hybrid GRU-LSTM network.

Further, Boulos and Zhang [5] detail how digital twins and other emerging technologies can be used to revolutionize personalized healthcare. Their paper highlights how current healthcare models are inefficient mainly due to the lack of personalized data that doctors have to work with when diagnosing patients. Specifically, factors such as genetics and environment are difficult or expensive to track consistently. To combat this problem, they suggested digital twins and predictive modeling. Digital twins provide doctors with models of patients that they can use to simulate treatment effects without affecting the real patient. Due to accuracy requirements, they suggest that explainable AI will play a pivotal role in the future of medicine, as the current issue of AI diagnoses is the lack of transparency in reasoning. Ferdousi et al. [6] explain how artificial intelligence can be used to improve health parameters, by upgrading DT models to better monitor system behaviors and perfect

them. Multiple methods were explored such as real-time processing techniques like Convolutional Neural Networks and LSTM for instant results, Anomaly detection to predict patterns, and Explainable AI to have reliable results.

Existing works of achieving XAI in DTs are based on statistical approaches and visualizations which may be difficult to interpret by non-technical users [9]. Wu et al. [9] introduced strategies of how Large Language Models (LLMs) can contribute to the advancement of XAI. Further, Zhang et al. [21] discuss how LLMs can help achieve explainability in Dynamic Digital Twins (DDTs), which are virtual representations of actual systems that evolve. The key challenge addressed is the low interpretability of AI-driven decision-making within these digital twins, which may hinder trust and adoption in critical domains such as infrastructure management and industrial automation. Their work can aid users to better understand system behaviors in dynamic environments. It is important to note that while LLMs enable transparency, reliance on pre-trained knowledge can limit domain specificity, requiring extra fine-tuning and hybrid approaches. Additionally, LLMs may introduce biases and illusions that do not match the model's behavior.

In this study, we focus on leveraging LLMs to enhance the outputs of XAI in stress management digital twin (SMDT) to generate user-friendly explanations. This will ensure that the explanations are based on actual model behavior, leading to more transparency and trustworthiness in digital twins.

3 Methodology

The architecture of the proposed approach of enhancing transparency and trust in digital twins is shown in Fig. 1. The architecture comprises a physical and digital space. The physical space is responsible for the collection of wearable data. The wearable data is then utilized in the digital space for analysis. The digital space consists of 3 modules: (i) the creation of stress management digital twin (SMDT), (ii) Predictive Analysis, and (iii) Explainability. The details of each module are discussed below.

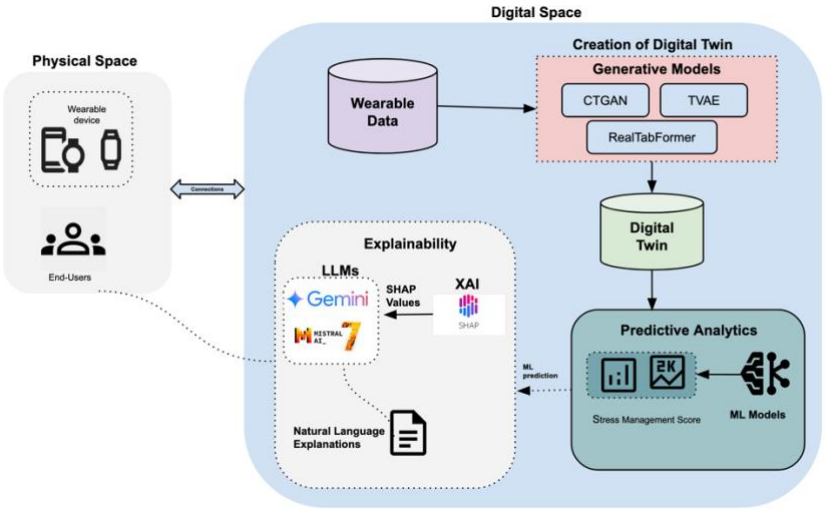

Fig. 1. Proposed architecture for LLM-Enhanced XAI in DT.

3.1 Data Collection

We collected the physical activity, sleep, and heart rate variability (HRV) Fitbit data tracked through a Google Pixel watch of two subjects, between October 13, 2022, and December 31, 2023, to create a digital twin for stress management score prediction. All the two subjects agreed to share their exported Fitbit data for this study through a consent form. A total of 17 features, consisting of 6 features from physical activity data, 8 features from sleep data, and 3 features from HRV data were extracted for the prediction of stress management score. The physical activity features are active minutes (sedentary, lightly, moderate, and very active), number of steps, and distance. We retrieved the respective stress management score of these features from Fitbit.

The HRV features include non-rapid eye movement heart rate (nremhr), root mean square of successive differences between normal heartbeats (rmssd), and resting heart rate. The sleep data features consist of light sleep, deep sleep, rapid eye movement sleep (rem), minutes asleep, minutes awake, minutes after wakeup, time in bed, and sleep score. A total of 435 samples were extracted for further analysis in our study. We split the collected data into training and testing datasets. The training dataset is used in the training of the generative models to create a digital twin. The testing dataset is used to evaluate the performance of ML models trained on the generated digital twin data. We used 235 samples as training dataset and 200 samples for testing dataset.

3.2 Creation of Stress Management Digital Twin (SMDT) Component

The Stress Management Digital Twin component leverages synthetic data generative models to create a virtual replica of the real data. We compare the performance of Realistic Relational and Tabular Transformer [22] (RealTabFormer), Tabular Variational Autoencoder [23] (TVAE), and Conditional Tabular Generative Adversarial Network [23] (CTGAN). RealTabFormer is a transformer-based framework that uses autoregressive transformer (GPT-2) and sequence-to-sequence (Seq2Seq) framework to generate synthetic data for non-relational and relational tabular data respectively. TVAE uses variational autoencoders (VAEs) to create synthetic data. CTGAN leverages mode-specific normalization, conditional generator, and training-by-sampling to learn the distribution and correlation of a real data to create a synthetic data.

3.3 Predictive Analytics Component

The stress management score prediction component uses the synthetic data from the SMDT module to train Machine Learning (ML) models for the prediction of stress management score. We compare the performance of four models, namely Random Forest, Extreme Gradient Boosting (XGBoost), Light Gradient Boosting Machine (LightGBM) and TabNet.

3.4 Explainability Component

The explainability component integrates eXplainable Artificial Intelligence (XAI) methods and Large Language Models (LLMs) to explain the predictions of ML models. In this study, we employed Shapley Additive exPlanations (SHAP) [8], as an XAI approach to get the local and global explanations of ML models. Local explanations focus on how each feature contributes to a single prediction of a model. The global explanations summarize the overall contribution of each feature across an entire dataset. SHAP adopts a game theory approach to explain the output of ML models by assigning SHAP value to each feature. The SHAP value measures how a feature contributes to a model's prediction by assigning positive or negative values. A positive SHAP value implies the feature increases the

model's prediction, while a negative value means the feature decreases the model's prediction. The SHAP value for a feature is computed as the weighted average of the marginal contributions of the feature across all possible combinations of features in a dataset using the formula in (1):

$$\emptyset_i = \sum_{S \subseteq F \setminus \{i\}} \frac{|S|!\,(|F| - |S| - 1)!}{|F|!} [f(S \cup \{i\}) - f(S)] \quad (1)$$

where $\emptyset_i$ denotes the SHAP value for feature i, F is the set of all features, S represents the subset of features excluding i, $f(S)$ is the model's prediction without the feature i, and $f(S \cup \{i\})$ is the model's prediction including the feature [8].

In this study, we focus on using LLMs to transform the visualized SHAP local explanations into natural language explanations. We evaluate the performance of two LLMs, namely Mistral 7B, and Google Gemini in the generation of understandable insights from the SHAP explanations.

4 Experiments

In this study, we implemented all experiments using Python. The experiments for stress management digital twin and training of ML models were executed on Google Colab notebooks. LLM explanations experiments ran on Apple M2 chip.

4.1 Creation of Stress Management Digital Twin (SMDT)

In the modeling of the SMDT, we used 235 samples from the collected data as training data to train the synthetic generative models. We fitted each model with all the extracted features from the real data without any transformation. The *realtabformer*[1] library was used in the training of RealTabFormer. The Synthetic Data Vault's (SDV)[2] Python package is used to train the TVAE and CTGAN models. We used the default parameters of the generative models for training to ensure a fair comparison. We used the GPU runtime on Google Colab notebooks during the training of RealTabFormer, and the CPU runtime for TVAE and CTGAN models. We initialized RealTabFormer with 200 epochs and the GPT-2 architecture to learn the relationships between the columns of the real data. During the training of RealTabFormer, each column is treated as a unique token to form a vocabulary for the GPT-2 to efficiently learn the relationships. The rows are represented as a sequence of tokens. An early stopping occurred after training for 79 epochs with a sampling efficiency of 99.6875%. The RealTabFormer leverages an optimal stopping criterion to stop training when the synthetic data's distribution is close to the real data's distribution. The TVAE was initialized with a loss factor of 2, regularization term of 1e-5 and 300 epochs for training. The CTGAN was trained with a learning rate of 2e-4, and weight decay of 1e-6 for both generator and discriminator for 300 epochs.

4.2 Training of ML Models for Stress Management Score Prediction

We train the ML models using only the generated synthetic data. Synthetic stress data is sampled from the synthetic generative model with highest data quality score to train the ML models for stress management score prediction. The mean imputation strategy is applied to replace missing values with a column's mean. We leveraged the Min-max normalization method to scale the predictive features' values to a range of 0 to 1.

[1] https://pypi.org/project/realtabformer/
[2] https://docs.sdv.dev/sdv

The filter feature selection approach was used in the selection of features. We applied the Pearson Correlation Coefficient (PCC) with a threshold of 0.8 to select the relevant features. Models were trained using tuned hyperparameters with a Repeated K-fold cross-validation of 5 folds and 3 repeats. We employed the Optuna hyperparameter optimization framework [24] to search for the best hyperparameters for each model.

4.3 Explanations for Stress Management Score Prediction

SHAP Explanations: We leveraged the tree explainer in the SHAP framework to compute SHAP values for global and local explanations of Random Forest, XGBoost, and LightGBM predictions on the test data. For the TabNet, we used the SHAP Kernel Explainer to generate explanations. A subset of the training data was used as background data for the Kernel Explainer to compute the SHAP values of the test data. We used the summary plot of the explainers to visualize the SHAP values for global explanation on the test data, and the waterfall plot to visualize the SHAP values for individual predictions (local explanation). The local SHAP explanation for a single prediction is computed in (2):

$$f(x) = \emptyset_o + \sum_{j=1}^{M} \emptyset_j \quad (2)$$

where $f(x)$ is the model's prediction for a specific data point x, $\emptyset o$ is the expected value of the model's output (average of the model's prediction across all data points.), M is the total number of features, and $\emptyset j$ denotes the SHAP values of the features. The global SHAP explanation can be derived from the aggregation of the local SHAP values.

Large Language Model (LLM) SHAP Values Explanation: We employed two LLMs, namely Mistral 7B and Google Gemini, to generate natural language explanations for SHAP values of a model's prediction for a specific data point. The feature values, SHAP values, expected value, and the model's prediction of the instance are given as

Table 1. Prompts used for LLMs to explain SHAP values

Section	Prompt 1	Prompt 2
Intro	Background information about the prediction task	
Data	Features values, SHAP values, baseline prediction, and model's prediction as JSON strings.	
Task	Provide a clear and concise summary of the SHAP values and how they contribute to the prediction.	Provide a clear and concise summary of the SHAP values and how they contribute to the prediction. Be sure to mention all features with their values and contributing SHAP values.

Table 2. Evaluation Metrics for LLM Explanations.

Metric	0	1	2
Accuracy: The correctness of the information provided in the explanation	Errors in SHAP values or contribution direction	Accurate with exact values or approximate values.	-
Completeness: The amount of information included in the explanation	Missing explanation for one or more features.	All features were described, but exact SHAP values or directions were not mentioned.	All feature values are given, and all contributions are described with directions and either exact values or descriptions ("contributed slightly").

input to the LLMs in a JSON file structure. The input representation for the LLM is shown in (3):

$$\{\{x_1 : v_1, \ldots, x_n : v_n\}, [\emptyset_1, \ldots, \emptyset_n], E[f(X)], f(X)\} \quad (3)$$

where $\{x_1: v_1, \ldots, x_n: v_n\}$ denotes the features and their values, $[\emptyset_1, \ldots, \emptyset_n]$ is SHAP values for the prediction, $E[f(X)]$ denotes expected value, and $f(X)$ is the prediction of the model.

We followed a zero-shot prompt engineering approach to instruct the LLMs to provide natural language explanations for the SHAP values provided. In this context of prompting, the LLMs are not provided with any examples of how the SHAP values should be explained. We crafted two prompts as shown in Table 1 to guide the LLMs. All the prompts give a brief background about the prediction task, the data, as shown in (3), and instructions of what to do. The prompt 2 differentiates from the prompt 1 by instructing the model to explicitly mention all features and their corresponding values and SHAP values. We do not control the length of the explanations generated by the LLMs, however, we set the temperature to 0.7 for both models. We leveraged the Ollama Python package to interact with the Mistral 7B model. For the Google Gemini, we utilized the '*gemini-2.0-flash-thinking-exp-01-21'* model to generate explanations.

4.4 Evaluation Metrics

Evaluation Metrics for SMDT: We used the Synthetic Data Metrics[3] (SDMetrics) to evaluate the fidelity and utility of the synthetic data generated by each generative model.

- *Fidelity.* The quality report method in the SDMetrics is used to evaluate the fidelity of the synthetic data. It quantifies how well the synthetic data captures the mathematical properties such as correlations and distributions of the real data. A higher score implies the synthetic data is close to the real data.
- *Utility* measures the success of using the generated synthetic data for Machine Learning prediction tasks. In the SDMetrics library, the Logistic regression model is used to compute the utility score for regression tasks. A score closer to 1 indicates the synthetic data generated can be used to perform ML tasks with high accuracy on the test data.

Evaluation Metrics for ML Models: We evaluated the performance of the ML models for the prediction of stress management score using Mean Absolute Error (MAE), and Root Mean Square Error (RMSE). A lower score for each metric indicates a better performance of the model. The MAE and RMSE of a prediction by an ML model are computed using the formula shown in (4) and (5) respectively, where yi is the true stress management score, $\hat{y}i$ is the predicted stress management score by a given model, n and is the number of samples being tested.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{4}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{5}$$

Evaluation Metrics for LLM Explanations: We adapt 2 metrics used in [25] to evaluate the explanations generated by Mistral 7B and Google Gemini. The explanations generated are evaluated in terms of accuracy, and completeness. The accuracy metric is scored on a scale from 0 to 1 and the completeness metric is scored on a scale from 0 to 2. Table 2

[3] https://docs.sdv.dev/sdmetrics

shows the definitions of the metrics used and the scoring scale. A higher score indicates the LLM model is generating quality explanations. In this study, the ratings of the explanations are done manually by two human independent raters.

5 Results

5.1 Stress Management Digital Twin (SMDT) Evaluation

We sampled 5000 rows of synthetic data from each generative model for fidelity and utility evaluation. The evaluation of the synthetic data generated by each model is shown in Table 3. The quality report score is based on the mean of the column shapes and pair trends score. The synthetic data generated by RealTabFormer obtained the best results with a quality report score of 93.39% and a utility score of approximately 0.26. The TVAE and CTGAN achieved a quality report score of 88.78% and 79.42% respectively. The synthetic data generated by CTGAN has the worst score of -0.4913 for utility. This implies that ML models trained on CTGAN's data may not perform well when evaluated on real data.

Table 3. Fidelity and Utility Evaluation of Synthetic Generative Models.

Model	Column Shapes (%)	Column Pair Trends (%)	Fidelity (Quality Report (%))	Utility
CTGAN	75.68	83.16	79.42	-0.4913
TVAE	84.85	92.7	88.78	0.2034
RealTabFormer	**90.62**	**96.15**	**93.39**	**0.2588**

5.2 Performance of ML Models on the Prediction of Stress Management Score

We used the 5000 samples generated by RealTabFormer to train the ML models for stress management score prediction. We trained the ML models on a subset of 14 features selected through Pearson correlation coefficient (PCC) of the synthetic data. We applied a threshold of greater than or equal to 0.8 for feature selection. Fig. 2 shows the Pearson Correlation coefficient heatmap of the synthetic and real data. It can be observed that the RealTabFormer accurately captures the correlations of the real data. Three features, namely, sedentary_minutes, distance, and timeInBed were discarded when the threshold was applied.

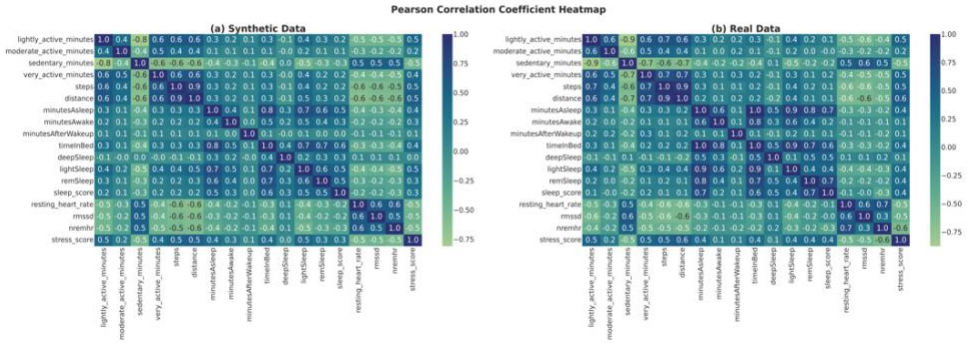

Fig. 2. Pearson Correlation Coefficient heatmap of RealTabFormer's synthetic data and real data.

We trained all ML models using a Repeated K-fold cross-validation of 5 folds and 3 repeats. We used a test data of 200 rows, not involved in the creation of the synthetic data,

to evaluate the performance of the ML models to further prove the utility of the data generated by RealTabFormer. The performance evaluation of models when trained using PCC features with default and tuned hyperparameters is shown in Table 4. The Random Forest model obtained the best MAE score of approximately 4.5 and LightGBM achieved the best RMSE of 5.9 when trained with default hyperparameters. The TabNet and XGBoost obtained the lowest. The *optuna* library was leveraged to tune the hyperparameters of each model. We set the study direction as minimize to find the lowest MAE score during the optimization process. For a fair comparison, we used 10 trails to search for the best hyperparameters for each model. The hyperparameter search space and best hyperparameter selected for each model can be found in [26]. As illustrated in Table 4, the performance of the models improved after tuning of the hyperparameters. The Random Forest model obtained the best performance with an MAE score of approximately 3.89 and RMSE score of 5.04. The XGBoost is the second-best performing model, achieving an MAE score of 3.9431 and RMSE score of 5.1266, followed by LightGBM with a score of approximately 4.17 and 5.36 for MAE and RMSE respectively. The TabNet is the least performing model with a score of 4.7746 for MAE and a score of 6.1493 for RMSE.

Table 4. Performance of ML Models on Real Test Data using default and tuned hyperparameters.

Category	Model	MAE	RMSE
PCC Features with Default Parameters	Random Forest	**4.5031**	6.3196
	XGBoost	6.1380	8.3424
	LightGBM	4.5374	**5.8969**
	TabNet	6.1071	7.3349
PCC Features with Tuned Hyperparameters	Random Forest	**3.8883**	**5.0406**
	XGBoost	3.9431	5.1266
	LightGBM	4.1737	5.3629
	TabNet	4.7746	6.1493

5.3 Stress Management Score Prediction Explanations

1) Global Explanations

Fig. 3 to 6 illustrate the summary plot of the global explanations for Random Forest, XGBoost, LightGBM, and TabNet respectively. The summary plot gives an overview of how different values of a feature affect the predictions of each model on the entire test data. In a summary plot, a data point is represented as a dot. The color of the dot represents the feature value. Higher feature values are represented as red and lower feature values are represented as blue. The x-axis represents the SHAP value for each feature. A positive SHAP value implies the feature increases the model's prediction, while a negative value means the feature decreases the model's prediction. The features are ranked by their importance on the y-axis. The width of the plot along each feature represents the distribution and density of SHAP values. Wider sections indicate higher density and more frequent feature values.

It can be deduced from Fig. 3, 4, and 5 that the Random Forest, XGBoost, and LightGBM exhibit similar behavior on the test data. All three models highlight *rmssd*, *lightly active minutes*, *steps*, *nremhr*, and *resting heart rate* features as the top 5 significant features influencing the prediction of stress management score. Similarly,

they all have the *minutesAfterWakeup* feature as the least important feature in the prediction. All 3 models predict that higher feature values for sleep data features (*lightSleep, minutesAsleep*, *sleep_score*, *remSleep*, *deepSleep*, and *minutesAwake*) except for *minutesAfterWakeup* increases the stress management score. In the case of the physical activity features, most high value data points for *lightly_active_minutes, steps,* and *very_active_minutes* are associated with high stress management score. For the heart rate variability features (*rmssd, nremhr,* and *resting heart rate*), the 3 models predict that majority of the lower feature values are associated with positive SHAP values, which increases the stress management score. It can also be observed that the positive SHAP values of the heart rate variability features (*rmssd, nremhr,* and *resting heart rate*) is a mixture of high and low feature values for the global explanation of all 3 models. This implies that the contribution of HRV features in the predictions may depend on the values of other features

As illustrated in Fig. 6, the top 5 features influencing the predictions of TabNet are steps, resting heart rate, lightSleep, rmssd, and very active minutes. Surprisingly, deepSleep is the least contributed feature in TabNet's prediction of stress management score. The TabNet predicts that higher values of lightSleep minutes is associated with high stress management score, while higher values of remSleep minutes, and sleep score is associated with low stress management score. Similarly, it predicts that higher values for HRV features lower the stress management score. For physical activity features, it predicts higher values, which increases the stress management score.

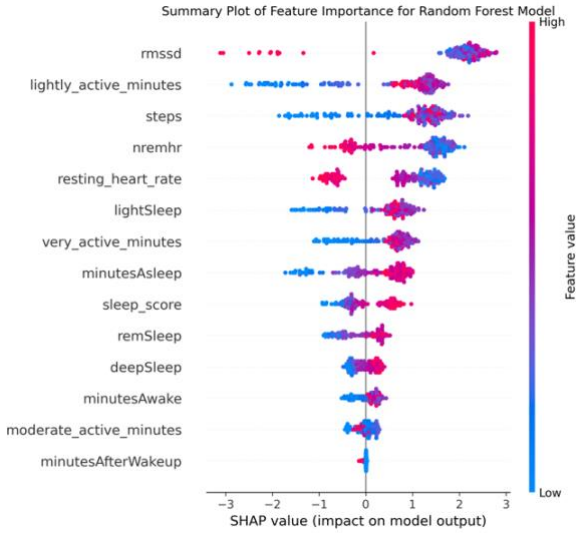

Fig.3. Summary plot of SHAP global explanation for Random Forest.

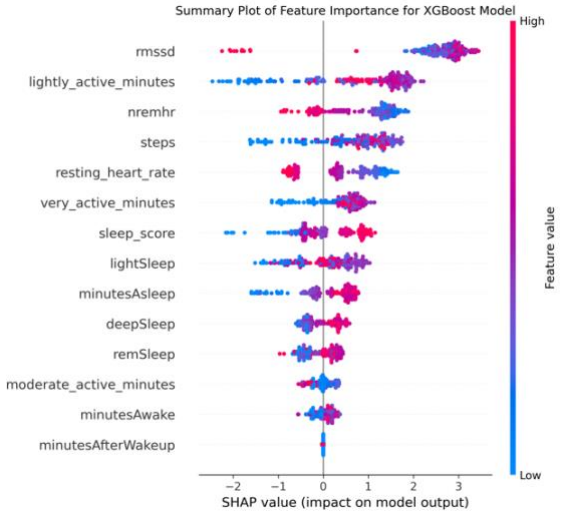

Fig.4. Summary plot of SHAP global explanation for XGBoost

2) LLM Explanations for Local SHAP Explanation

We built a stress management score prediction application using Streamlit as shown in Fig. 7. The application uses the Random Forest model for the prediction of stress management score. End-users can also request a SHAP local explanation or LLM enhanced SHAP local explanation of the model's prediction. The LLM uses the SHAP values provided in the waterfall plot to generate natural language explanations for easy understanding. Fig.8 shows a SHAP waterfall plot of a detailed breakdown of how features contribute to a specific prediction. Fig. 9 and 10 show the explanations

generated by Mistral 7b and Gemini respectively when prompted with prompt 2 described in Table I. The Mistral 7B model generates shorter explanations than Google Gemini. However, its explanation has few errors.

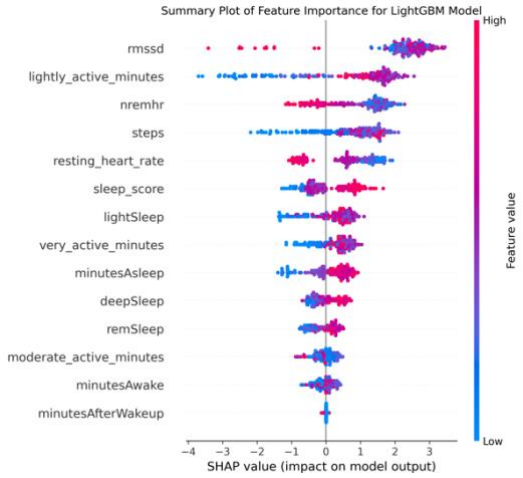

Fig.5. Summary plot of SHAP global explanation for LightGBM.

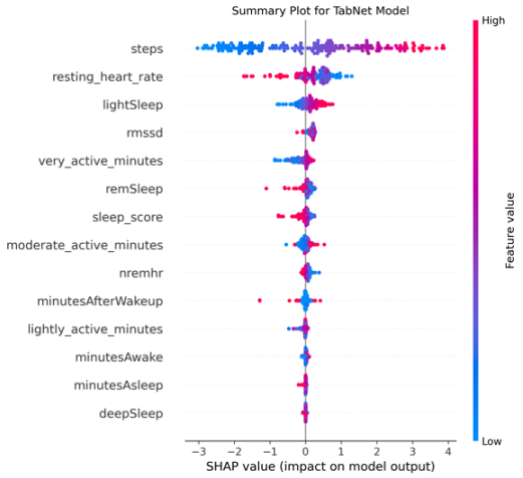

Fig.6. Summary plot of SHAP global explanation for TabNet

The accuracy of explanations in digital twin applications is important, as any misinterpretation can lead to mistrust and undermine confidence of end-users in the system. We generated the SHAP values of 5 samples from the test data to evaluate the explanations by LLMs. Two human independent raters manually rated the LLM explanations using the metrics in Table 2. The average score for each prompt for each model is shown in Table 5. Table 6 shows the Kappa inter-rater agreement scores between the raters. The Google Gemini model performed well across all prompts with an overall average score of 2 for accuracy, and 1.8 for completeness. In the case of Mistral 7B, it achieved an overall average score of 0.2, and 0.9 for accuracy, and completeness respectively. Although the Mistral 7B was deployed locally, it has a higher latency than the Google Gemini. The Google Gemini has a median latency of 10.57s for the 10 requests sent (5 samples x 2 prompts) while Mistral 7B has a median latency of 36.91s.

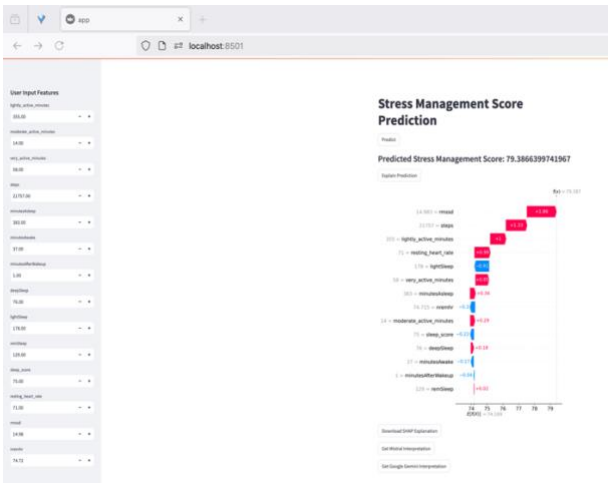

Fig.7. Stress Management Score Prediction Application

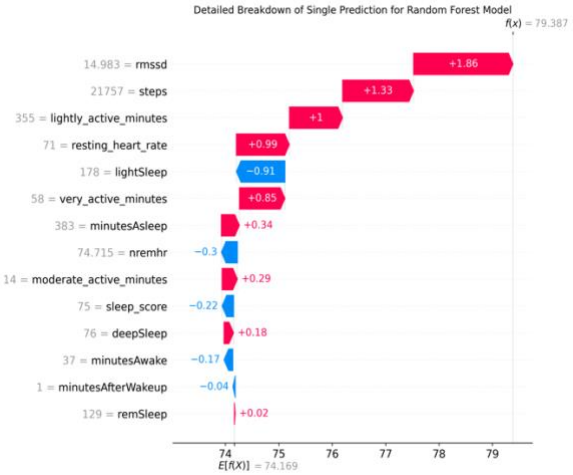

Fig.8. SHAP Local Explanation for Random Forest Single Instance Prediction.

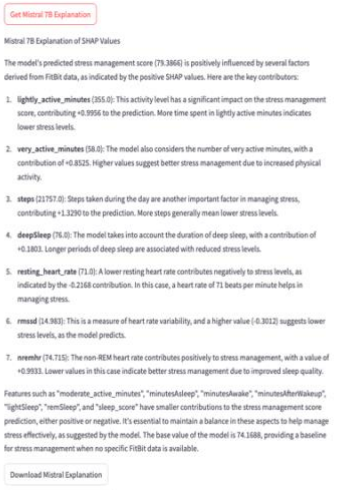

Fig.9. Mistral 7B's explanation of SHAP Local Explanation from Fig.8.

Fig.10. Gemini's explanation of SHAP Local Explanation from Fig.8.

Table 5. Average rating of LLMs on defined metrics.

Prompt	Mistral		Gemini	
	Accuracy	Completeness	Accuracy	Completeness
1	0	0.4	2	1.6
2	0.4	1.4	2	2
Overall	**0.2**	**0.9**	**2**	**1.8**

Table 6. Inter-Rater Agreement Scores (Cohen's Kappa) for Accuracy, and Completeness Across Prompts

Prompt	Mistral		Gemini	
	Accuracy	Completeness	Accuracy	Completeness
1	No variability	1.0	No variability	1.0
2	1.0	0.69	No variability	No variability

6 Conclusion

Transparency in AI-driven Health Digital Twins (HDTs) is essential to ensure that non-expert healthcare stakeholders understand the decision processes of ML models. In this paper, we presented a framework called Stress Management Digital Twin (SMDT) that integrates explainable AI (XAI) methods with Large Language Models (LLMs) to generate natural language explanations for predictions of ML models. This enhances transparency and trustworthiness in HDTs. We leveraged synthetic data generative models to create a digital twin of collected limited Fitbit data for the stress management score. The RealTabFormer outperformed CTGAN and TVAE in terms of fidelity and utility. ML models were trained on data generated by RealTabFormer to predict stress management scores. The Random Forest model achieved the best mean absolute error (MAE) of approximately 3.89% when evaluated on real data. To ensure transparency in our SMDT, the Shapley Additive exPlanations (SHAP) framework was used to measure the contribution of each feature on the predicted stress management score to provide insights and interpretations of the model's decision. Furthermore, Google Gemini and Mistral 7B models were leveraged to transform the visualized SHAP local explanations into natural language narratives. From our experiments, Google Gemini generated accurate natural language narratives of the SHAP explanations than Mistral 7B. In the prompting of the LLMs to generate the narratives, we did not specify the targeted audience. However, we observed that both models were able to generate narratives that can be easily understood by an audience with different knowledge backgrounds. Our findings from this study demonstrate that the proposed digital twin can be used for what-if-analysis of stress management scores while providing user-friendly explanations to enhance transparency and trust in HDTs.

While our experimental results prove that the integration of XAI and LLMs to generate natural language explanations for predictions of ML models could improve transparency in HDTs, our proposed approach has some notable limitations. The responses of LLMs are based on the type of prompts received. Poorly crafted prompts can result in wrong explanations for the end-users. Additionally, some LLMs tend to make mistakes, hence explanations generated will still require human in the loop to validate to avoid any misinterpretations. In our upcoming future research, we will focus on evaluating the performance of LLMs on different prompt techniques to ensure the best explanations are generated for end-users. We also aim to conduct usability studies, focusing on both expert and non-expert healthcare stakeholders to evaluate the usefulness of the proposed work. Additionally, we will employ an independent large language model to evaluate the explanations generated by Gemini to avoid human bias.

References

1. 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.
2. M. Liu, S. Fang, H. Dong, and C. Xu, "Review of digital twin about concepts, technologies, and industrial applications," *J Manuf Syst*, vol. 58, pp. 346–361, Jan. 2021, doi: 10.1016/J.JMSY.2020.06.017.
3. 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.
4. S. Kumi, M. Hilton, C. Snow, R. K. Lomotey, and R. Deters, "SleepSynth: Evaluating the use of Synthetic Data in Health Digital Twins," *Proceedings - 2023 IEEE International Conference on Digital Health, ICDH 2023*, pp. 121–130, 2023, doi: 10.1109/ICDH60066.2023.00027.
5. M. N. Kamel Boulos and P. Zhang, "Digital Twins: From Personalised Medicine to Precision Public Health,"

Journal of Personalized Medicine 2021, Vol. 11, Page 745, vol. 11, no. 8, p. 745, Jul. 2021, doi: 10.3390/JPM11080745.
6. R. Ferdousi, F. Laamarti, and A. El Saddik, "Artificial intelligence models in digital twins for health and well-being," *Digital Twin for Healthcare: Design, Challenges, and Solutions*, pp. 121–136, Jan. 2023, doi: 10.1016/B978-0-32-399163-6.00011-1.
7. R. Ferdousi, F. Laamarti, M. A. Hossain, C. Yang, and A. El Saddik, "Digital twins for well-being: an overview," *Digital Twin*, vol. 1, p. 7, Feb. 2022, doi: 10.12688/DIGITALTWIN.17475.2.
8. S. M. Lundberg, P. G. Allen, and S.-I. Lee, "A Unified Approach to Interpreting Model Predictions", doi: 10.5555/3295222.3295230.
9. X. Wu *et al.*, "Usable XAI: 10 Strategies Towards Exploiting Explainability in the LLM Era," Mar. 2024, Accessed: Mar. 06, 2025. [Online]. Available: https://arxiv.org/abs/2403.08946v1
10. F. Doshi-Velez and B. Kim, "Towards A Rigorous Science of Interpretable Machine Learning," Feb. 2017, Accessed: Mar. 07, 2025. [Online]. Available: https://arxiv.org/abs/1702.08608v2
11. N. Kroeger, D. Ley, S. Krishna, C. Agarwal, and H. Lakkaraju, "Are Large Language Models Post Hoc Explainers?," Oct. 2023, Accessed: Mar. 07, 2025. [Online]. Available: https://arxiv.org/abs/2310.05797v3
12. B. Bertalanič, V. Hanžel, and C. Fortuna, "Explainable semantic wireless anomaly characterization for digital twins," *Computer Networks*, vol. 251, p. 110660, Sep. 2024, doi: 10.1016/J.COMNET.2024.110660.
13. M. Z. Naser, "Digital twin for next gen concretes: On-demand tuning of vulnerable mixtures through Explainable and Anomalous Machine Learning," *Cem Concr Compos*, vol. 132, p. 104640, Sep. 2022, doi: 10.1016/J.CEMCONCOMP.2022.104640.
14. P. K. Gupta, B. D. Mazumdar, S. N. Pillai, and R. S. Komaragiri, "Towards the Development of eXplainable Digital Twins for Precision Agriculture," *1st International Conference on Pioneering Developments in Computer Science and Digital Technologies, IC2SDT 2024 - Proceedings*, pp. 64–69, 2024, doi: 10.1109/IC2SDT62152.2024.10696477.
15. P. Bhattacharya, M. S. Obaidat, S. Sanghavi, V. Sakariya, S. Tanwar, and K. F. Hsiao, "Internet-of-Explainable-Digital-Twins: A Case Study of Versatile Corn Production Ecosystem," *Proceedings of the 2022 IEEE International Conference on Communications, Computing, Cybersecurity and Informatics, CCCI 2022*, 2022, doi: 10.1109/CCCI55352.2022.9926502.
16. K. Kobayashi and S. B. Alam, "Explainable, interpretable, and trustworthy AI for an intelligent digital twin: A case study on remaining useful life," *Eng Appl Artif Intell*, vol. 129, p. 107620, Mar. 2024, doi: 10.1016/J.ENGAPPAI.2023.107620.
17. D. Jox, D. Hummel, J. Hinrichs, and C. Krupitzer, "A Conceptual Framework for Predictive Digital Dairy Twins: Integrating Explainable AI and Hybrid Modeling," *Proceedings of the International Food Operations and Processing Simulation Workshop, FOODOPS*, vol. 2024-September, 2024, doi: 10.46354/I3M.2024.FOODOPS.007.
18. D. An and Y. Q. Chen, "Explainable Artificial Intelligence (XAI) Empowered Digital Twin on Soil Carbon Emission Management Using Proximal Sensing," *2023 IEEE 3rd International Conference on Digital Twins and Parallel Intelligence, DTPI 2023*, 2023, doi: 10.1109/DTPI59677.2023.10365455.
19. I. H. Sarker, H. Janicke, A. Mohsin, A. Gill, and L. Maglaras, "Explainable AI for cybersecurity automation, intelligence and trustworthiness in digital twin: Methods, taxonomy, challenges and prospects," *ICT Express*, vol. 10, no. 4, pp. 935–958, Aug. 2024, doi: 10.1016/J.ICTE.2024.05.007.
20. S. Krishnaveni, T. M. Chen, M. Sathiyanarayanan, and B. Amutha, "CyberDefender: an integrated intelligent defense framework for digital-twin-based industrial cyber-physical systems," *Cluster Comput*, vol. 27, no. 6, pp. 7273–7306, Sep. 2024, doi: 10.1007/S10586-024-04320-X/FIGURES/8.
21. N. Zhang *et al.*, "Large Language Models for Explainable Decisions in Dynamic Digital Twins," May 2024, Accessed: Mar. 06, 2025. [Online]. Available: https://arxiv.org/abs/2405.14411v2
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. 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: Mar. 12, 2024. [Online]. Available: https://github.com/DAI-Lab/CTGAN
24. T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, "Optuna: A Next-generation Hyperparameter Optimization Framework," *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 2623–2631, Jul. 2019, doi: 10.1145/3292500.3330701.
25. A. Zytek, S. Pido, S. Alnegheimish, L. Berti-Equille, and K. Veeramachaneni, "Explingo: Explaining AI Predictions using Large Language Models," Dec. 2024, doi: 10.1109/BIGDATA62323.2024.10825114
26. "Section V Subsection B Hyperparameter Search Space for Prediction Models - Google Docs." Accessed: Mar. 07, 2025. [Online]. Available: https://docs.google.com/document/d/1V_VlJVIecYKTuBP0nxzw-ONQsLstFQWBlz6aozj_8K0/edit?tab=t.0