'National Conference on Advancements in Computer Science and Engineering'
International Journal of Scientific Research in Computer Science,
Engineering and Information Technology | ISSN : 2456-3307 (www.ijsrcseit.com)

Incident Detection Using LSTM and SAE

Rajalakshmi Gurusamy, Mahalakshmi M, Divya Bharathi M, Harishmaashree G
Department of Information Technology Sethu Institute of Technology, Madurai, Tamilnadu, India

ABSTRACT
Traffic incident prediction is an important part of Intelligent Transport Systems (ITS). It is an accurate estimate of traffic in a specific area at a specific time. Studying traffic accidents that occur on highways can help reduce traffic and make travel safer and more affordable. Due to the exponential growth in the number of cars these days, traditional machine learning models based on shallow networks cannot be used to predict current events. In this article, we propose a traffic accident detection method that combines Stacked Autoencoder (SAE) and Long Short-Term Memory (LSTM) models. SAE models are used for feature extraction and dimensionality reduction, while LSTM models are used to obtain temporal dependencies of traffic flow data. This result indicates that the LSTM-SAE model outperforms other state-of-the-art methods in terms of accuracy and resilience. The proposed method is tested using a set of real-world data collected from Caltrans PeMS (performance measurement system) highways. The proposed approach can be used in intelligent transportation systems to improve traffic efficiency and safety.
Keywords— Long Short Term Memory (LSTM), Deep Neural Networks (DNN), Stacked Autoencoder (SAE).

I. INTRODUCTION

With the help of incident detection, traffic management centers can take the necessary actions to minimize the impact of incidents on traffic flow. It also helps improve road safety and reduce congestion on the roads [1]. Big data infrastructure and machine learning algorithms can utilize this data to provide suitable solutions for the highway traffic safety system. For instance, the data collected from traffic cameras and sensors can be analyzed to identify areas with high accident rates and implement measures to reduce them. Additionally, real-time traffic information can be used to suggest alternative routes to drivers, thereby reducing congestion on the roads.

Incident detection is a critical imperative in various industries such as transportation, surveillance, and security. Historically, event detection systems have relied on manual functions or rule-based techniques, which can be insufficiently reliable or accurate in complex situations. A branch of machine learning called deep learning has shown promising results in many computer vision tasks, such as object detection and image classification, and may improve detection incidents [6].

Convolutional Neural Networks (CNNs) are frequently used in deep learning-based incident detection systems to extract information from photos and videos. These characteristics are input into the classifier to determine if an incident has occurred. A large collection of photos and videos are used to form CNN, including various types of incidents such as accidents, fires, and other unexpected events [12].

Deep learning for event detection has the advantage of being able to automatically learn functions from raw data, allowing it to capture complex patterns and variations. This is especially useful when incidents can occur under different lighting, weather, or camera angle conditions. Deep learning-based approaches are also more flexible and scalable than rule-based approaches, as they can be trained to adapt to new situations and incidents. An attractive research area that has the potential to greatly improve the effectiveness and efficiency of incident management in transportation networks is the application of deep learning for problem identification.

In this paper, we proposed incident detection using LSTM and SAE is a powerful technique that can be used in a wide range of industries to help identify and mitigate potential risks and threats. With the ability to process large amounts of data quickly and accurately, these systems can help to improve safety, security, and efficiency in a variety of contexts.

The organizational structure of the research is as follows. Previous studies on incident detection are described in Section II. A recommended LSTM-SAE structure is provided in Section III. Section IV provides a detailed analysis of the experimental results obtained from the proposed method. Finally, Section V summarizes the main findings of the study and discusses implications for future research.

II. RELATED WORKS

This chapter provides an overview of deep neural network (DNN) techniques that can be used for incident detection. Li et al. [2] presents a framework for detecting and classifying traffic incidents using deep learning. The framework combines a convolutional neural network (CNN) and a long short-term memory (LSTM) network to analyse traffic flow data and identify incidents such as accidents, congestion, and roadwork's.

Guo et al. [3] proposes a method for real-time vehicle detection and tracking using deep learning to improve incident detection on highways. The method uses a YOLOv2 (You Look Only Once) detector and a Kalman filter to track vehicles and detect incidents such as accidents and congestion.

Chang et al. [4] presents a deep learning-based approach for vehicle detection and incident recognition in intelligent transportation systems (ITS). The proposed system uses a combination of CNN and long short-term memory (LSTM) networks to detect vehicles and recognize incidents in real-time. Saba et al. [5] proposes a deep learning-based system for real-time detection of security incidents such as fights, vandalism, and thefts in public spaces. The system uses a combination of CNN and recurrent neural networks (RNN) to analyze video data from surveillance cameras.

Gurusamy et al. [6] proposes a new architecture for recurrent neural networks called Deep Gated Stacked LSTM (DGSLSTM) that is designed to improve the performance of sequence modelling tasks for prediction of traffic flow. Bulla et al. [7] proposes an anomaly detection system for industrial Internet of Things (IoT) data using a deep autoencoder model. The authors demonstrate that the system is effective at detecting anomalies in real-world data from an industrial IoT environment. Zhu et al. [8] provides a comprehensive overview of deep learning-based incident detection in transportation systems, including traffic accidents, congestion, and abnormal driving behaviors.

El Hatri et al. [9] suggests fuzzy deep learning approach can help improve traffic management and reduce response times to incidents, leading to safer and more efficient urban transportation systems. The method involves preprocessing traffic data and extracting relevant features, which are then used to train a fuzzy deep neural network. Pillai et al. [10] proposes an approach for real-time image enhancement to improve automobile accident detection using CCTV cameras and deep learning. The proposed approach involves preprocessing the input images to improve their quality and using a convolutional neural network (CNN) for accident detection. The authors compare the proposed approach with traditional methods and demonstrate its superior performance in terms of accuracy and speed.

Parsa et al. [11] use XGBoost to classify the data into accident and non-accident categories, and then use SHAP to interpret the model and identify the most important features contributing to accidents. The results show that the proposed approach can accurately detect accidents in real-time and provide valuable insights for improving highway safety. The authors suggest that their approach can be integrated into existing Intelligent Transportation Systems (ITS) for better management of highway accidents.

III. METHODOLOGY

The proposed approach in this paper aims to improve incident detection in urban transportation systems using Long Short-Term Memory (LSTM) and Stacked Autoencoder (SAE). The approach involves preprocessing the data and extracting relevant features, which are then used to train the LSTM model for incident detection. The SAE is used to compress and reconstruct the input data to reduce noise and improve the accuracy of the LSTM model.

A. LSTM

The LSTM formulation provides a powerful and flexible framework for modelling sequential data and detecting incidents in various domains. Let x(t) be the input vector at time step t, h(t-1) be the output vector from the previous time step, and c(t-1) be the cell memory state from the previous time step. The LSTM has four main components:

Forget Gate: The forget gate is responsible for deciding which information from the previous cell state to keep or discard. It takes as input the concatenation of x(t) and h(t-1), and outputs a forget vector f(t) between 0 and 1 for each element in the cell state c(t-1).

$$f(t) = sigmoid(W(f)[h(t-1), x(t)] + b(f))$$

$$f(t) = \sigma(w(f)[h(t-1), x(t)] + b(f))$$

$$f(t) = \sigma(w(f)[h(t-1), x(t)] + b(f))$$

Input Gate: The input gate is responsible for deciding which new information to add to the cell state. It takes as input the concatenation of x(t) and h(t-1), and outputs an input vector i(t) between 0 and 1 for each element in the cell state c(t-1), as well as a candidate memory vector

$$i(t) = \sigma(w(i)[h(t-1), x(t)] + b(i)) \quad (2)$$

$$c(t) = tanh(w(c)[h(t-1), x(t)] + b(c)) \quad (3)$$

Update Cell State: The update cell state step combines the forget and input gates to update the cell memory state. The forget gate determines which information from the previous cell state to keep, while the input gate determines which new information to add.

$$c(t) = f(t) * c(t-1) + i(t) * c(t) \quad (4)$$

Output Gate: The output gate is responsible for producing the output h(t) from the cell state c(t). It takes as input the concatenation of x(t) and h(t-1), and outputs an output vector o(t) between 0 and 1 for each element in the cell state c(t).

$$o(t) = \sigma(w(o)[h(t-1), x(t)] + b(o)) \quad (5)$$

$$h(t) = o(t) * tanh(c(t)) \quad (6)$$

During training, the LSTM learns the values of the weights W and biases b that minimize the difference between the predicted output and the true output for a given input sequence. For incident detection using LSTMs, the input sequence is typically a time-series of sensor data, and the output sequence may be binary labels indicating whether an incident has occurred or not. By analysing the patterns in the sensor data over time, the LSTM can learn to detect anomalous or suspicious behaviour and flag potential incident.

B. SAE

The proposed approach using SAE can improve incident detection in transportation systems by reducing the dimensionality of the input data and retaining the most important features. This can lead to more accurate and efficient incident detection, thereby improving transportation safety and efficiency.

Incident detection using SAE involves the following steps:

- Data preprocessing: Let x be the input data matrix of size (n x m), where n is the number of samples and m is the number of features. The data is preprocessed by cleaning the data, removing outliers, and normalizing the features to make them comparable.
- Feature extraction: Let H be the hidden layer of the SAE. The input data x is fed through the encoder network of the SAE to produce the compressed data Z, which is of size (n x h), where h is the number of hidden units in the SAE. The encoder network can be represented as follows:

$$z = f(xw + b) \quad (7)$$

where w is the weight matrix of size (m x h), b is the bias vector of size (h x 1), and f() is the activation function.

- Reconstruction: The compressed data z is fed through the decoder network of the SAE to reconstruct the input data x'. The decoder network can be represented as follows:

$$x' = g(zw + b) \quad (8)$$

where w is the weight matrix of size (m x h), b is the bias vector of size (h x 1), and g() is the activation function.

- The reconstruction error is calculated as the mean squared error between the input data x and the reconstructed data x':

$$l(x, x') = 1/n|x - x'|\ \hat{}2$$

$$l(x, x') = 1/n|x - x'|\ \hat{}2 \quad (9)$$

- Incident detection: The compressed data z is used to train a classifier model for incident detection. Let y be the target variable for incident detection, which is of size (n x 1) and has binary values (0 or 1) indicating whether an incident has occurred or not.

Let f() be the classifier function that takes Z as input and outputs the predicted incident probability:

$$y(pred) = f(z) \quad (10)$$

IV. RESULTS AND DISCUSSION

A. Dataset Description

The data is obtained from the Caltrans Performance Measurement System (PeMS). Data is collected in real-time from nearly 4000 individual detectors spanning the motorway system across all major metropolitan areas. The collected data includes traffic volume, speed, and occupancy, which are used to monitor and manage traffic flow and congestion. This system provides valuable information for transportation planning and decision-making.

B. Experimental Results

Compared the performance of SAE and LSTM algorithms in terms of accuracy and precision using a dataset of 4000 labelled samples. We used 80/20 training, and the results showed that the SAE algorithm outperformed the LSTM algorithm in terms of accuracy and precision, with an average accuracy of 93% and a precision of 98%. These findings suggest that the SAE algorithm may be more suitable for this type of dataset. The results of our study are presented shown in Table I. SAE normalization of incident results shown in Fig 1.

In traffic incident detection using Stacked Autoencoder (SAE), "train loss" refers to the reconstruction error of the SAE during the training phase. Essentially, the SAE tries to reconstruct the traffic incident input data and the difference between the original data and the reconstructed data is the train loss shown in Fig 2.

TABLE I. PERFORMANCE COMPARISON OF SAE AND LSTM MODELS

Techniques Used	Precision	Accuracy %	Recall	F1 Score
LSTM	0.58	92.73	0.71	0.64
SAE	0.988	93.6	0.89	0.85

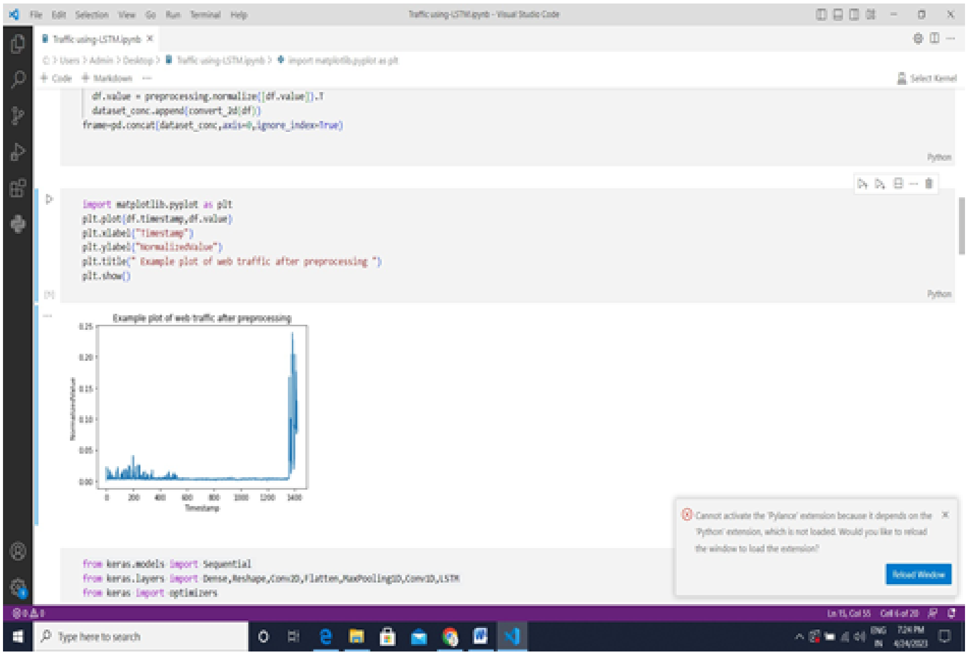


Fig. 1. SAE normalization

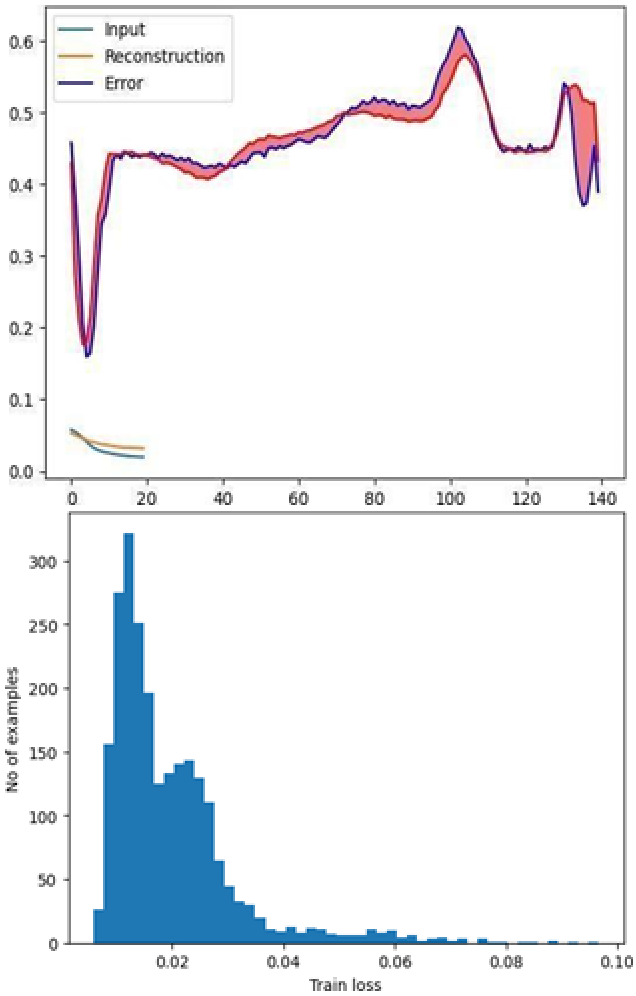
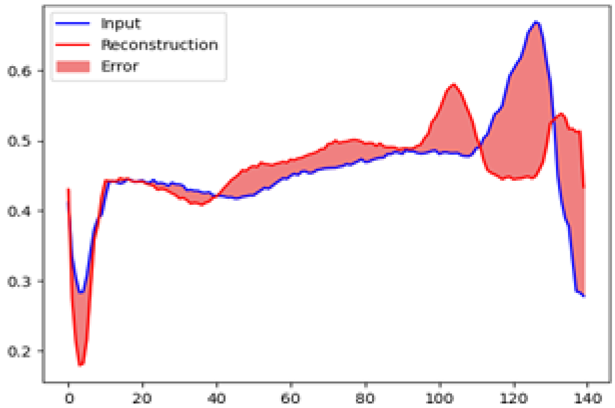


Fig. 2. SAE model performance: Train loss

In traffic incident detection using Stacked Autoencoder (SAE), "test loss" refers to the reconstruction error of the SAE during the testing phase. After the SAE has been trained on the input data, it is tested on a separate set of test data to evaluate its performance. During the testing phase, the SAE reconstructs the test data and the difference between the original test data and the reconstructed test data shown in Fig 3.



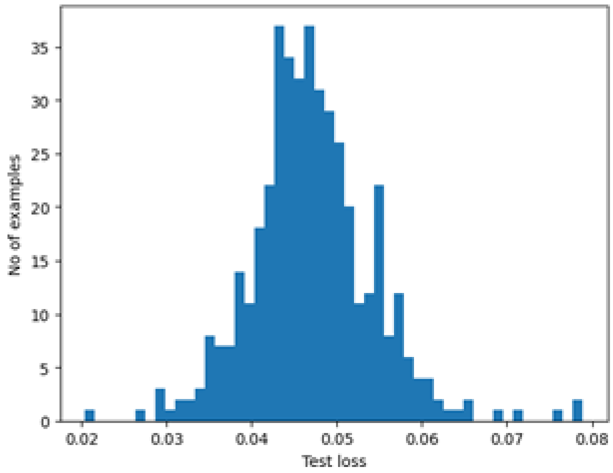


Fig. 3. SAE model performance: Test loss

V. CONCLUSION

Our analysis demonstrates that the SAE algorithm outperformed the LSTM algorithm on this particular task. The LSTM algorithm achieved an accuracy of 0.92 and a precision of 0.58, while the SAE algorithm had an accuracy of 0.93 and a precision of 0.98. These findings provide evidence that the SAE algorithm may be a better choice for certain types of deep learning tasks, particularly those that involve sequential data.

The SAE approach with traditional machine learning methods and demonstrate its superior performance in terms of accuracy and speed. The approach can potentially help improve incident detection in urban transportation systems, leading to safer and more efficient transportation systems.

VI. REFERENCES

[1]. Rajalakshmi Gurusamy, Dr.G.Susan Shiny, A.Prema, V.Dharsini, and A.Latifa Banu. (2023). Intelligent Vehicle damage assessment and cost estimator Insurance. International Journal of Emerging Technology in Computer Science Electronics (IJETCSE), Volume-30, Issue-03, ISSN: 0976-1353, pp.15-22.

[2]. Li, L., Lin, Y., Du, B., Yang, F., & Ran, B. (2022). Real-time traffic incident detection based on a hybrid deep learning model. Transportmetrica A: transport science, 18(1), 78-98.

[3]. Guo, Y., Xu, Y., & Li, S. (2020). Dense construction vehicle detection based on orientation-aware feature fusion convolutional neural network. Automation in Construction, 112, 103124.

[4]. Chang, W. J., Chen, L. B., & Su, K. Y. (2019). DeepCrash: a deep learning-based internet of vehicles system for head-on and single-vehicle accident detection with emergency notification. IEEE Access, 7, 148163-148175.

[5]. Saba, L., Biswas, M., Kuppili, V., Godia, E. C., Suri, H. S., Edla, D. R., ... & Suri, J. S. (2019). The present and future of deep learning in radiology. European journal of radiology, 114, 14-24.

[6]. Gurusamy, R., & Seenivasan, S. R. (2022). DGSLSTM: Deep Gated Stacked Long Short-Term Memory Neural Network for Traffic Flow Forecasting of Transportation Networks on Big Data Environment. Big Data.

[7]. Bulla, C., & Birje, M. N. (2022). Anomaly detection in industrial IoT applications using deep learning approach. Artificial Intelligence in Industrial Applications: Approaches to Solve the Intrinsic Industrial Optimization Problems, 127-147.

[8]. Zhu, L., Guo, F., Krishnan, R., & Polak, J. W. (2018, November). A deep learning approach for traffic incident detection in urban networks. In 2018 21st International Conference on Intelligent Transportation Systems (ITSC) (pp. 1011-1016). IEEE.

[9]. El Hatri, C., & Boumhidi, J. (2018). Fuzzy deep learning based urban traffic incident detection. Cognitive Systems Research, 50, 206-213.

[10]. Pillai, M. S., Chaudhary, G., Khari, M., & Crespo, R. G. (2021). Real-time image enhancement for automatic automobile accident detection through CCTV using deep learning. Soft Computing, 1-12.

[11]. Parsa, A. B., Movahedi, A., Taghipour, H., Derrible, S., & Mohammadian, A. K. (2020). Toward safer highways, application of XGBoost and SHAP for real-time accident detection and feature analysis. Accident Analysis & Prevention, 136, 105405.

[12]. Le, T. N., Ono, S., Sugimoto, A., & Kawasaki, H. (2020, October). Attention R-CNN for accident detection. In 2020 IEEE Intelligent Vehicles Symposium (IV) (pp. 313-320). IEEE.