EAI Endorsed Transactions
on Scalable Information Systems

Research Article EAI.EU

Fusing Attention and Convolution: A Hybrid Model for Brain Stroke Prediction

R. Bhuvanya[1,*], T. Kujani[2] and K. Sivakumar[3]

[1]Department of Computer Science and Engineering, Sri Ramachandra Faculty of Engineering and Technology, Sri Ramachandra Institute of Higher Education and Research, Chennai-600016, Tamil Nadu, India
[2]Department of Computer Science and Engineering, Vel Tech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology, Chennai-600062, Tamil Nadu, India
[3]Department of Information Technology, Nehru Institute of Engineering and Technology, Coimbatore

Abstract

INTRODUCTION: A stroke, a sudden interruption of blood flow to the brain, is a leading cause of disability and death. Early diagnosis is paramount for minimizing brain damage and maximizing treatment effectiveness.

OBJECTIVES: Traditional diagnostic methods can be time-consuming and have limited Accuracy.

METHODS: This study investigates the efficacy of various machine-learning models for stroke prediction. Specifically, it compares established models like K-Nearest Neighbor, Artificial Neural Network, Long Short Term Memory (LSTM), and stacked LSTM with a newly proposed Transformer Convolutional Neural Network (TCNN) architecture, which fuses Transformer and Convolutional neural network (CNN) models.

RESULTS: The TCNN demonstrates significant promise, achieving a superior accuracy of 98% when optimized with the AMSGrad optimizer.

CONCLUSION: These findings suggest that the TCNN architecture has the potential to revolutionize stroke prediction accuracy compared to existing methods, potentially leading to improved patient outcomes.

Keywords: AdaGrad, Brain Stroke detection, CNN, Machine Learning, Transformer

Received on 21 08 2024, accepted on 05 12 2024, published on 11 12 2024

doi: 10.4108/eetsis.7022

*Corresponding author. Email: bhuvanyaraghunathan@gmail.com; bhuvanyaraghunathan12@outlook.com

1. Introduction

Stroke is considered one of the most dangerous and fatal conditions affecting humans because it occurs suddenly in the brain. This happens when the brain's blood supply is cut off, impairing the brain's capacity to take in oxygen and nutrients [1]. Brain cell death, therefore, happens in a matter of minutes. The World Health Organisation (WHO) lists ischaemic heart disease as the primary cause of mortality worldwide, with stroke coming in second. A poor diet, smoking, high blood pressure, high cholesterol, and inactivity are the main causes of brain stroke. For successful brain stroke treatment and recovery, early detection is crucial. This approach increases the likelihood of successful intervention, minimizes brain damage, improves recovery outcomes, reduces mortality rates, lowers healthcare costs, and helps prevent future stroke. Hence, this study focuses on predicting brain stroke by incorporating a novel approach, namely TCNN, by fusing Transformer and Convolution. Figure 1 presents the computational framework for stroke risk assessment.

The proposed method creates a hybrid model that combines the best features of transformers and convolutional neural networks (CNNs) to offer a novel approach to brain stroke prediction [2]. By using CNNs to capture fine-grained spatial information and the

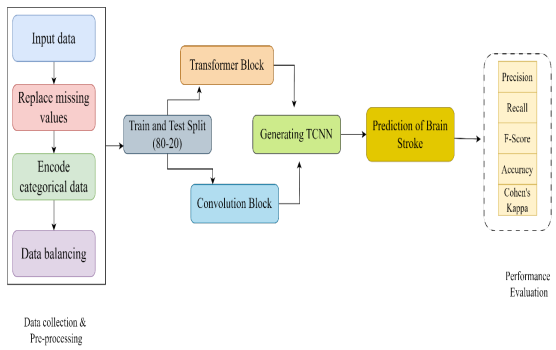

Figure 1. Computational framework for stroke risk assessment

Transformer's multi-head attention mechanism to identify long-range correlations within the data, this novel architecture enables the model to capture both [3]. This combination significantly improves stroke prediction accuracy compared with traditional methods.

The main insights this study offers are summed up as follows:

1. The proposed method uses a multi-head attention mechanism and a convolution process to analyze a brain stroke dataset and identify strokes more accurately.
2. This paper examines the performance of the hybrid transformer-CNN model in further detail by investigating the effects of several optimization strategies.
3. The suggested model provides a comparative examination of the machine learning classifiers. This comparison analysis sheds light on the suggested strategy's efficacy and any possible benefits over currently used stroke prediction techniques.

The structure of this document is as follows. Section 1 summarises the introduction, outlining the significance of this research area. Section 2 delves into the existing frameworks for brain stroke prediction and establishes the groundwork for the proposed approach. Section 3 constitutes the core of this paper, introducing the proposed TCNN model designed to enhance stroke prediction accuracy. Subsequently, the experiment findings are shown in Section 4, where they are used to assess the viability of the suggested model. Section 5, which summarises the main conclusions and suggests possible avenues for further study, brings the work to a close.

2. Literature Review

Bathla and Kumar [4] combined feature selection techniques with machine learning classifiers to create a hybrid system that can accurately predict brain strokes. To alleviate the class disparity, the Synthetic Minority Over-Sampling Technique (SMOTE) was utilized. The researchers compared the performance of five classifiers (Naive Bayes, SVM, Random Forest, AdaBoost, and XGBoost) and three feature selection methods (Pearson Correlation, Mutual Information, and Feature Importance). The Random Forest classifier with the Feature Importance feature selection method yielded the highest Accuracy of 97.17%, reducing the feature set by 36.3%. They plan to incorporate Computed Tomography (CT) and Magnetic resonance imaging (MRI) images for classification in future work. Choi et al. [5] utilized electroencephalogram (EEG) data from elderly Korean participants aged 65 or older to predict stroke disease using deep learning models. They evaluated the models, including CNN-LSTM, LSTM, Bidirectional LSTM, and CNN-Bidirectional LSTM, with the latter achieving the highest Accuracy of 94.0%. To predict strokes using unbalanced healthcare data, Dev et al. [6] offer research on an Artificial Bee Colony (ABC) optimized Deep Neural Network (DNN) model. This work identified a gap in feature selection mechanisms for stroke prediction and demonstrates that incorporating feature selection can significantly enhance prediction accuracy. With accuracy, precision, and recall rates of 87.09%, 84.28%, and 85.72%, respectively, the suggested ABC-FS-optimized DNN model performs better than alternative machine learning approaches. To overcome the issues of incomplete data and class imbalance, Liu et al. [7] describe a hybrid machine-learning strategy for predicting cerebral strokes utilizing a dataset of 43,400 patient records with 783 stroke events. The authors propose an automatic hyperparameter optimization (AutoHPO) based on a deep neural network (DNN) as a two-step technique that uses random forest regression to impute missing values. The study by Peñafiel et al. [8] presents a predictive model for stroke risk using an Electronic Health Record, focusing on interpretability and handling missing data. The model utilizes a Dempster-Shafer theory-based approach and outperforms other machine learning methods, especially with incomplete data. The model extracts and validates important rules for stroke prediction and identifies key factors such as past cerebrovascular disease, high haemoglobin levels, diabetes, and body fat as significant predictors of stroke risk. Rahman et al. [9] compared the effectiveness of different machine learning (ML) algorithms and deep neural networks (DNNs) in predicting strokes. This study assessed many models with varying architectures, including Random Forests, Support Vector Machines (SVM), and Deep Neural Networks. Regarding Accuracy, Random Forest performs better than other classifiers (0.99). Plans for future study entail expanding the dataset or utilizing the same model on several other datasets. A comparative comparison of machine learning methods for stroke prediction is presented by Sailasya et al. [10], with Naive Bayes demonstrating the best Accuracy. The study advances medical data analysis and shows how machine learning (ML) may be used to forecast serious illnesses like stroke. The models' training on textual data as opposed to real-time brain scans is a limitation of the work, and it is suggested that the research be expanded by utilizing all available machine learning methods. Adaptive moment estimates with maximum (AdaMax), Root Mean Squared Propagation (RMSProp), and Adaptive learning rate method (Adadelta) are three multi-optimizers that Uppal et al. [11] employed as a classification methodology for stroke data. According to the trial, the RMSProp optimizer is the most effective, with a data training accuracy of 95.8% and a data testing accuracy of 94.9%. Around 95% accuracy was attained, which may not be adequate for important medical diagnoses. Additionally, all of the data utilized in this work are textual; however, compiling a CT scan dataset to predict stroke risk in the future may be more useful. The incorporation of IoT technologies into healthcare systems has significantly enhanced patient care. Naresh [12] investigates the application of Discrete Wavelet Transform (DWT) for processing ECG signals within an IoT-enabled health monitoring system. Leveraging DWT's time-frequency localization features, the research ensures effective analysis of non-stationary signals. The system architecture comprises signal acquisition, preprocessing, feature extraction, and real-time IoT-based transmission to cloud servers. Performance metrics indicate notable improvements in signal clarity and data compression. Basava [13] designed Smart Comrade Robot, driven by AI, to improve elderly care by integrating robotics and artificial intelligence for daily support, health monitoring, and emergency assistance. It enhances safety, provides companionship, and alleviates caregiver stress through features like real-time health tracking, fall detection, and emergency alerts. Leveraging advanced technologies such as IBM Watson Health and Google Cloud AI, it delivers personalized care to enhance the quality of life for older adults. Artificial intelligence (AI) transforms radiology by enhancing diagnostic Accuracy and efficiency. Tools like Convolutional Neural Networks (CNNs) and Variational Autoencoders (VAEs) are used by Surendar Rama Sitaraman [14] to support radiologists by streamlining image analysis, automating data processing, and detecting abnormalities. VAEs also generate synthetic medical images, aiding in data augmentation and privacy protection. However, the widespread adoption of AI faces challenges such as the need for large annotated datasets, issues with model interpretability, and ethical concerns. Despite these obstacles, AI holds significant potential to improve patient outcomes in the future.

3. Proposed Method

The proposed method combines the integration of transformers and convolutional neural networks (T-CNN) [15] in a unified architecture for brain stroke prediction. The Transformer component learns contextual relationships in the medical data, and the CNN component extracts relevant features from the input data [16]. The proposed T-CNN model uses self-attention mechanisms to prioritize important information and convolutional filters to capture the spatial patterns in the data. Combining these two architectures, this novel method can provide comprehensive and accurate predictions of brain stroke risk. Figure 2 presents a functional overview of the proposed TCNN architecture.

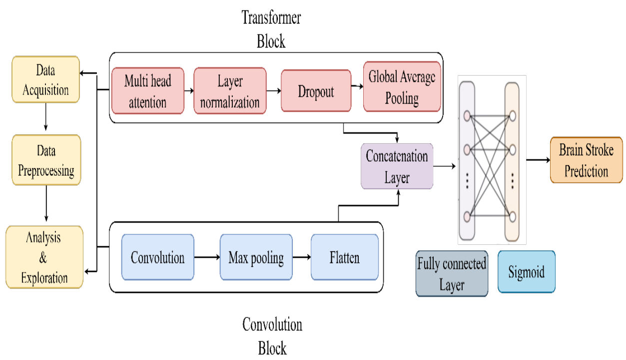

Figure 2. Functional overview of the proposed framework

Transformers are a type of deep learning architecture that has shown promise in various applications. Unlike traditional methods, transformers can seamlessly integrate information from diverse sources, leading to a richer understanding of stroke risk. By combining multiple data types, transformers have the potential to make more accurate predictions compared to models relying on a single source [17]. The Transformer model for learning contextual relationships is described in Eq. (1)
Given an input sequence of vectors,

$$X = \{x_1, x_2, ..., x_n\} \quad (1)$$

Where (x_i) represents the embedding vector of the i^{th} input token, Feedforward neural networks (FFN) with several layers of self-attention make up the Transformer [18]. Positioning codes are appended to the input embedded data to deliver details regarding the relative locations of tokens, as the Transformer does not record the sequence order by default. Eq. (2) and (3) define the positional encoding for the i^{th} position.

$$PE(i,2j) = \sin (i/10000^{2j/d}) \quad (2)$$

$$PE(i,2j+1) = \cos(i/10000^{2j/d}) \quad (3)$$

Where d is the dimension of the embeddings, and j ranges from 0 to d/2−1. The input embeddings are combined with the positional encodings described in Eq. (4).

$$Z_i=e_i+PE_i \quad (4)$$

The self-attention layer is a transformer's central component [19]. Focusing on pertinent segments of the input sequence enables the model to identify interdependencies among its constituents. The self-attention mechanism converts the attention scores into probabilities using the softmax function [20]. These probabilities determine how much focus each token in the sequence should receive relative to the others. The formulation is mentioned in Eq. (5) as follows:

$$\text{Attention}(Q, K, V) = \text{SoftMax}(QK^T/\sqrt{d_k})\,V \quad (5)$$

where,
$Q=ZW_Q$ (Query matrix), $K=ZW_K$ (Key matrix), $V=ZW_V$ (Value matrix), and W_Q, W_K, W_V are learned parameter matrices. The outputs are routed via layer normalization and residual connections via the Feed Forward Network (FFN). Eq. (6) represents the FFN.

$$FFN(x)=\max(0, xW_1+b_1)\,W_2+b_2 \quad (6)$$

Here, x represents the input, and W,b denotes the weights and bias. Through the introduction of non-linearity and the ability to learn complicated feature representations, the FFN improves the model's capacity to detect intricate patterns and correlations in the data. In the Transformer architecture, each sub-layer is designed to stabilize and enhance the training process using layer normalization and residual connections. This combination is crucial for the model's performance and training efficiency. Thus, the Transformer encoder block processes the input word embeddings, capturing relationships between words [21]. Its final output vector, hT, represents the entire sentence. Eq. (7) means the shape of the final output vector.

$$hT.shape=(batch\_size,embedding\_dim) \quad (7)$$

In this deep learning architecture designed for stroke prediction, a convolutional neural network (CNN) branch operates parallel to the transformer encoder block, offering a complementary analytical pathway for the input data. The CNN branch operates on word embeddings to extract localized features. After convolution and pooling, the CNN's output is flattened into a 1D vector (h_C). This vector mentioned in Eq.(8) encapsulates the features learned by the CNN filters. Mathematically, its dimension can be represented as,

$$hC.shape = (batch\_size, flattened\_size) \quad (8)$$

Flattened_size depends on the filter and pooling operations used in the CNN architecture. This fusion layer strategically combines the Transformer and CNN outputs using concatenation along a specific axis. As mentioned in Eq.(9), the tf. concat function performs this operation, joining the vectors side-by-side.

$$h = tf.concat([hT, hC], axis=1) \quad (9)$$

This results in a new vector (h) with a dimension of batch size, embedding_dim, and flattened_size as denoted in Eq.(10)

$$h.shape=(batch\_size,embedding\_dim+flattened\_size) \quad (10)$$

CNN's capacity to extract localized characteristics from sequential input is one of its strongest points. By applying 1D convolutional, CNN can identify localized aspects crucial for the classification task. Crucially, the information extracted by the CNN is strategically integrated with the global understanding captured by the Transformer. The CNN procedure is followed by flattening the output and concatenating it with the vector produced by the global average pooling layer of the Transformer. This creates a combined representation that incorporates information from global and local analysis levels. This fused representation is input into fully linked layers with sigmoid activation for the last classification challenge. Table 1 illustrates the algorithm of TCNN.

Table 1. Algorithm of proposed TCNN

TCNN- Fusion of Transformer and CNN
Input: Kaggle Stroke dataset, num_filters, kernel_size, hidden_units
Output: Predicted Stroke Status (logits= Normal, Stroke)

```
Begin
    Data= load dataset ( )
        if data contains empty values
            replace missing values with appropriate values
        end if
        if column in the data has object type
            encode categorical data
        end if
        If data is imbalanced:
            balance=SMOTE(strategy='minority')
        end if
        X = data without 'stroke', y = data['stroke']
        [X1, X2, y1, y2] = split_data(X, y)
            scaler = StandardScaler()
        if standardize:
            scaled_data = scaler.fit_transform(data)
        end if
    Reshape the data for CNN
    for each sample X:
        transformer_output =TransformerBlock(X)
```

```
cnn_output = Flatten(MaxPooling1D(Conv1D(X,num_filters, kernel_size)))
fused_features= concat([transformer_output,cnn_output],axis=1)
logits=Dense(fused_features,hidden_units, activation='relu', num_classes)
Compute performance evaluation metrics
End
```

4. Performance Analysis

4.1. Dataset Description

The dataset was extracted from Kaggle [22]. From the obtained dataset the independent attributes are id, gender, heart_disease, age, bmi, ever_married, work_type, Hypertension, residence_type, avg_glucose_level, smoking_status, and the stroke column is considered as a dependent attribute. A stroke is indicated by several 1, indicating that there has been no stroke. The influence of specific feature values on brain stroke prediction is visualized using the comprehensible Artificial framework SHAP (Shapley Additive Explanations) to obtain knowledge about the feature relevance and model comprehension.

4.2. Data Preprocessing

This dataset contains 12 attributes. This work excluded the id column as it doesn't influence the outcome. Missing values in bmi and smoking_status were imputed, followed by label encoding of categorical features. Finally, data normalization was applied to numeric features, scaling them to a common range between 0 and 1. This ensures all features contribute equally to the model, avoiding biases from different measurement scales.

Machine learning models may perform noticeably worse when the dataset has a class imbalance. The proposed study used the Synthetic Minority Over-Sampling Technique (SMOTE) [23], which deliberately duplicates data points from the minority class to increase its representation in the training set. Figure 3 (a) and (b) illustrate the diagrammatic illustration of the samples before and after applying SMOTE.

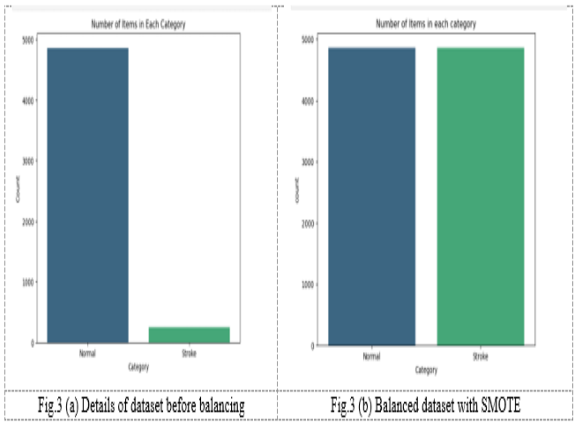

Figure 3. (a) Details of dataset before balancing
Figure 3. (b) Balanced dataset with SMOTE

4.3. Exploratory Data Analysis

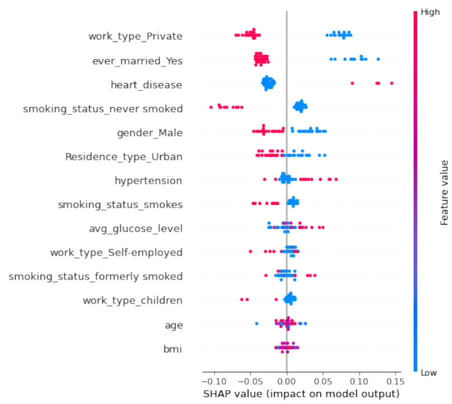

Figure 4. Model interpretability with SHAP summary plot

Figure 4 shows the model interpretability using a SHAP summary graphic. This graph, a SHAP summary plot, illustrates how different characteristics affect a machine learning model's output, particularly when forecasting stroke outcomes. By calculating the contribution of each attribute to the prediction, this technique explains individual forecasts. The SHAP values, which show how each attribute affects the prediction, are represented on the x-axis. A characteristic with a positive SHAP value raises the possibility of the positive class (having a stroke, for example).

In contrast, a negative SHAP value indicates a drop in the chance. The model's characteristics are listed on the y-axis. Each dot represents a sample's SHAP value. The dot's colour represents the feature's initial value. Higher feature values are indicated by red, and lower feature values are indicated by blue.

Age, Hypertension, and average glucose level are well-established risk factors for stroke. The SHAP summary plot analysis indicates that age is a heavily weighted feature in the model's stroke risk prediction. As a result, the model predicts that older people will have a greater risk of stroke. Another significant risk factor for stroke is persistently elevated blood pressure. The SHAP summary plot suggests the model incorporates hypertension and blood glucose levels as important predictors of stroke risk. Higher values of these features correspond to an increased predicted risk of stroke, which aligns with established medical knowledge.

4.4. Evaluation Metrics

Performance evaluation metrics are essential for determining how well machine learning models work. Therefore, the evaluation tools are thought to include precision, recall, F1 score, Accuracy, and kappa coefficient, also referred to as Cohen's kappa. The precision of the model is determined by dividing all of its positive predictions by the percentage of genuine positive forecasts. [24] Eq.(11) shows the formula to calculate precision.

Precision=True_P/True_P+False_P (11)

Recall measures the percentage of accurate positive predictions among all real positive events in the dataset. It is computed by comparing the ratio of true positives to the total of false negatives and true positives. The formula for calculating recall is mentioned in Eq.(12).

Recall=True_P/ True_P +False_N (12)

Where True_P, False_P, False_N denote the True Positive, False Positive and False Negative, respectively.'
F1 offers a single score that considers recall and Accuracy, which is especially helpful for unbalanced datasets. The F1 score calculation algorithm is provided in Eq. (13) below.

F1=2(Precision×Recall)/(Precision+Recall) (13)

The ratio of accurately predicted occurrences to all instances in the dataset is used to calculate Accuracy, which gauges the model's overall correctness. q. (14) shows the formula to calculate Accuracy.

Accuracy =Accurate predictions/(Total no.of predictions) (14)

Inter-rater agreement for categorical items is statistically measured using Cohen's kappa, which considers chance-based agreement. t provides a more robust evaluation of classifier performance than simple Accuracy, particularly in cases of class imbalance. The formula used to calculate Cohen's kappa is mentioned in Eq. (15).

Cohen's Kappa = (Observed- Expected)/(1-expected) (15)

The suggested method's performance is assessed using the abovementioned performance evaluation metrics. Additionally, the proposed approach is contrasted with baseline machine models, including stacked LSTM with two layers, K-Nearest Neighbour, Artificial Neural Network, and Long Short-Term Memory (LSTM).

4.5. Results and Discussion

Using the "Adam" optimizer function, the suggested TCNN network is implemented in the brain stroke prediction datasets over a range of epoch counts. Table 2 below summarizes the performance evaluation across different numbers of epochs.

Table 2. Performance evaluation across different numbers of epochs

S.no	No. of epochs	Precision		Recall		F1 Score	
		Normal	Stroke	Normal	Stroke	Normal	Stroke
1	10	0.88	0.95	0.95	0.86	0.91	0.91
2	25	0.93	0.92	0.93	0.92	0.93	0.92
3	35	0.95	0.87	0.85	0.95	0.90	0.91
4	50	0.93	0.94	0.94	0.92	0.93	0.93

Figure 5 shows the correlation between the quantity of training epochs and the model's accuracy and kappa coefficient.

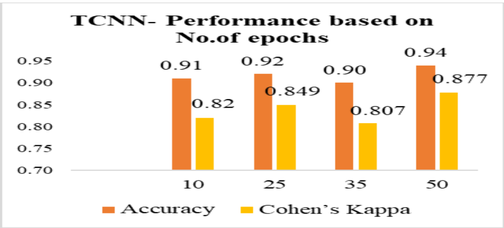

Figure 5. Performance of TCNN with varied epochs

Further enhancing the results, the TCNN model is evaluated with various optimizer functions by choosing 50 epochs. Table 3 illustrates the results of precision, F-measure, recall, Accuracy, Cohen's kappa, and the applied optimizer function.

Table 3. Impact of different optimizers on TCNN

S.no	Optimizers	Precision		Recall		F1 Score		Accuracy	Cohen's Kappa
		Normal	Stroke	Normal	Stroke	Normal	Stroke		
1.	Adam	0.93	0.94	0.94	0.92	0.93	0.93	0.93	0.866
2.	Adagrad	0.84	0.80	0.79	0.85	0.81	0.82	0.81	0.631
3.	Adamax	0.88	0.98	0.98	0.87	0.93	0.92	0.93	0.851
4.	**AMSGrad**	**0.96**	**0.98**	**0.98**	**0.96**	**0.97**	**0.97**	**0.98**	**0.910**
5.	Nadam	0.93	0.94	0.94	0.93	0.94	0.93	0.94	0.870
6.	Adadelta	0.69	0.79	0.83	0.63	0.76	0.70	0.73	0.758
7.	Nesterov Accelerated Gradient	0.84	0.99	0.99	0.81	0.91	0.89	0.90	0.800
8.	RMSprop	0.93	0.93	0.93	0.93	0.93	0.93	0.93	0.857
9.	Stochastic Gradient Descent	0.91	0.93	0.93	0.91	0.92	0.92	0.92	0.843

The above table shows that the proposed TCNN model with AMSGrad optimizer performs well and achieves 98% accuracy and 91% kappa score. A high kappa score indicates better performance of the model. Hence, the TCNN model with AMSGrad optimizer was chosen for further evaluation. Accuracy and loss functions act as supplementary metrics for assessing the performance of the methodologies discussed above. Figures 6(a) and (b) depict the graphical representation of the TCNN model (with 50 epochs), offering insights into both the overall Accuracy of the models and the progression of the loss function during training and testing.

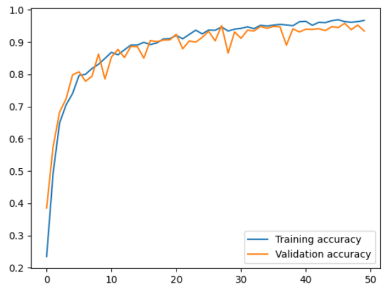

Figure 6. a) Accuracy over Training and Validation

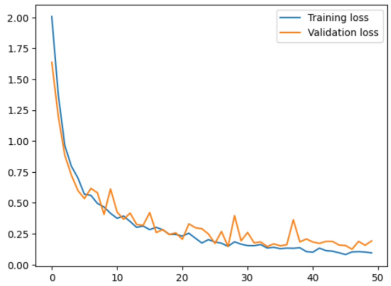

Figure 6. (b) Loss over Training and Validation

Figure 7 further illustrates the suggested strategy's performance using a gain chart, also called a lift chart. It is a visual tool for assessing how well a classification model performs, especially when dealing with binary classification issues. The chart plots the proportion of positive instances against the proportion of the population targeted by the model.

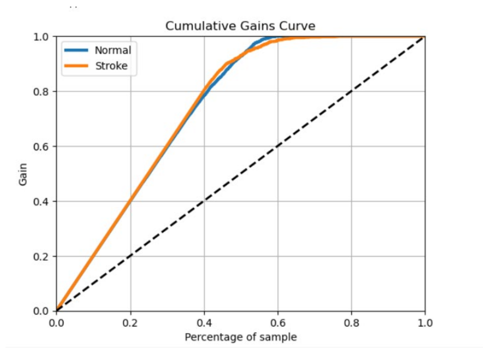

Figure 7. Lift chart

The proposed model curve starts at (0,0) and rises steeply to (100,100), indicating that it correctly identifies all positive instances with minimal population targeting. Further, to visualize the spread of probabilities assigned by a classification model to each class label, a predicted probability graph is depicted in Figure 8. t typically shows the frequency or density of predicted probabilities across possible values. In addition to offering insights about the model's performance and calibration, this graph is crucial for comprehending the degree of confidence in the predictions made. The well-calibrated model would ideally have a smooth and evenly distributed curve across the entire range of probabilities, indicating balanced confidence in forecasts.

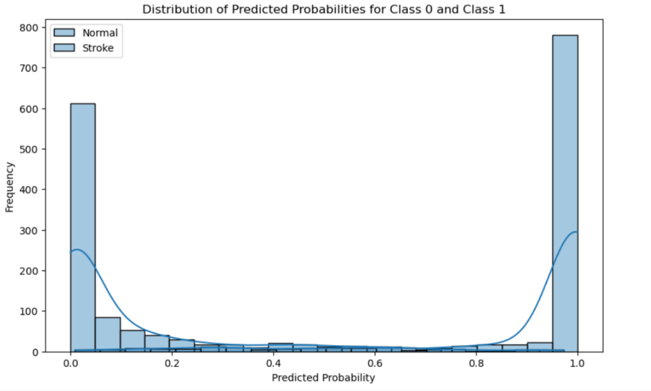

Figure 8. Histogram of Predicted Class Probabilities

The model distinguishes well between the two classes for most samples, with high-confidence predictions for both normal and stroke. However, there is a chance for improvement in reducing the number of less confident predictions in the middle range, which will be managed in the future by tuning the model, gathering more data, and trying different modelling approaches. Additionally, the suggested TCNN is contrasted with the current models, which include stacked LSTM, KNN, ANN, and LSTM, in that order. Area Under the Receiver Operating Characteristic Curve (ROC-AUC) curves illustrate how well the suggested model and all others already in use perform. The graphical depiction of ROC-AUC is illustrated in Figure 9.

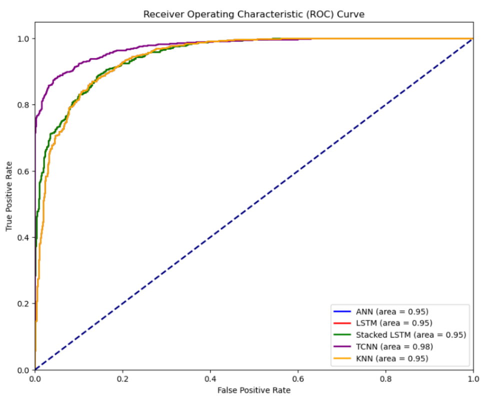

Figure 9. Performance Evaluation: ROC Curves of Multiple Models

Also, the following graphs, 10 and 11, depict the pictorial representation of Accuracy and Cohen's kappa for all the adopted models.

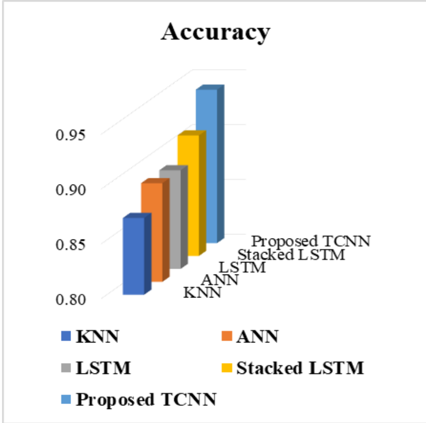

Figure 10. Graphical depiction of Accuracy

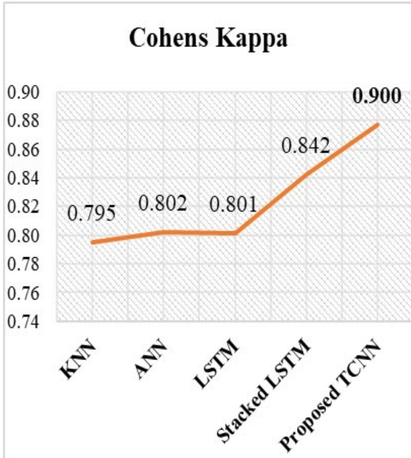

Figure 11. Cohen's kappa scores

The above results show that the TCNN model with AMSGrad optimizer outperforms all existing techniques and achieves 98% accuracy. This paper compared several machine learning algorithms, including K-Nearest Neighbors (KNN), Long Short-Term Memory (LSTM) networks, Artificial Neural Networks (ANNs), and stacked LSTMs, for their effectiveness in brain stroke prediction. The investigation revealed that a proposed TCNN architecture, optimized with the AMSGrad optimizer, achieved superior results. It seems TCNNs are just really good at picking up on the important patterns in the data over time, which helps predict strokes better.

5. Conclusion and Future Enhancements

The mitigation of brain damage and enhancement of patient outcomes are contingent upon the timely identification of stroke. Every minute counts when it comes to stroke intervention, and accurate prediction models can be instrumental in getting patients the help they need as soon as possible. The finding highlights the potential of TCNNs to identify stroke-related patterns in data, leading to faster diagnoses. The multi-head attention mechanism captures these long-range relationships well. t can learn relationships between different parts of the data, potentially identifying subtle stroke symptoms that CNN might miss. This improves the model's ability to identify strokes, even manifesting in less typical or widespread patterns. Plans include adding more complete patient data and refining the model to achieve even higher precision, which should result in better approaches to stroke treatment and prevention.

Declarations

Funding: The authors did not receive any funding. Conflicts of Interests: Authors do not have any conflicts. Data Availability Statement: No datasets were generated or analyzed during the current study.
Code availability: Not Applicable.
Authors' Contributions: R. Bhuvanya, T. Kujani, is responsible for designing the framework, analyzing the performance, validating the results, and writing the article. . Sivakumar, is responsible for collecting the information required for the framework, provision of software, critical review, and administering the process.

References

[1] Dritsas E, Trigka M. Stroke risk prediction with machine learning techniques. ensors. 022; 22(13):4670.
[2] Luan Y, Lin S, "Research on Text Classification Based on CNN and LSTM," 2019 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA); Dalian, China, 2019, pp. 352-355, doi: 10.1109/ICAICA.2019.8873454.
[3] Wolf T, Debut L, Sanh V, Chaumond J, Delangue C, Moi A, Cistac P, Rault T, Louf R, Funtowicz M. Davison J. Transformers: State-of-the-art natural language processing. n Proceedings of the 2020 conference on empirical methods in natural language processing: system demonstrations, 2020. 38-45.
[4] Bathla P, Kumar R. A hybrid system to predict brain stroke using a combined feature selection and classifier. Intelligent Medicine. 024; 4(02):75-82.
[5] Choi Y A, Park S J, Jun J A, Pyo C S, Cho K H, Lee H S, Yu J H. Deep learning-based stroke disease prediction system using real-time biosignals. ensors. 021; 21(13):4269.
[6] Dev A, Malik S K. Artificial bee colony optimized deep neural network model for handling imbalanced stroke data: ABC-DNN for prediction of stroke. International Journal of E-Health and Medical Communications (IJEHMC). 021;12(5):67-83.
[7] Liu T, Fan W, Wu C. A hybrid machine learning approach to cerebral stroke prediction based on an imbalanced medical dataset. Artificial intelligence in medicine. 019; 101:101723.
[8] Penafiel S, Baloian N, Sanson H, Pino J A. Predicting stroke risk with an interpretable classifier. EEE Access. 020; 9:1154-1166.
[9] Rahman S, Hasan M, Sarkar A K. Prediction of brain stroke using machine learning algorithms and deep neural network techniques. European Journal of Electrical Engineering and Computer Science. 023; 7(1):23-30.
[10] Sailasya, Gangavarapu, Gorli L Aruna Kumari. Analyzing the Performance of Stroke Prediction using ML Classification Algorithms. 2021).
[11] Uppal M, Gupta D, Juneja S, Gadekallu T R, El Bayoumy I, Hussain J, Lee S W. Enhancing accuracy in brain stroke detection: Multi-layer perceptron with Adadelta, RMSProp and AdaMax optimizers. ront. ioeng. Biotechnol. 023; 11:1257591. doi: 10.3389/fbioe.2023.1257591.
[12] Naresh, K.R.P. Applying Discrete Wavelet Transform for ECG Signal Analysis in IOT Health Monitoring Systems. nternational Journal of Information Technology & Computer Engineering, 2022, 10(4), ISSN 2347–3657.
[13] Basava, R.G. AI-powered smart comrade robot for elderly healthcare with an integrated emergency rescue system. World Journal of Advanced Engineering Technology and Sciences, 2021, 02(01), 122–131.
[14] Surendar Rama Sitaraman. Implementing AI Applications in Radiology: Hindering and Facilitating Factors of Convolutional Neural Networks (CNNs) and Variational Autoencoders (VAEs). Journal of Science and Technology, 2022, 7 (10), ISSN:2456-5660.
[15] Chen Y, Chen X, Xu A, et al. A hybrid CNN-Transformer model for ozone concentration prediction. ir Qual Atmos Health. 022; 15(9):1533–1546.
[16] Wu N, Green B, Ben X, et al. Deep transformer models for time series forecasting: the influenza prevalence case. rXiv:200108317. 2020.
[17] Tianyang Lin, Yuxin Wang, Xiangyang Liu, Xipeng Qiu. A survey of transformers, AI Open, 2022; 3:111-132. ttps://doi.org/10.1016/j.aiopen.2022.10.001.
[18] Rodrawangpai B, Daungjaiboon W. Improving text classification with transformers and layer normalization. Machine Learning with Applications. 022; 10:100403.
[19] Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez A N, Kaiser L, Polosukhin I. Attention Is All You Need. Nips), 2017. rXiv preprint arXiv:1706.03762, 2017; 10: p.S0140525X16001837.
[20] Yi Tay, Mostafa Dehghani, Dara Bahri, and Donald Metzler. 2022. Efficient Transformers: A Survey. CM Comput. urv. 55, 6, Article 109 (June 2023), 28 pages. ttps://doi.org/10.1145/3530811.
[21] Junzhe Li, Chenglong Wang, Xiaohan Fang, Kai Yu, Jinye Zhao, Xi Wu, Jibing Gong. Multi-label text classification via hierarchical Transformer-CNN. In Proceedings of the 2022 14th International Conference on Machine Learning and Computing (ICMLC '22); Association for Computing Machinery; New York: NY, USA, 2022 120–125. ttps://doi.org/10.1145/3529836.3529912
[22] https://www.kaggle.com/datasets/fedesoriano/stroke-prediction-dataset
[23] Ushasree D, Praveen Krishna A V, Mallikarjuna Rao C, "Enhanced stroke prediction using stacking methodology (ESPESM) in intelligent sensors for aiding preemptive clinical diagnosis of brain stroke", Measurement: Sensors. 024; 33.
[24] Bhuvanya R, Kavitha M. A real-time e-commerce accessories recommender system by coupling deep learning and histogram features. Intell. uzzy Syst. 2023; 45(1), 1179–1193. https://doi.org/10.3233/JIFS-223754.