International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9
***Article Received:** 25 July 2023 **Revised:** 12 September 2023 **Accepted:** 30 September 2023*

Design of a Novel Convolutional Deep Network Model for Car Accident Prediction

V Raviteja Kanakala[1],K.Jagan Mohan[2]
Department of IT
Annamalai University
Chidambaram, Tamilnadu, IndiaIndia
raviteja.kanakala@gmail.com,aucsejagan@gmail.com

Abstract- Real-time collision risk estimation is thought to be essential to a sophisticated traffic management system. To swiftly determine accident probability is the goal of real-time crash risk prediction. However, due to the complex traffic situation on urban arterials, urban arterials were rarely included in previous studies, which mostly focused on highways. This paper suggests using Convolutional Deep Network model (CDNM) to forecast the probability of vascular accidents in real time. This model has the benefit of being able to use both LSTM and CNN. CNN retrieves the time-invariant characteristics, while LSTM captures the data's long-term dependability. To estimate the likelihood of an accident, many sorts of data are used, including weather, traffic, and signal timing data. There are also many other data preparation methods employed. The problem of data imbalance is also addressed by normalization which oversamples the crash cases. Using a variety of measures, the CDNM is enhanced on the training data and assessed on the test data. Five more benchmark models are constructed for model comparison. K-NN, ISVM, ANN, CNN, CNN-EVT and GAN are some of the models in this group. Experimental findings show that the proposed CDNM beats the competition in terms of sensitivity, specificity, accuracy, AUC and G-mean value. The findings of this paper demonstrate that CDNM can real-time prediction of crash risk at arterials.

Keywords- accident prediction, deep learning, risk analysis, prediction, probability analysis

1. Introduction

The goal of actual crash risk prediction is to quickly calculate crash risk. a category occurrence (such as an accident or not) is the outcome of the conventional binary classification issue of predicting crash risk [1]. According to the research area, there are often two types of studies: urban arterials and highways. Most recent studies focused on freeways. The links between accident risk and several other sorts of characteristics have been the subject of much research. For instance, it was discovered that speed standard deviation had a favorable impact on collision frequency [2]. The likelihood of crashes was strongly connected with more traffic. However, there isn't much research regarding predicting accident risks in real-time at arterials. The author in [3] used real-time weather and traffic information gathered from metropolitan arterials to first assess collision likelihood and severity. He discovered that the likelihood of a collision is strongly correlated with the variance in occupancy as well as the logarithm of the flow's coefficient of variation.

The traffic data, but were averaging over a one-hour period, which could be too lengthy to adequately capture the momentary traffic condition shortly before a collision. Using four metropolitan arterials, [4] looked into the connections between real-time traffic and signal timing aspects and crash frequency. The average speed, the volume of left turns going upstream, the fraction of green lanes going downstream, and the presence of rain were all shown to have a substantial impact on collision frequency, according to the authors [5].

The rarity of accident occurrences makes it difficult to anticipate crash risk with any degree of accuracy. In actual life, non-crash instances outnumber crash cases by a wide margin, producing data that is wildly out of proportion [6]. If the original data is used without any modifications, the crash prediction model often performs very poorly. Data resampling is one typical strategy used to solve this issue. Machine learning and matched case-control approaches are the two sorts of accessible techniques. A common under-sampling technique is matched case-control, as stated by [7]. The occurrences of crashes are the ones that are chosen initially. Location, time of day, and weekday are the matching criteria for each crash. These variables are then used to choose a subset of the non-crash events for each crash. Machine learning and statistical methods are the two main forms of crash risk prediction techniques in real-time. Loglinear model, logistic regression, conditional logit model, etc. are examples of statistical approaches [8]. These models are based on reliable assumptions and frequently incorporate matched-case control data.SVM, Random Forest, and other machine-learning techniques are a few examples become popular in light of these restrictions. Recent years have seen the implementation of deep learning to address a variety of transportation issues.

A deep stack de-noise auto-encoder model was created by [9] to aggregately anticipate traffic accidents by learning from the hierarchical characteristics of human movement. The deep CNN model was used by [10] to forecast Beijing traffic speed. Images of the time and space interactions of traffic flow are created by translating spatial and temporal traffic dynamics. According to the results, CNN fared better than other algorithms like Random Forest and KNearestNeighbor. RNN has also shown to be quite successful in teaching time-series data. The recurrent neural network (RNN) adds recurrent connections, which enable information to remain, in contrast to the conventional neural network, this only changes the input vector's current state to the output vector. The RNN does not, however, capable of capturing long-term dependency. The long-short memory neural network (LSTM) was developed as a result. Memory cells and gates with long-term information storage are included LSTM enhances the performance of RNN. However, all these individual methods fail to give promising outcomes. To address the issues encountered in the existing approaches, this work proposes a novel Convolutional Deep Network Model (CDNM) for analyzing the risk factors. The model intends to analyze the feature representation and performs better prediction.

The work is drafted as: Section 2 provides a comprehensive analysis of various prevailing approaches and discusses the merits and demerits of the general methods. In section 3, the methodology is elaborated, and the numerical outcomes are discussed in section 4. The work summary is provided in section 5.

II. Related works

There are several research that have already used LSTM in the transportation industry. LSTM was used by [11] to look at Beijing's projected short-term traffic which uses a two-dimensional network to take into account geographical and temporal correlation in the transportation network. Compared to techniques like the standard RNN and the autoregressive integrated moving average model, the LSTM produces better outcomes. Similarly to this, author in [12] predicted short-term traffic flow using LSTM. Because it could retain a significant amount of past SVM and single-layer feed-forward neural networks excelled by LSTM in terms of processing data and automatically selecting the best time delays. There isn't much research out there right now that employs LSTM-based real-time crash risk predictions. Utilizing LSTM to forecast accident risk in real-time, [13] discovered that their models' sensitivity was substantially more than 60.67% of the conditional logistic model.

The advantages of several different neural networks combined in hybrid neural networks are receiving an increasing amount of attention these days. In several earlier investigations, the temporal and spatial correlations were addressed using LSTM and CNN, respectively. As an illustration, a spatiotemporal convolutional network with extended short-term memory was utilized by [14] to forecast the frequency of crashes across the city using a variety of data sources, including taxi trip data, properties of the parameters for the land use network and roads. Accordingly, the author in [15] forecast short-term traffic flow using an LSTM-CNN model.

The convolutional LSTM network was used by [16] to estimate the pace of Beijing's traffic. These researches findings demonstrated that combining LSTM and CNN produced superior outcomes versus using only LSTM and CNN alone. The use of LSTM-CNN, which lacks a spatial component, is only possible for time-series data. Time series data may be learned using the LSTM and CNN in many methods. Author in [17] assert that whereas LSTM excels at CNN and can identify by understanding one can spot patterns of regional trends and the same patterns by looking for relationships through time and sequential correlations that appear at various times. There hasn't been much research done on applying time series data using LSTM-CNN. An LSTM-CNN model was created by [18] to classify time data and evaluated the model's effectiveness using 85 datasets from UCR. The findings show how this model performs better with greater classification accuracy than other benchmark techniques.

The TreNet model, which included CNN and LSTM, was also put out by [19]. LSTM is employed to identify enduring associations, the CNN model, on the other hand, is utilized to extract pertinent characteristics from local raw data. TreNet outperformed a single CNN and an LSTM in terms of time series prediction results. For LSTM-CNN to work well on time-series data, local trends, and long-term dependence must be learned independently since they may

Article Received: *25 July 2023* ***Revised:*** *12 September 2023* ***Accepted:*** *30 September 2023*

be complementary to one another [20]. Time-series data come in numerous forms, including traffic volume, signal timing, traffic speed, etc. are used in real-time collision risk prediction. The LSTM-CNN's distinctive design makes it possible to effectively capture both the data's long- and short-term traits. Therefore, it is intriguing to look at how well LSTM-CNN predicts crash risk in real-time.

III. Methodology

This work concentrates on modelling an efficient approach for accident prediction where the inputs are taken from online resources. The proposed model performs pre-processing of samples to avoid over-fitting or under-fitting issues. Here, normalization is used for pre-processing. One of the most discriminating algorithms, Layers makes up the Convolutional Neural Network (CNN) that combines convolutional and pooling layers. One layer is layered upon another. In the composite layers, with whom the bottom layer's data rate is decreased, the convolutional layer's weight is shared significantly, and the convolutional layer's release is softened. The CNN is given certain invariance qualities by the weight sharing in the convolutional layer and the right selection of pooling algorithms.

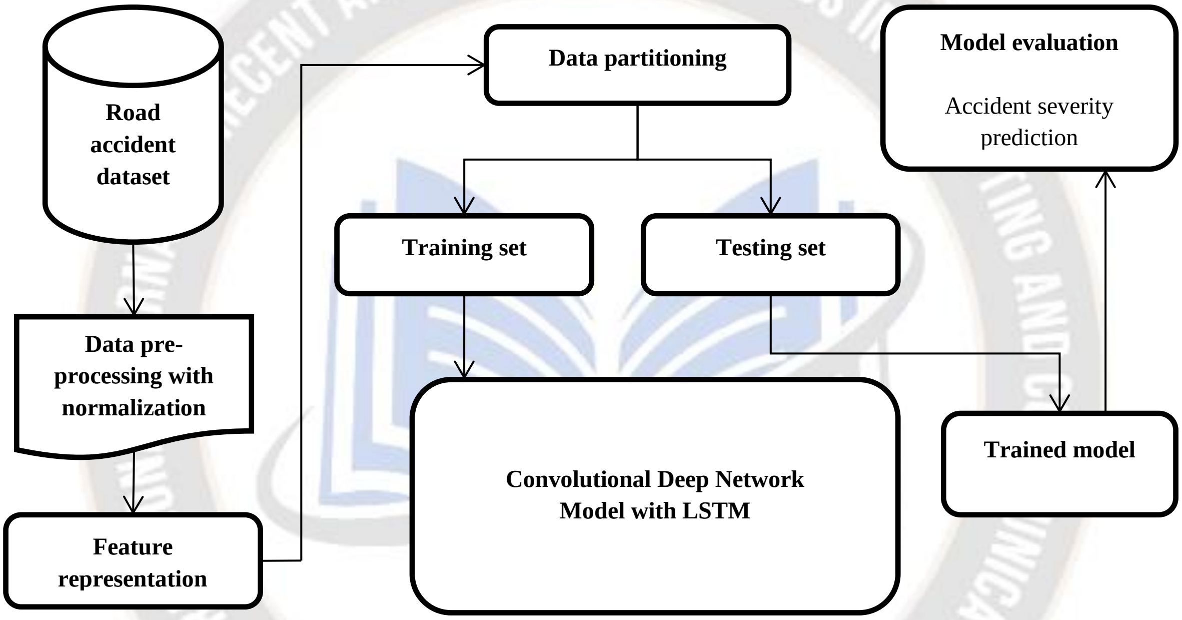

Fig 1 Block diagram of the anticipated model

In terms of computer vision and image recognition, according to studies, CNN is one of the best DL vision algorithms. The convolutional layer of a CNN produces feature maps via employing learnable kernels with sigmoid, softmax, and hyperbolic tangent as examples of non-linear or linear activation functions convolutional feature mapping from the prior layer is performed. Then, numerous feature map inputs were mixed with each output from a feature map. As shown in the subsequent Eq. (1):

$$x_j^i = f\left(\sum_{i \in M_j} x_i^{l-1} * k_{ij}^l + b_j^l\right) \quad (1)$$

Where, x_j^i is the current layer's output, x_i^{l-1} is that of the layer that came before it, k_{ij}^l is that layer's kernel and b_j^l is that layer's biases. Each output map in the input map collection M_j reflects an addictive bias B. Since there must precisely be n output maps for every n input feature map, the quantity of feature mappings used in the CNN's input and output pooling layer cannot change. But the down-sampling technique will result in a smaller size for each dimension of the final map. The following equation may be used to represent the operation:

$$x_j^i = down\,(x_j^{i-1}) \quad (2)$$

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9
***Article Received:** 25 July 2023 **Revised:** 12 September 2023 **Accepted:** 30 September 2023*

In addition to being used as its regular function, CNN always employs convolutional and pooling functions as activation functions to incorporate nonlinearity. The general design of CNN is shown in Fig 1.

A. LSTM

Long short-term memory (LSTM) and recurrent neural networks (RNNs) learning model is employed which has promise for several applications, including language modeling and prediction building. Complex long-time-lag problems that recurrent network algorithms have never been able to address are solved by LSTM. By substituting memory cells for the RNN hidden layer, Long-term associations may be learned using the LSTM learning algorithm. The cell state is the top-level idea of the LSTM, as shown by the horizontal line that goes across the top. The information in the "Get" cell state is changed by the algorithm. As indicated in Fig. 2, an input gate (i_t), forget gate (f_t), and output gate (o_t) may be described as:

$$f_t = \sigma (W_f [h_{t-1}, x_t] + b_f) \quad (3)$$

$$i_t = \sigma (W_i [h_{t-1}, x_t] + b_i) \quad (4)$$

$$C_i = \tanh(W_c [h_{c-1}, x_t] + b_c) \quad (5)$$

$$C_t = f_t * C_{t-1} + i_t * C_t \quad (6)$$

$$O_i = \sigma (W_o [h_{t-1}, x_t] + b_o) \quad (7)$$

$$h_t = O_t * \tanh(C_t) \quad (8)$$

B. Hybrid CNN with LSTM

CNN-LSTM, a potent duo that has been proposed to restrict the usage of CNNs to get priceless information and high-efficiency learning, are Convolutional Neural Networks and Long-Term Short-Term Memory. Superiority of the LSTM algorithm in detecting and modeling significantly short-term and long-term changes in time series reciprocal relationships reinforced by the order of tuples data. The suggested CNN-LSTM consists of two fundamental key parts to accomplish the aforementioned goal:

1) A pooling and convolutional layer-based one-dimensional CNN that creates features by applying a mathematical operation to the input data.

2) To utilize the created features, LSTM and thick layers are used.

CNN serves as the encoder and the decoder is provided by LSTM in the proposed CNN-LSTM algorithm. The LSTM decoder receives a feature from the encoder and processes it. The decoder then recognizes it simulates any inborn temporal correlations that may exist in the sample, both short- and long-term.

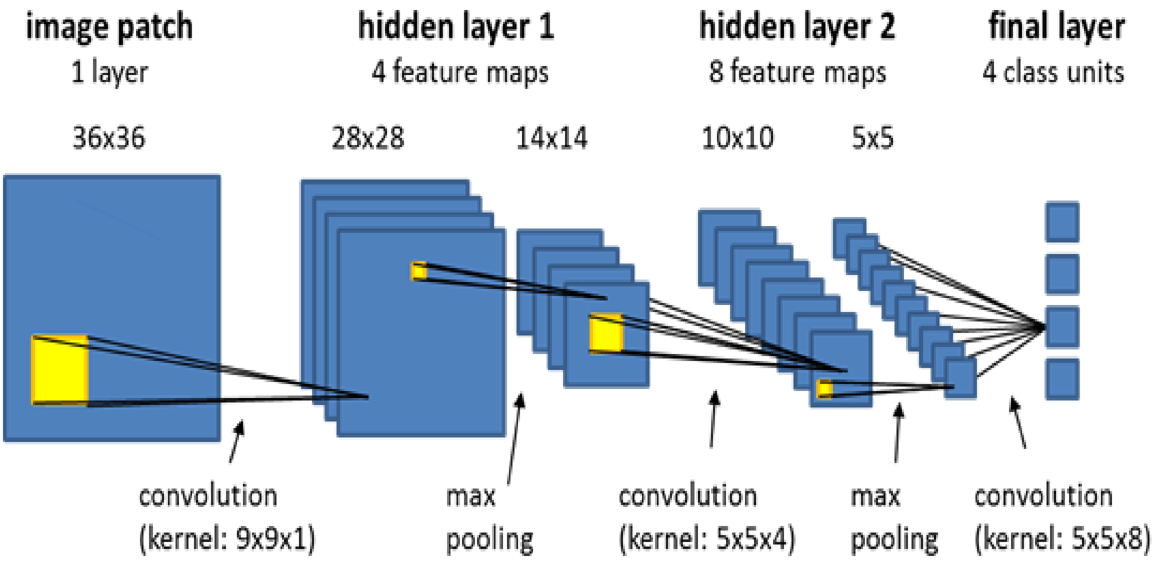

Fig 2 CNN architecture

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9
***Article Received:** 25 July 2023 **Revised:** 12 September 2023 **Accepted:** 30 September 2023*

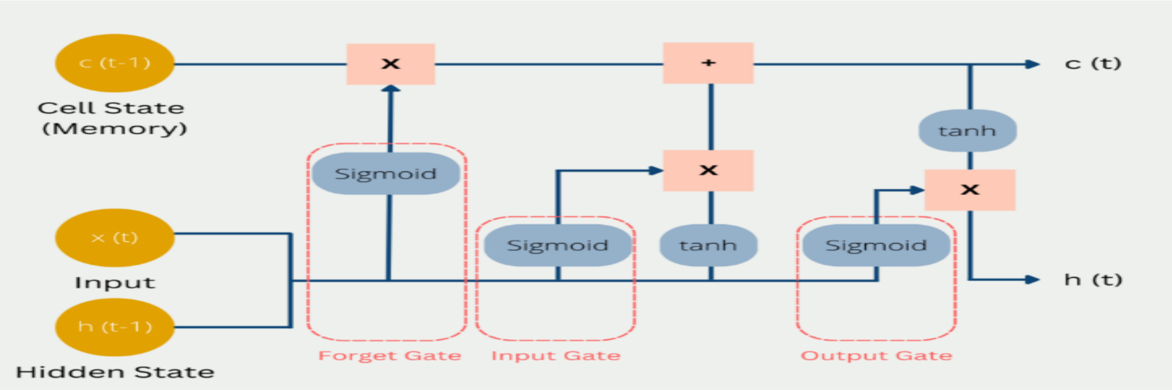

Fig 3 LSTM architecture

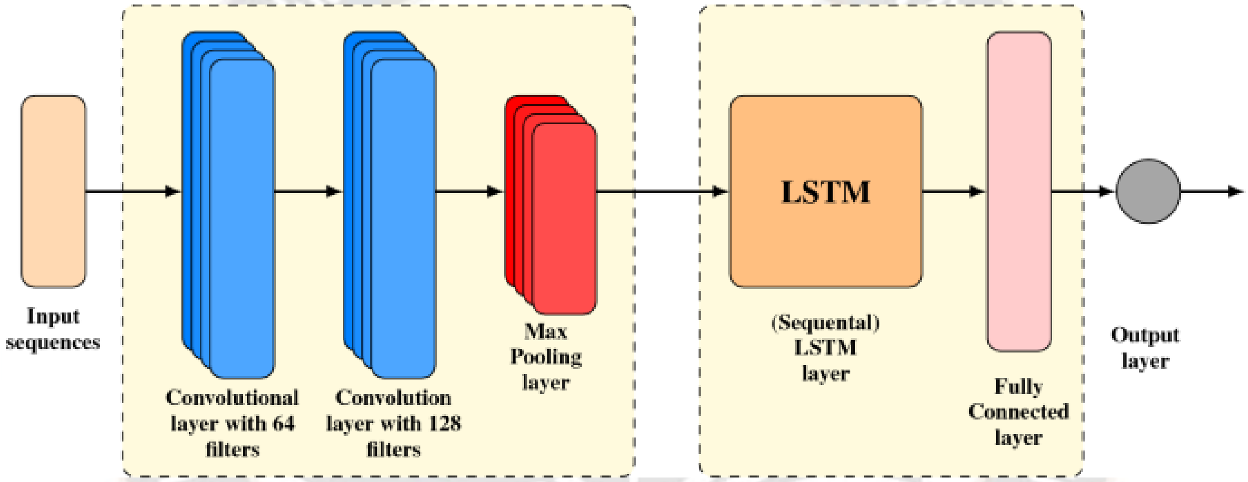

Fig 4 Convolutional Deep Network model

The following is a quick description of the order of events for each stage:

- The input layer receives data input;
- First Convolution Layers: Examines the incoming data for presenting the findings to the feature maps in (i) above;
- The second convolution layer locates the feature maps once more and employs each convolution layer has 32 feature mappings and a kernel size that is three times the input sequence to emphasize any distinguishing traits;
- Max pooling layer: eliminates specific characteristics from the feature maps above, oversimplifying them and creating matrices with short dimensions;
- Dropout layer: strengthens the neural network to prevent overfitting in the model;
- Flatten layer: Creates a single, lengthy vector from the feature maps that may be used as the input for decoding, from the feature maps that have been stripped down to their bare minimum;
- Repeat Vector layer: In the output sequence, the internal representation of the input sequence is repeated once for each time step.
- With the LSTM decoder, 100-unit hidden layer may output both the entire layout and each individual unit may be produced independently unit delivers a value once every day, which serves as the foundation for predicting what will occur in the output sequence over the following days;
- Fully connected layer: It has been demonstrated that the LSTM decoder may perform at any one

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9
***Article Received:** 25 July 2023 **Revised:** 12 September 2023 **Accepted:** 30 September 2023*

moment similarly to both output and entirely linked layers by comprehending each stage of the output sequence to construct comparable layers for predicting the output of a single sequence;

- The anticipated number of new risk and deaths during the successive days is shown in the output layer x.

IV. Numerical results and discussion

Using the Matlab code (R2021b), our research was run on a 2.50GHz Intel(R) Core(TM) i5-7300HQ processor. The four outcomes that the model would have anticipated are True positive (TP), False positive (FP), False negative (TN), False negative (FN), and True positive (TP). When FN results are positive, they are wrongly labeled as negative, while when FP results are negative, they are incorrectly classified as positive. TN and TP results are accurately categorized. Diabetes, hypertension, and prehypertension are the outcomes of datasets respectively as are diabetes from datasets. Using a 10-fold CV, we created the models for every classification model with an average of all the models serving as performance metrics. One of the performance metrics used in this study, precision (p) has been defined as:

Algorithm	Sensitivity (%)	Specificity (%)	Accuracy (%)	AUC	G-mean
k-NN	64	68	65	0.62	0.3588
ISVM	67	68	67	0.63	0.6415
ANN	69	78	73	0.65	0.8685
CNN	83	86	84	0.78	0.4165
CNN-EVT	92	94	95.6	0.89	0.9349
GAN	95	96	98	0.92	0.9564
	95.1	**96.2**	**98.1**	**0.93**	**0.957**

Table 1 Comparative analysis of proposed vs. existing

The definition of recall, denoted by the letter (r) is:

$$r = \frac{TP}{TP + FN} \tag{10}$$

The word "F-measure" is denoted by the symbol (f) which has the following definition:

$$f = \frac{2 * pr}{p + r} \tag{11}$$

The meaning of the symbol (acc), which stands for "accuracy", is:

$$Acc = \frac{TP + FN}{TP + FN + FP + TN} \tag{12}$$

We also contrasted the suggested model performance with existing models while taking into account the area under the curve (AUC). AUC of the given k training data is described as:

$$AUC\ (x^+, x^-) = \frac{1}{k^+ k^-} \sum_{i=1}^{k^+} \sum_{j=1}^{k^-} 1_{h(x_i^+) > h(x_j^-)} \tag{13}$$

Where the answer to the phrase $1_{h(x_i^+) > h(x_j^-)}$ is a $'1'$ when the elements are $h(x_i^+) > h(x_j^-)$, $for\ all\ i = 1,2, \dots, k^+, for\ all\ j = 1,2, \dots, k^-$ and '0' when they are not. When the model achieves the highest level of accuracy, the AUC value ought to be seen to be near or equal to t.

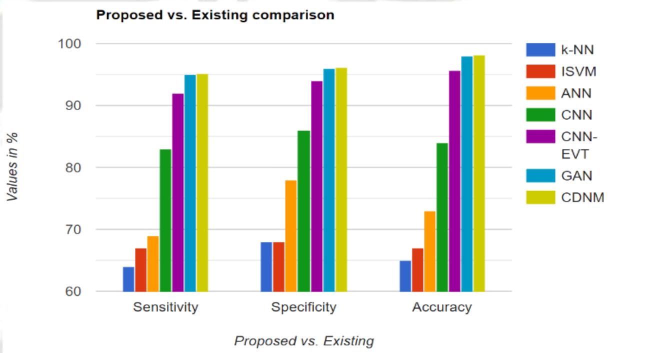

Fig 5 Comparison of proposed vs. existing

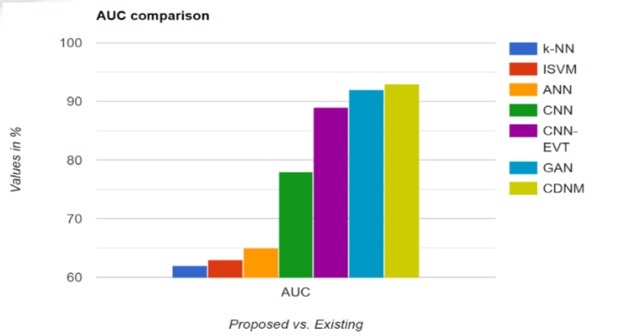

Fig 6 Comparison of various approaches

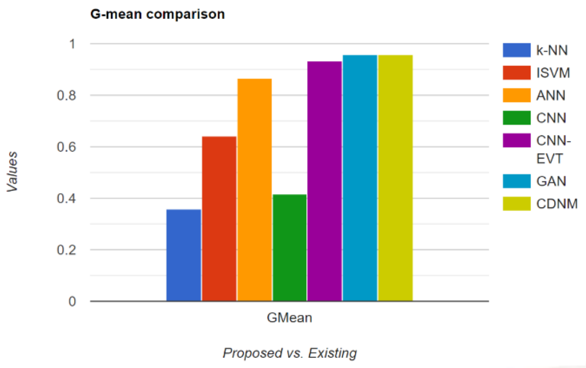

Fig 7 G-mean comparison

Table 1 displays the evaluation of several metrics, including AUC, G-mean, accuracy, sensitivity, specificity, and precision. Comparisons are made between the proposed CDNM and several metrics, including k-NN, ISVM, ANN, CNN, and CNN-EVT. The suggested CDNMSen G-mean is 95.64%, which is greater than other techniques by 59.76%, 31.49%, 8.79%, 53.99%, and 86.15%. The projected CDNM sensitivity is 95%, which is greater than other techniques by 31%, 28%, 26%, 12%, and 3%. The expected ts-LA has a 96% specificity, which is 28%, 28%, 18%, 10%, and 2% greater than that of other techniques. The expected model's accuracy is 98%, which is greater than previous techniques by 33%, 31%, 25%, 14%, and 2.4%. The AUC of the expected model is 92%, which is greater than other techniques by 30%, 29%, 27%, 6%, and 3% (see Fig 5, Fig 7 and Fig 8).

V. Conclusion

Convolutional Deep Network model (CDNM) was used in this work to forecast car crash risk in real-time. First, the weather, traffic, and signal timing yearly averages were acquired. The second step involved pre-processing raw data and cleaning it up based on correlation and importance. Third, to solve the problem of imbalanced data, normalization was employed. Finally, CDNM for real-time crash risk predictions was created. For comparisons with various assessment measures, several benchmark models are implemented. The findings show that the suggested CDNM performs better than the competition in several areas. It initially achieves an 88% sensitivity level and a 12% false alarm rate. Second, as opposed to techniques such as LSTM, CNN, XGBoost, Bayesian Logistics Regression, etc. it has the greatest AUC value, 0.93. The prediction accuracy highlights the baseline for the CDNM effective performance which demonstrates that the features retrieved by the network appropriately differentiate crash and non-crash occurrences.

Overall, this paper is successful in demonstrating the viability of real-time deep neural network crash risk prediction. In addition, CDNM performance may be improved by using CNN to extract features in a new way. The findings of this research might be applied to the creation of a modern traffic management system with the potential to lower accidents. The most recent research still has a lot of problems, though. First, adding additional layers or experimenting with more combinations of various hyperparameters may enhance the performance of CDNM. Additionally, geographic characteristics could be taken into account as potential CNN inputs, which might enhance the model's performance. Different time-slice impacts could be assessed second since the model's performance may be affected by how the data is handled.

REFERENCES

[1] Jiang, Y.; Chen, L.; Zhang, H.; Xiao, X. Breast cancer histopathological image classification using convolutional neural networks with small SE-ResNet module. PLoS ONE 2019, 20, e0214587.

[2] Zhang, J.; Wu, Q.; Shen, C.; Zhang, J.; Lu, J. Multilabel image classification with regional latent semantic dependencies. IEEE Trans. Multimed. 2018, 20, 2801–2813.

[3] Zhou, F.; Hang, R.; Liu, Q.; Yuan, X. Integrating convolutional neural network and gated recurrent unit for hyperspectral image spectral-spatial classification. In Chinese Conference on Pattern Recognition and Computer Vision (PRCV); Springer: Cham, Switzerland, 2018; pp. 409–420

[4] Casado-Sanz N, Guirao B, Attard M (2020) Analysis of the risk factors affecting the severity of traffic accidents on Spanish crosstown roads: the drivers perspective. Sustainability 12(6):2237

[5] Gu X, Li T, Wang Y, Zhang L, Wang Y, Yao J (2018) Traffic fatalities prediction using support vector machine with hybrid particle swarm optimization. J Algorithms Comput Technol 12(1):20–29

[6] Habib MT, Majumder A, Jakaria A, Akter M, Uddin MS, Ahmed F (2018) Machine vision-based papaya disease recognition. J King Saud Univ Comput Inf Sci 32(3):300–309

[7] Laytin AD, Seyoum N, Kassa S, Juillard CJ, Dicker RA (2020) Patterns of injury at an Ethiopian referral

hospital: using an institutional trauma registry to inform injury prevention and systems strengthening. Afr J Emerg Med 10(2):58–63

[8] Li L, Shrestha S, Hu G (2017) Analysis of road traffic fatal accidents using data mining techniques. In: 2017 IEEE 15th international conference on software engineering research, management and applications (SERA). IEEE, pp 363–370

[9] Sarkar S, Vinay S, Raj R, Maiti J, Mitra P (2019) Application of optimized machine learning techniques for predicting occupational accidents. Comput Oper Res 106:210–224

[10] Tiwari P, Kumar S, Kalitin D (2017) Road-user specific analysis of traffic accident using data mining techniques. In: International conference on computational intelligence, communications, and business analytics. Springer, Berlin, pp 398–410

[11] Hossain M, Muromachi Y (2012) A Bayesian network-based framework for real-time crash prediction on the basic freeway segments of urban expressways. Accid Anal Prev 45(8):373–381

[12] Shew C, Pande A, Nuworsoo C (2013) Transferability and robustness of real-time freeway crash risk assessment. J Saf Res 46(9):83–90

[13] Xu C, Wang W, Liu P, Li Z (2015) Calibration of crash risk models on freeways with limited real-time traffic data using Bayesian meta-analysis and Bayesian inference approach. Accid Anal Prev 85:207–218

[14] Shi Q, Abdel-Aty M, Yu R (2016) Multi-level Bayesian safety analysis with unprocessed Automatic Vehicle Identification data for an urban expressway. Accid Anal Preven 88(1):68–76

[15] Kwak H, Kho S (2016) Predicting crash risk and identifying crash precursors on Korean expressways using loop detector data. Accid Anal Prev 88:9–19

[16] Bejnordi, B.E.; Litjens, G.; Timofeeva, N.; Otte-Höller, I.; Homeyer, A.; Karssemeijer, N.; van der Laak, J.A. Stain specific standardization of whole-slide histopathological images. IEEE Trans. Med. Imaging 2015, 35, 404–415.

[17] Wang, J.; Yu, L.; Lai, K; Zhang, X. Community-based Weighted Graph Model for Valence-Arousal Prediction of Affective Words. IEEE-ACM Trans. Audio Speech Lang. Process. 2016, 24, 1957–1968.

[18] Y. Feng, B. Hu, H. Hao, Y. Gao, Z. Li, and J. Tan, “Design of distributed cyber–physical systems for connected and automated vehicles with implementing methodologies,” IEEE Trans. Ind. Informat., vol. 14, no. 9, pp. 4200–4211, Sep. 2018.

[19] Chen and Y.-F. Ho, “Centimeter-grade metropolitan positioning for lane-level intelligent transportation systems based on the Internet of vehicles,” IEEE Trans. Ind. Informat., vol. 15, no. 3, pp. 1474–1485, Mar. 2019.

[20] Sherubha, “Graph Based Event Measurement for Analyzing Distributed Anomalies in Sensor Networks”, Sådhanå(Springer), 45:212, https://doi.org/10.1007/s12046-020-01451-w