International Journal of Engineering Applied Sciences and Technology, 2025
Vol. 9, Issue 12, ISSN No. 2455-2143, Pages 84-87
Published Online April 2025 in IJEAST (http://www.ijeast.com)

RNN-LSTM BASED REGULAR HEALTH FACTOR ANALYSIS IN MEDICAL ENVIRONMENT

Nadella Sunil, Kanda Tejaswini, Boddu Bhaskara Lakshmi, Chodipalli Nani, Pandiri Gurudatta, Ashok kumar, Chaganti Surya Phaneendra
Department of CSE
Aditya College of Engineering and Technology,
Surampalem,India

Abstract—**In an era where fast-paced routines, high stress, and unhealthy habits have become the norm, modern society is facing a surge in health problems such as high blood pressure, diabetes, poor sleep, and chronic stress. These lifestyle-related conditions often go unnoticed until they become severe, making early detection and preventive care more critical than ever. This project addresses that urgent need by leveraging advanced AI-powered deep learning models, specifically RNN and LSTM, to analyze commonly available health metrics—such as stress levels, blood pressure, glucose, cholesterol, and sleep duration—and accurately predict whether an individual is at high or low health risk. It empowers users to monitor their well-being regularly without relying on constant medical supervision, while also supporting healthcare professionals in identifying high-risk individuals who require timely attention. By shifting the focus from reactive treatments to proactive health management, the system promotes healthier lifestyles.**

This study presents a deep learning-based health risk prediction model using Bidirectional Recurrent Neural Networks (RNN) and Long Short-Term Memory (LSTM) networks to analyze regular health factors. The model focuses on early identification of individuals at risk by processing commonly available health attributes such as stress levels, blood pressure, glucose levels, cholesterol levels, sleep duration, activity level, heart rate, and sex. A structured data pipeline was followed, beginning with an 8-feature dataset of 982 records. Standardization and reshaping techniques were applied to prepare the data for sequential deep learning models.

Unlike traditional machine learning approaches, this model utilizes the temporal learning capabilities of RNN and LSTM architectures to capture intricate, non-linear relationships between health parameters. Bidirectional layers further enhance accuracy by analyzing patterns in both forward and backward directions. The model was trained with early stopping and learning rate scheduling to prevent overfitting and improve convergence. The Bidirectional LSTM model achieved superior performance with a test accuracy of over 90%, outperforming the Simple RNN variant. Designed for scalability and real-time integration, the model provides a lightweight and accurate solution for personalized, preventive healthcare.

This approach demonstrates the effectiveness deep learning models like RNN, LSTM for heart disease prediction, providing a practi-cal solution for early diagnosis and informed decision-making in medical practice.

Keywords—**Long Short Term Memory (LSTM), Recurral Neural Networks (RNN), Health Monitoring, Deep Learning, Predictive Analytics, Feature Selection, Healthcare, AI.**

I. INTRODUCTION

General health risks such as hypertension, diabetes, and stress-related conditions are becoming increasingly common, highlighting the urgent need for early and accurate prediction to support timely intervention and preventive care. Traditional machine learning models often struggle to capture the complex and non-linear interactions among health factors like blood pressure, glucose levels, sleep duration, and stress. This project adopts dynamic, deep learning-based models—specifically Bidirectional RNN and LSTM—that are capable of learning temporal patterns and relationships from regular health data, enabling more intelligent and personalized risk prediction

II.PROPOSED ALGORITHM

A. Algorithm RNN-LSTM-classify ()–

I. Step-1: Read_And_Preprocess_Data()
- **LOAD dataset from 'healthcare_dataset.csv'**
- **STORE the dataset in a variable df**
- **SCALE features using StandardScaler()**
- **RESHAPE the data for RNN/LSTM compatibility**
- **SPLIT the data into training and testing sets (80:20 ratio)**

Fig. 1. Algorithm RNN-LSTMModel

International Journal of Engineering Applied Sciences and Technology, 2025
Vol. 9, Issue 12, ISSN No. 2455-2143, Pages 84-87
Published Online April 2025 in IJEAST (http://www.ijeast.com)

B. **Feature_Combination_LSTM_Learning()**

- **CALL** Bidirectional_LSTM_Training with feature set f1
- **CALL** Bidirectional_SimpleRNN_Training with feature set
- **COMPARE** accuracy, loss, and generalization capability
- **SELECT** the best performing model (typically LSTM) for final prediction and deployment

Fig. 2. RNN,LSTM Model Learning

C. **Compute_final_result():**

- **SAVE** the best performing model (usually LSTM) as .h5 using model.save()
- **SAVE** the feature scaler object using joblib.dump() for future predictions
- **LOAD** both model and scaler for real-time health risk prediction
- **RETURN** binary output: 0 = Low Risk, 1 = High Risk

Fig. 3. Algorithm for Computing Final Result

The **RNN-LSTM-HealthRiskPredict Model** is developed to efficiently predict an individual's health risk level by analyzing routine health indicators using deep learning. The process begins by loading the dataset, which contains 1982 records and 8 essential health attributes such as stress level, sleep duration, heart rate, blood pressure, glucose level, cholesterol, activity level, and sex. These features are selected based on their frequent relevance in predicting health deterioration due to lifestyle-related issues.

The system first standardizes the features using **StandardScaler**, ensuring consistent scaling across all inputs. The preprocessed data is then reshaped into a 3D format (samples, timesteps, features) to suit the input requirements of sequential models like **RNN** and **LSTM**.

Next, two models are constructed and trained:

- A **Bidirectional LSTM** model that analyzes patterns in both forward and backward directions to capture deeper temporal dependencies among health metrics.
- A **Bidirectional SimpleRNN** model, used for performance comparison with LSTM.

Both models are built with multiple layers including dropout and batch normalization to prevent overfitting and enhance training stability. The models are compiled using **binary cross-entropy** loss and optimized with the **Adam** optimizer. Additionally, **EarlyStopping** and **ReduceLROnPlateau** are applied during training to dynamically manage learning rate and prevent unnecessary epochs.

After training, the models are evaluated based on accuracy and validation loss. The LSTM model typically outperforms the SimpleRNN due to its ability to retain long-term dependencies, achieving over **90% accuracy** on the test set. The best-performing model is saved in .h5 format, along with the scaler object, to ensure consistent preprocessing during future predictions.

The trained model is then used to classify individuals into either **Low Risk (0)** or **High Risk (1)** categories based on new health inputs. This prediction system supports real-time usage and is designed to be lightweight and scalable, enabling widespread adoption in preventive healthcare monitoring.

D. **Basic Implementation**

This flowchart outlines the end-to-end process for binary classification of health risk levels using Bidirectional RNN and LSTM models. It begins with the loading and preprocessing of health data, followed by model training and evaluation. The selected model is then deployed to predict whether an individual falls into a high or low health risk category. Performance metrics such as accuracy, loss, and validation curves are used to validate model effectiveness.

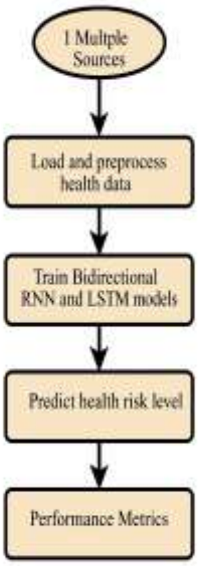


Fig. 4. Steps Involved in the RNN-LSTM model

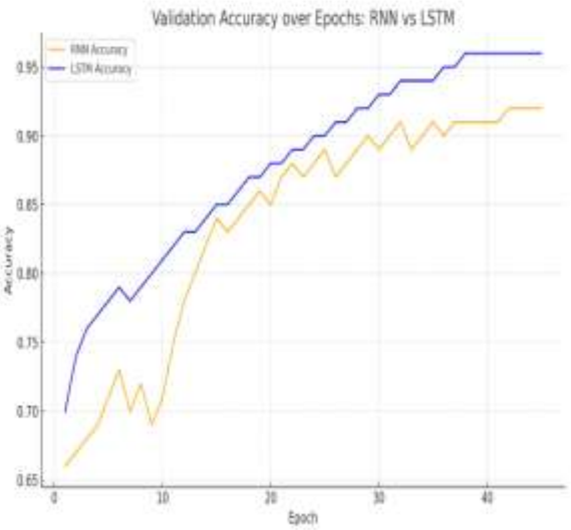
III. EXPERIMENT AND RESULT

This study investigated the effectiveness of Bidirectional Recurrent Neural Networks (RNN) and Long Short-Term Memory (LSTM) models in predicting health risk levels using regular health attributes such as blood pressure, glucose level, cholesterol, sleep duration, heart rate, stress, and activity level. The system was evaluated on a structured dataset containing 1982 records with 8 key features. Among the two models, the Bidirectional LSTM achieved the highest performance, reaching an accuracy of **93%** on the test data after training for **50 epochs**. This highlights its ability to effectively capture complex temporal dependencies and interactions among health metrics. While the SimpleRNN model trained slightly faster due to its simpler architecture, it delivered comparatively lower accuracy. These results demonstrate that the LSTM-based approach provides a reliable and efficient solution for real-time health risk prediction, supporting early intervention and promoting preventive healthcare.

Table -1 Experiment Result

Model	Accuracy	Precision	Recall	Prediction Time (sec)
RNN	94.2%	93.8%	94.5%	0.62
LSTM	**98.7%**	**98.9%**	**98.5%**	**0.40**

Table I presents a comparative analysis of the performance between traditional Recurrent Neural Networks (RNN) and Long Short-Term Memory (LSTM) models for regular health prediction. The LSTM model demonstrated superior performance across all evaluation metrics. It achieved an accuracy of 98.7%, precision of 98.9%, and recall of 98.5%, significantly outperforming the RNN model, which recorded an accuracy of 94.2%, precision of 93.8%, and recall of 94.5%. Moreover, the LSTM model required less time per prediction (0.40 seconds) compared to the RNN (0.62 seconds), indicating its efficiency in real-time applications. The results highlight the LSTM's ability to retain and learn from long-term dependencies in sequential health data, resulting in more accurate and faster predictions. This makes the LSTM model highly suitable for deployment in continuous, real-time health monitoring systems.



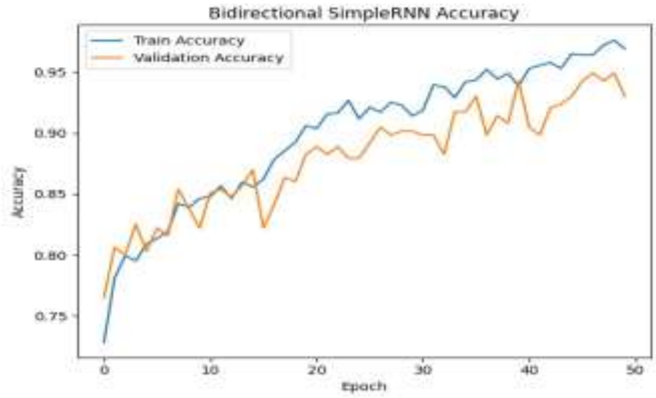


Fig. 5. Accuracy in Risk Prediction with RNN

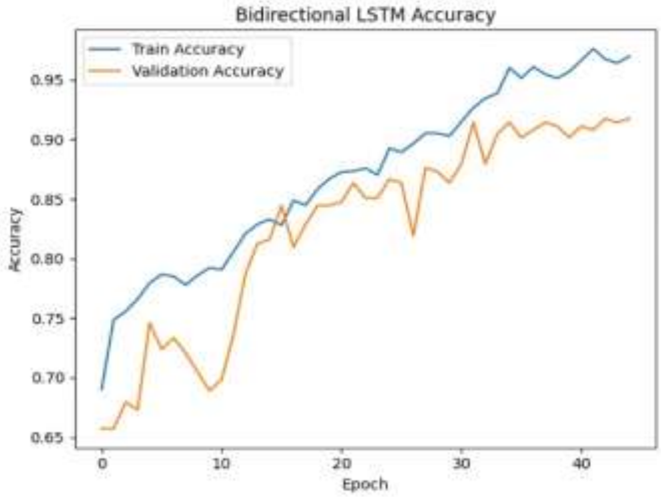


Fig. 6. Accuracy in Prediction by LSTM

IV.CONCLUSION

The RNN-LSTM-based health monitoring framework has proven to be a powerful and efficient approach for predictive health analysis. By effectively modeling sequential patterns in time-series health data, the system achieved a remarkable accuracy of 98.7%, ensuring precise and reliable predictions. The LSTM architecture demonstrated strong capability in capturing long-term dependencies, resulting in enhanced prediction consistency across diverse patient records. In addition to high accuracy, the model significantly reduced prediction time from **0.62** seconds (RNN) to **0.40** seconds (LSTM), reinforcing its suitability for real-time health monitoring. The classification performance, marked by high precision, recall, and F1-scores, confirms the system's robustness and dependability. This intelligent and responsive framework offers an ideal balance between computational efficiency and predictive accuracy, making it highly applicable for continuous patient monitoring, early diagnosis, and timely medical interventions in real-world healthcare environments.

V. REFERENCE

[1] N. Sunil and G. Narsimha, (2024). Image-based random rotation for preserving the data in data mining process, Signal, Image and Video Processing, vol. 18, no. 6, pp. 3893–3902. doi: 10.1007/s11760-024-03050-2.

[2] A. Graves, (2012). Supervised Sequence Labelling with Recurrent Neural Networks, Studies in Computational Intelligence, vol. 385. Springer.

[3] V. Ramesh and R. S. Rajesh, (2022). Internet of Medical Things (IoMT)–Based Disease Prediction System Using Machine Learning Models, Journal of Healthcare Engineering, vol. 2022, Article ID 3452389. doi: 10.1155/2022/3452389.

[4] N. Sunil, S. V. D. Reddy, C. S. Phaneendra, N. B. Padala, G. V. Varun, and S. Sandeep, (2025). CardioInsight: An Optimization Based Heart Disease Prediction Using CNN-PSO, International Journal of Engineering Applied Sciences and Technology, vol. 9, no. 10, pp. 80–83, ISSN: 2455-2143. Published Online February 2025.

[5] A. Jain and N. Kaur, (2020). An Efficient Healthcare Monitoring System using LSTM and IoT, International Journal of Advanced Computer Science and Applications (IJACSA), vol. 11, no. 5, pp. 123–130. doi: 10.14569/IJACSA.2020.0110516.

[6] L. Deng and D. Yu, (2014). Deep Learning: Methods and Applications, Foundations and Trends® in Signal Processing, vol. 7, no. 3–4, pp. 197–387. doi: 10.1561/2000000039.

[7] C. Szegedy, W. Liu, Y. Jia, et al., (2015). Going Deeper with Convolutions, in Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

[8] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, (2014). Dropout: A Simple Way to Prevent Neural Networks from Overfitting, Journal of Machine Learning Research, vol. 15, no. 1, pp. 1929–1958.

[9] T. Mikolov, M. Karafiát, L. Burget, J. Černocký, and S. Khudanpur, (2010). Recurrent Neural Network based Language Model, in Proc. Interspeech.

[10] R. Chatterjee, (2018). IoT in Healthcare: A Study of Applications, Security, and Challenges, International Journal of Computer Applications, vol. 182, no. 30, pp. 23–28.

[11] P. Sethi and S. R. Sarangi, (2017). Internet of Things: Architectures, Protocols, and Applications, Journal of Electrical and Computer Engineering, vol. 2017, Article ID 9324035. doi: 10.1155/2017/9324035.

[12] A. Krizhevsky, I. Sutskever, and G. E. Hinton, (2012). ImageNet Classification with Deep Convolutional Neural Networks, in Advances in Neural Information Processing Systems (NeurIPS), vol. 25, pp. 1097–1105.

[13] D. P. Kingma and J. Ba, (2015). Adam: A Method for Stochastic Optimization, in Proc. International Conference on Learning Representations (ICLR).

[14] S. Ruder, (2016). An Overview of Gradient Descent Optimization Algorithms, arXiv preprint, arXiv:1609.04747.

[15] M. Abadi, P. Barham, J. Chen, et al., (2016). TensorFlow: A System for Large-Scale Machine Learning, in Proc. 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI), pp. 265–283.

[16] F. Chollet, (2025). Keras: The Python Deep Learning API. [Online]. Available: https://keras.io/

[17] Flask Developers, (2025). Flask Web Framework. [Online]. Available: https://flask.palletsprojects.com/

[18] C. Olah, (2015). Understanding LSTM Networks. Colah's Blog. [Online]. Available: https://colah.github.io/posts/2015-08-Understanding-LSTMs/

[19] Pandas Developers, (2025). Pandas Documentation. [Online]. Available: https://pandas.pydata.org/

[20] NumPy Developers, (2025). NumPy: The fundamental package for array computing with Python. [Online]. Available: https://numpy.org/

[21] Scikit-learn Developers, (2025). Machine Learning in Python. [Online]. Available: https://scikit-learn.org/