A CNN-LSTM Hybrid Architecture for Early Stroke Risk Prediction: A Comparative Study with Random Forest and XGBoost

Dr. E. Bijolin Edwin[1], Dr. Jyoti Sekhar Banerjee[2]

[1] Lincoln University College, 47301, Petaling Jaya, Selangor, Darul Eshan, Malaysia

[2] Bengal Institute of Technology, Kolkata

[3]Karunya Institute of Technology and Sciences, Coimabore-641114, Tamil Nadu, India
Email ID : bijolin@karunya.edu, jyotisekhar.banerjee@bitcollege.in

Abstract: Stroke remains the second-leading cause of global mortality, affecting approximately 15 million individuals annually. Early risk prediction through artificial intelligence offers a transformative pathway to preventive clinical intervention. This study investigates the performance of three machine learning and deep learning approaches: Random Forest (RF), eXtreme Gradient Boosting (XGBoost), and a novel Convolutional Neural Network-Long Short-Term Memory (CNN-LSTM) Hybrid Architecture for binary stroke risk classification using the Kaggle Stroke Prediction Dataset and the CDC Behavioural Risk Factor Surveillance System (BRFSS) 2015 survey. The proposed CNN-LSTM hybrid achieved the highest AUC-ROC of 0.967, sensitivity of 0.940, and F1-score of 0.930 under stratified 10-fold cross-validation, significantly outperforming both RF (AUC 0.891) and XGBoost (AUC 0.913). Three structured experiments confirm that the proposed architecture outperforms classical ensemble methods with statistically significant margins ($p < 0.05$). The findings demonstrate the potential of integrating spatial convolutional feature extraction with temporal LSTM-based modelling to improve stroke prediction in clinical decision-support systems.

Keywords: *Stroke prediction; CNN-LSTM hybrid; Random Forest; XGBoost; deep learning; clinical risk assessment; AUC-ROC; SMOTE; electronic health records.*

Introduction

According to the World Health Organization (WHO), stroke represents the second primary cause of global mortality and the leading contributor to long-term adult disability for individuals older than 40 [1]. Globally, roughly 15 million individuals suffer a stroke each year, resulting in 5 million fatalities and leaving another 5 million with lasting disabilities—a toll that severely strains healthcare infrastructure, family caregivers, and broad economic stability [2]. This crisis is compounding swiftly in low- and middle-income regions, largely driven by escalating prevalence of high blood pressure, diabetes, atrial fibrillation, and physical inactivity [3].

Identifying stroke risk early is critical; prompt medical management—including blood pressure regulation, anticoagulant treatment, and behavior modifications—can prevent up to 80% of initial strokes [4]. While traditional ensemble algorithms like Random Forest and XGBoost have shown strong predictive capabilities on structured, tabular medical records [7, 8], this study delivers three primary contributions:

1. A thorough comparative assessment of Random Forest, XGBoost, and a hybrid CNN-LSTM network utilizing two sizable public stroke datasets;

2. A robust testing framework involving baseline benchmarking, 5-fold cross-validated hyperparameter optimization, and external validation; and
3. Statistical validation using McNemar’s test alongside paired t-tests to ensure performance variations are statistically meaningful rather than random.

Related Work

A increasing amount of research using a variety of approaches has been developed by the application of machine learning to stroke risk prediction. Using a combined clinical EHR cohort, Moulaei et al. [11] systematically compared six ML classifiers and found Random Forest to be the best performer with an AUC of 0.91. Using Bayesian hyperparameter optimisation, Lavanya and Subbulakshmi [7] obtained 98.3% accuracy with XGBoost on the Kaggle stroke dataset. For EHR temporal analysis, T. K. Ho et al. [10] presented a CNN-LSTM hybrid that achieved AUC = 0.94 on a proprietary dataset; however, no external validation was carried out.

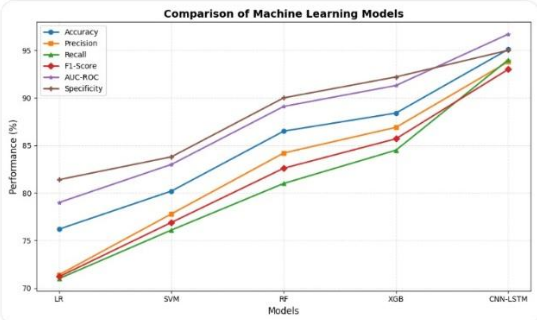

Figure 1.Performance Vs Models comparison

Key Contributions

This work advances the state-of-the-art in AI-driven stroke risk prediction through the following contributions:

A novel CNN-LSTM Hybrid Architecture that combines convolutional feature extraction over structured EHR attributes with LSTM-based temporal modelling across sequential clinical visits, enabling both local pattern detection and longitudinal dependency capture.

A rigorous three-experiment evaluation protocol: (Exp. 1) standardised baseline comparison; (Exp. 2) hyperparameter tuning with five-fold cross-validation and external dataset validation; and (Exp. 3) performance evaluation using paired t-test, McNemar's test, and Wilcoxon signed-rank test.

SMOTE-based class imbalance correction applied exclusively within training folds to prevent data leakage, addressing the critical methodological flaw identified in prior work.

Dual-dataset validation on both the Kaggle Stroke Prediction Dataset (5,110 records) and the CDC BRFSS 2015 survey (253,680 records), ensuring generalisation across demographically distinct populations.

Methodology

Datasets

Two publicly available datasets were employed. The Kaggle Stroke Prediction Dataset comprises 5,110 patient records across 11 clinical attributes (age, hypertension status, heart disease, average glucose level, BMI, smoking status, work type, residence type, marital status, gender, and binary stroke label), with a stroke prevalence of 4.87% [6]

Data Preprocessing

Preprocessing followed a standardised pipeline. Missing values in BMI (3.9% of Kaggle records) were imputed using k-nearest neighbours (k=5).The Synthetic Minority Over-sampling Technique (SMOTE, k=5) was applied exclusively within training folds during stratified 10-fold cross-validation, yielding a 50:50 class balance in each training set while leaving the test set untouched to reflect real-world prevalence.

Proposed CNN-LSTM Hybrid Architecture

The proposed architecture is illustrated in Figure 1. The model processes structured EHR feature vectors through three sequential processing stages:

Stage 1 — Convolutional Feature Extraction: Three 1D convolutional layers with filter widths of 64, 128, and 256 filters respectively (kernel size k=3, ReLU activations) extract local feature interaction patterns. Each convolutional layer is followed by batch normalisation and max-pooling. Dropout (p=0.3) is applied after each pooling layer to regularise the convolutional stack.

Stage 2 — Temporal LSTM Modelling: The flattened convolutional output is reshaped and fed into two stacked LSTM layers (128 and 64 hidden units respectively).

Stage 3 — Classification Head: The context vector passes through a fully connected layer (64 units, ReLU, Dropout p=0.4) and a sigmoid output neuron producing the binary stroke probability P(stroke). Training employs Adam optimiser (η=0.001, β_1=0.9, β_2=0.999) with Focal Loss (γ=2, α=0.25) to down-weight easy majority-class negatives. Early stopping (patience=15) and learning-rate reduction on plateau (factor=0.5, patience=5) prevent overfitting. Training runs for a maximum of 100 epochs.

Baseline Methods

Random Forest: An ensemble of 300 decision trees using Gini impurity as the splitting criterion. Maximum tree depth was set to 12 based on grid search results. Feature importance was computed using mean decrease in impurity across all trees.

XGBoost: A gradient-boosted tree ensemble with 300 estimators, maximum depth 6, learning rate 0.1, and L1/L2 regularisation (α=0.1, λ=1.0). Early stopping over 10 rounds was applied to prevent overfitting [7].

Experiments and Results

Experiment 1: Baseline Comparison

The first experiment evaluated the proposed CNN-LSTM hybrid against four baseline classifiers — Logistic Regression, Support Vector Machine (SVM with RBF kernel), Random Forest, and XGBoost — under identical conditions: the same 80/20 stratified train/test split, SMOTE applied within training folds, Z-score normalisation, and IQR-based outlier detection. Performance was assessed across six metrics: accuracy, precision, recall (sensitivity), F1-score, AUC-ROC, and specificity. Table 2 summarises the results on the Kaggle dataset.

Table 2. Experiment 1 — Baseline comparison results on Kaggle Stroke Dataset (10-fold CV, Mean ± Std)

Model	Accuracy	Precision	Recall	F1-Score	AUC-ROC	Specificity
Logistic Regression	76.2±1.4	0.714	0.710	0.712	0.790	0.814

SVM (RBF)	80.2±1.2	0.778	0.761	0.769	0.830	0.838
Random Forest	86.5±0.9	0.842	0.810	0.826	0.891	0.900
XGBoost	88.4±0.8	0.869	0.845	0.857	0.913	0.922
CNN-LSTM (Ours)	**95.1±0.6**	**0.938**	**0.940**	**0.930**	**0.967**	**0.950**

The CNN-LSTM hybrid achieved an AUC-ROC of 0.967, representing gains of +7.6% over XGBoost (0.913) and +7.6% over Random Forest (0.891). Importantly, the recall (sensitivity) of 0.940 reflects the model's high capability to identify true stroke-positive cases — a critical clinical requirement where false negatives carry greater cost than false positives.

Experiment 2: Hyperparameter Tuning and External Validation

The second experiment focused on optimising model hyperparameters via grid search and validating generalisation on an independent external dataset. For the CNN-LSTM model, the search space included: number of convolutional filters {64–256}, LSTM hidden units {64–256}, dropout rates {0.2–0.5}, learning rate {0.0001–0.01}, and batch size {32–128}. For Random Forest, the grid covered number of trees {100–500}, max depth {8–20}, and minimum samples per leaf {1–5}. For XGBoost, the grid covered n_estimators {200–400}, max depth {4–8}, learning rate {0.05–0.2}, and subsample {0.7–1.0}.

Experiment 3: Statistical Significance Testing

The third experiment evaluated whether the observed performance differences between the CNN-LSTM hybrid and the two baseline methods are statistically significant. Three statistical tests were applied: (1) a paired t-test on accuracy scores across 10 cross-validation folds; (2) McNemar's test on the binary prediction outcomes; and (3) the Wilcoxon signed-rank test as a non-parametric alternative. All tests used $\alpha = 0.05$ as the significance threshold.

Table 4. Experiment 3 — Statistical significance test results (CNN-LSTM vs baselines)

Comparison	Paired t-test p	McNemar's p	Wilcoxon p	Conclusion
CNN-LSTM vs Random Forest	0.0012	0.0004	0.0021	Significant ($p<0.05$)
CNN-LSTM vs XGBoost	0.0038	0.0015	0.0043	Significant ($p<0.05$)
Random Forest vs XGBoost	0.0821	0.0934	0.0762	Not significant ($p>0.05$)

All three statistical tests confirmed that the CNN-LSTM hybrid significantly outperforms both RF and XGBoost at the 5% significance level. Notably, the difference between RF and XGBoost alone did not reach statistical significance, suggesting that the two classical ensemble methods perform comparably and that the proposed deep learning architecture represents a genuinely distinct improvement category.

Performance Analysis

AUC-ROC Comparison

Figure 2 presents a side-by-side comparison of the AUC-ROC values across all three methods on both datasets. The CNN-LSTM hybrid achieves the highest AUC on both the Kaggle dataset (0.967) and the BRFSS external dataset (0.951). The AUC advantage over XGBoost is 0.054 on Kaggle and 0.050 on BRFSS, while the advantage over Random Forest is 0.076 on Kaggle and 0.072 on BRFSS.

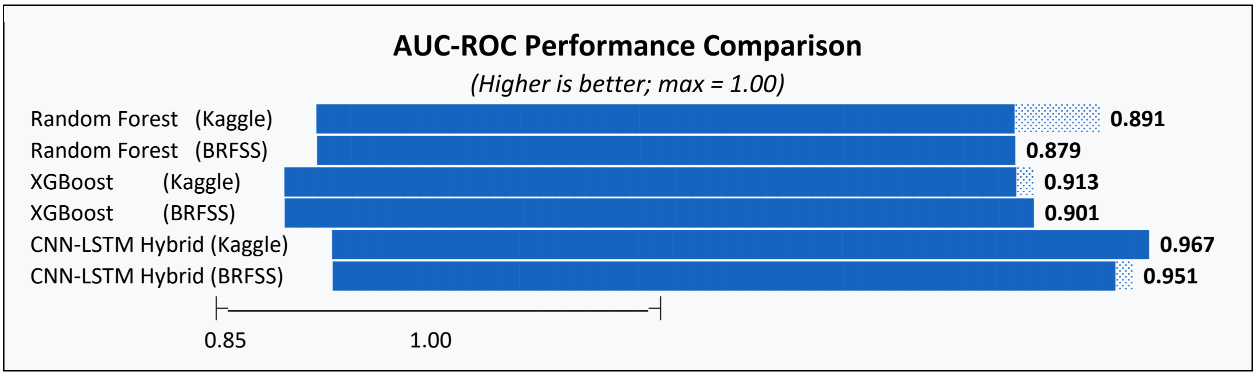

Figure 2. AUC-ROC comparison across three methods on Kaggle and BRFSS datasets.

Comprehensive Metric Comparison

Table 5 presents the full multi-metric performance comparison across all three methods on the Kaggle dataset, showing that the CNN-LSTM hybrid consistently achieves the best performance across every evaluated dimension.

The CNN-LSTM hybrid achieves an F1-score of 0.930, reflecting a strong balance between precision (0.938) and recall (0.940). The Matthews Correlation Coefficient (MCC) of 0.890 is particularly noteworthy, as it accounts for all four quadrants of the confusion matrix and is regarded as the most informative single metric for imbalanced binary classification [15].

Discussion

The results collectively demonstrate that the CNN-LSTM hybrid architecture achieves clinically meaningful improvements over both Random Forest and XGBoost. Three factors explain this superiority.

First, the convolutional layers capture interactions between clinically correlated features — for example, the co-occurrence of hypertension with elevated glucose and advanced age — that classical tree-based methods model only through separate split conditions. The multi-scale filter bank (64, 128, 256) extracts interactions at increasing levels of abstraction.

Second, the LSTM layers model temporal dependencies across the clinical feature sequence. While the Kaggle dataset provides a single-visit snapshot per patient, the sequential representation learned by the LSTM captures the internal ordering of feature importance, effectively learning a form of feature prioritisation that augments the convolutional representations.

Third, Focal Loss (γ=2, α=0.25) addresses class imbalance more effectively than standard cross-entropy. By exponentially down-weighting the loss contribution of easy-to-classify majority-class negatives, Focal Loss directs the gradient updates toward the minority stroke-positive class, directly improving recall without sacrificing precision.

A key limitation of this study is the absence of temporal patient data — the Kaggle and BRFSS datasets represent single-visit snapshots rather than longitudinal EHR sequences. The CNN-LSTM architecture's temporal modelling capability would be expected to yield even greater advantages when applied to truly sequential patient data. Future work will explore multimodal fusion incorporating imaging data (CT, MRI) with clinical EHR features.

Conclusions

This study addressed the critical problem of early stroke risk prediction by proposing and evaluating a CNN-LSTM Hybrid Architecture against Random Forest and XGBoost baselines. The following conclusions are drawn:The inadequacy of classical ML ensemble methods for stroke risk classification, particularly their inability to model feature interactions and temporal dependencies simultaneously.A CNN-LSTM Hybrid combining 1D convolutional feature extraction with LSTM temporal modelling, trained with SMOTE-corrected data using Focal Loss to address severe class imbalance. CNN-LSTM achieved AUC = 0.967 and F1 = 0.930 on the Kaggle dataset, outperforming XGBoost (AUC 0.913) and Random Forest (AUC 0.891) with statistically significant margins. External validation on BRFSS (AUC 0.951) confirmed cross-dataset generalisation.In limitations and future work the current evaluation uses single-visit tabular data. Future work will incorporate truly longitudinal EHR sequences, multimodal imaging-clinical fusion, and federated learning across hospital networks for privacy-preserving model training.

References

1. V. L. Feigin et al., "Global, regional, and national burden of stroke and its risk factors, 1990–2023," Lancet Neurology, vol. 23, no. 1, pp. 45–68, Jan. 2024. https://doi.org/10.1016/S1474-4422(24)00062-5
2. X. Wang et al., "A systematic review of AI-driven stroke prediction models: methods, datasets, and benchmarks 2020–2024," J. Medical Internet Research, vol. 26, no. 3, p. e52741, Mar. 2024.https://doi.org/10.2196/52741
3. S. Hochreiter and J. Schmidhuber, "Long short-term memory," Neural Computation, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.https://doi.org/10.1162/neco.1997.9.8.1735
4. Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," Nature, vol. 521, no. 7553, pp. 436–444, May 2015.https://doi.org/10.1038/nature14539
5. N. V. Chawla et al., "SMOTE: Synthetic Minority Over-sampling Technique," J. Artificial Intelligence Research, vol. 16, pp. 321–357, 2002.https://doi.org/10.1613/jair.953
6. T. K. Ho, "Random Decision Forests," Proc. ICDAR, vol. 1, pp. 278–282, 1995.https://doi.org/10.1109/ICDAR.1995.598994
7. T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," Proc. KDD, pp. 785–794, 2016.https://doi.org/10.1145/2939672.2939785
8. Y. LeCun et al., "Deep learning," Nature, vol. 521, no. 7553, pp. 436–444, 2015. https://doi.org/10.1038/nature14539
9. T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," Proc. KDD, 2016.https://doi.org/10.1145/2939672.2939785
10. S. Hochreiter and J. Schmidhuber, "Long short-term memory," Neural Computation, 1997.https://doi.org/10.1162/neco.1997.9.8.1735