Layout: T3_Grey Book ID: 454584_1_En Book ISBN: 978-3-030-10373-6
Chapter No.: 11 Date: 4-3-2019 Time: 2:32 pm Page: 127/137

11 Applications of Deep Learning in Severity Prediction of Traffic Accidents

11.1 Introduction

Future prediction is a fascinating topic for human endeavor and is identified as a critical tool in transportation management. Understanding an entire transportation network is more difficult than transportation on a single road. The main purpose of this effort is to provide a superior route with high safety level and support traffic managers in efficiently managing road network.

Recent studies have predicted that in 2030, traffic accidents will be the fifth leading cause of death worldwide (Sameen and Pradhan 2017a, b, c; Sameen et al. 2016). The costs of fatalities and driver injuries due to traffic accidents also greatly affect the society. These insights call for investigating various aspects of traffic accident data analysis and modeling in numerous geographic regions. In general, two research methods, namely, statistical methods and neural networks (NNs) dominate the studies on traffic crash forecasting. Statistical techniques, such as *k*-nearest neighbors (Lv et al. 2009), support vector machine (Li et al. 2008, 2012), and logistic regression (Al-Ghamdi 2002), are applied to predict the frequency and injury severity of traffic accidents. Artificial NN (ANN) is used to implement flexibility, generalizability, and strong forecasting power (Delen et al. 2006; Moghaddam et al. 2011). A large amount of high-resolution data are generated on freeway networks using traffic sensors to predict traffic crashes. However, deep learning theory gradually begins to exhibit superiority over other techniques.

The advent of artificial intelligence has enabled deep learning to experience much patronage, especially with the aid of high-speed computing machines. The use of computational intelligence methods has encouraged paradigm shift from conventional traffic forecasting to short-term traffic forecast based on deep learning approaches. Employing deep learning principle can resolve many issues related to dimensionality with the aid of distributed calculation (Ma et al. 2015). Success on applying deep learning has been recorded, especially in computer vision, speech recognition, and natural language processing (Krizhevsky et al. 2012; Graves et al. 2013; Sarikaya et al. 2014). With deep learning theory, several alternatives to NN, such as feedforward NN, recurrent NN (RNN), and convolutional NN (CNN), have been developed to ease traffic prediction.

Although deep learning models play a pivotal role in traffic accident forecast, they have drawbacks. The primary drawback of deep learning models is their low explanatory power and the large datasets required for training. The basic objective of deep learning is to develop the representation of the actual predictor vector so that transformed data can be used for classification or linear regression. Thus, this study aims to analyze and discuss the difference among three basic network architectures, namely feedforward NN, CNN, and RNN, in predicting the injury severity of traffic accidents on high-speed highways.

11.2 Deep Learning Models

11.2.1 Feedforward NNs

In machine learning, NNs are a family of biological learning models. A simple NN model is an interconnection of neurons or nodes, which comprise three layers, namely input, hidden, and output layers. In this study, the model represents a nonlinear mapping between the input values (accident predictors) and output parameters (injury severity levels). Neurons are a systematic connection of weight vectors which are usually structured in layers with full connections between successive layers. The output signal is a function of the inputs to the node, which is modified by a simple activation function (Gardner and Dorling 1998; Mokhtarzade and Zoej 2007).

The possibility of learning has attracted the research fraternity in NNs. The most common learning algorithm for NNs is the backpropagation developed by Paul Werbos in 1974 which was later rediscovered by Rumelhart and Parker. The algorithm was designed to reduce the error function by

B. Pradhan and M. Ibrahim Sameen, *Laser Scanning Systems in Highway and Safety Assessment*, Advances in Science, Technology & Innovation,
https://doi.org/10.1007/978-3-030-10374-3_11

using an iterative approach, as shown in Eq. (11.1). Despite the success of such NNs in remote sensing applications, their computational complexity is relatively high and has a drawback of overlearning (Baczyński and Parol 2004; Mia et al. 2015).

$$E = \frac{1}{2}\sum_{i=1}^{L}\left(d_j - o_j^M\right)^2, \tag{11.1}$$

where d_j and o_j^M denote the output and the current response of the node 'j' in the output layer, respectively, and 'L' represents the number of nodes in the output layer. In this approach, weight parameters are corrected and added to the preceding values, as shown in Eq. (11.2).

$$\begin{cases} \Delta w_{i,j} = -\mu \frac{\partial E}{\partial w_{i,j}} \\ \Delta w_{i,j}(t+1) = \Delta w_{i,j} + \alpha \Delta w_{i,j}(t) \end{cases}, \tag{11.2}$$

$w_{i,j}$ denotes the parameter between nodes i and j, Δ is the learning rate that controls the amount of adjustment, α is a momentum factor between 0 and 1 and 't' represents the number of iterations. The parameter α refers to the smoothing factor resulting from its ability to take care of the rapid changes between weights (Yang 1995).

11.2.2 CNNs

Another effective NN in areas of computer vision and classification is CNN (Krizhevsky et al. 2012). The first CNN was developed by Yann LeCun (Fig. 11.1), and it has contributed immensely to many works in computer vision and deep learning environment. This method is also referred to as LeNet and was used in the early years mainly for character recognition works, such as zip digits, codes, and handwriting. CNN can also process data in multiple array formats (color images, signals, sequences, audio, and video) depending on the dimensions of the convolution operations (1D, 2D or 3D) (LeCun et al. 2015).

Four processes, namely convolution, pooling or subsampling, nonlinearity (rectified linear unit or ReLU), and classification (fully connected layer), are typically involved in CNN operations (LeCun et al. 2015). These operations comprise the building blocks of CNN. Features, such as image and time series, can be extracted from input data using convolution operation. This process preserves the spatial interactions between the samples of entry data by learning small subsets of the input files. ReLU is an additional nonlinear operation used after every convolution operation in networks. ReLU yields output in an element-wise manner, which substitutes negative values in the feature map with zero and resolves real-world problems by utilizing nonlinearity in the network. Subsampling or downsampling is a spatial pooling used to reduce the dimensionality of each feature map but retains the most crucial information. Max, average, or sum is among the different types of spatial pooling available. Max pooling uses the biggest element within the rectified feature map in any defined spatial neighborhood. Average pooling can be considered instead of the greatest element or the sum of all elements in that window. However, according to Schindler and Van Gool (2008), max pooling is proven to be better than average pooling. AQ1

In a traditional multilayer perceptron, connected layer uses a softmax activation function in the output layer. The output from the convolutional and pooling layers represents the high-level features of the input data. The main reason behind implementing fully connected layer is its capability to classify features in the input data into different classes based on the training dataset.

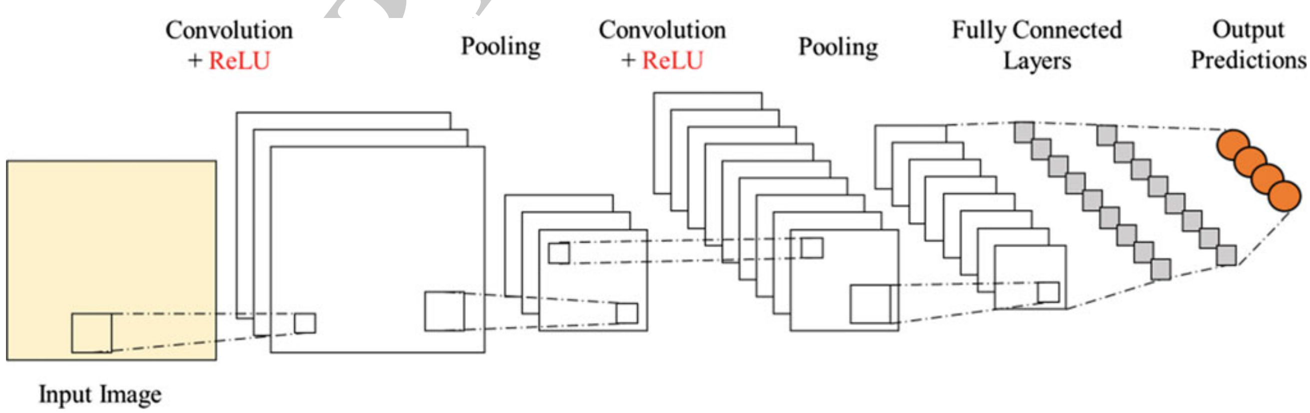

Fig. 11.1 Typical CNN model architecture

Layout: T3_Grey Book ID: 454584_1_En Book ISBN: 978-3-030-10373-6
Chapter No.: 11 Date: 4-3-2019 Time: 2:32 pm Page: 129/137

11.2.3 RNNs

RNNs are NNs with feedback connections specifically designed to model sequences. These NNs are computationally more powerful and biologically more reasonable than feedforward NNs (no internal states). The feedback connections provide a memory of past activations to the RNN, which enables learning of the temporal dynamics of sequential data. RNN uses contextual information, making it powerful for mapping between input and output sequences. However, traditional RNNs have a problem called vanishing or exploding gradient. Hochreiter and Schmidhuber (1997) proposed long short-term memory (LSTM) to resolve such problems.

Memory blocks substitute hidden units in LSTM that contain self-connected memory cells and three multiplicative units (input, output, and forget gates). The gates enable reading, writing, and resetting operations in the memory block and control the behavior of the memory block. Figure 11.2 shows a diagram representing a single LSTM unit.

Let c_t be the sum of inputs at time step t and its previous time step activations, the LSTM updates for time step i given inputs x_t, h_{t-1} and c_{t-1} are (Donahue et al. 2015) as follows:

$$i_t = \sigma(W_{xi}.x_t + W_{hi}.h_{t-1} + W_{ci}.c_{t-1} + b_i) \tag{11.3}$$

$$f_t = \sigma\left(W_{xf}.x_t + W_{hf}.h_{t-1} + W_{cf}.c_{t-1} + b_f\right) \tag{11.4}$$

$$c_t = i_t.\tanh(W_{xc}.x_t + W_{hc}.h_{t-1} + b_c) + f_t.c_{t-1} \tag{11.5}$$

$$o_t = \sigma(W_{xo}.x_t + W_{ho}.h_{t-1} + W_{co}.c_t + b_o) \tag{11.6}$$

$$h_t = o_t.\tanh(c_t) \tag{11.7}$$

where σ is an element-wise nonlinearity, such as a sigmoid function; W is the weight matrix; x_t is the input at time step t; h_{t-1} is the hidden state vector of the previous time step and b_i denotes the input bias vector.

11.3 Proposed Models

This study proposes three different network architectures based on simple NN, CNN, and RNN models. Figure 11.3 shows the architecture of the NN model with two hidden layers of 50 hidden units. The model takes a vector with eight variables as inputs and predicts the severity of traffic accidents as only property damage, possible/evident injury, or disabling injury/fatality. The total parameters of this network are 3225 distributed as 72, 450, 2550, and 153 for the network layers. The backpropagation algorithm trains the model using the Nadam optimizer and a batch size of four. The network parameters are selected via grid search and tenfold cross-validation assessments.

The second proposed model is based on a CNN, as shown in Fig. 11.4. In this model, the input data are transformed

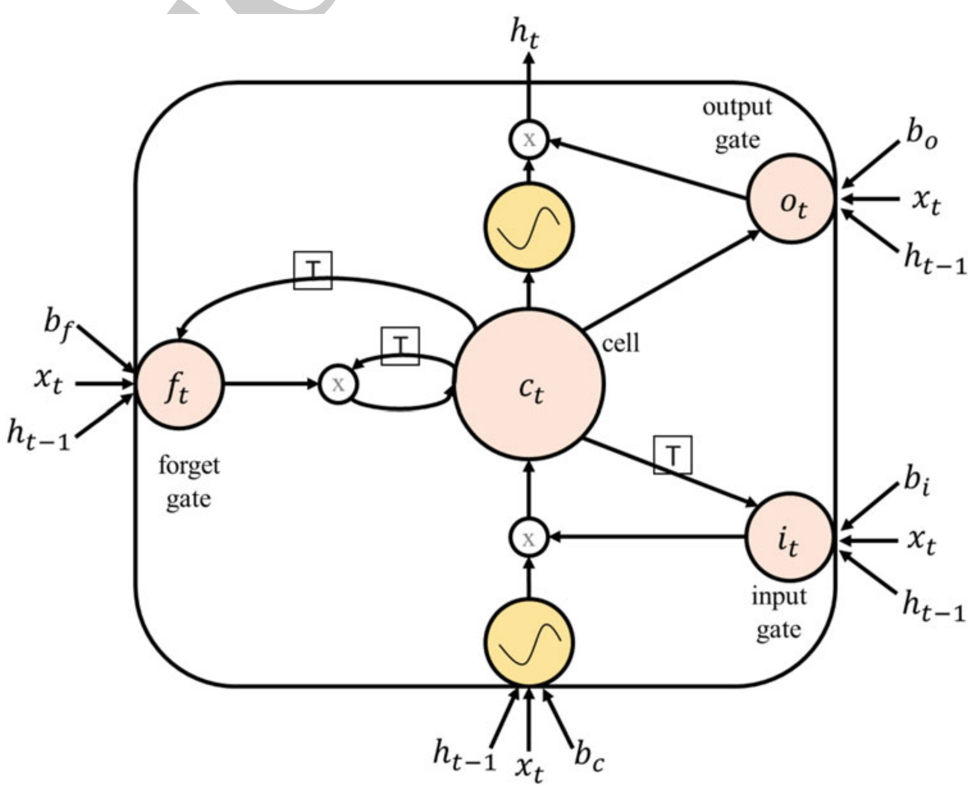

Fig. 11.2 Structure of a memory cell in LSTM-RNN

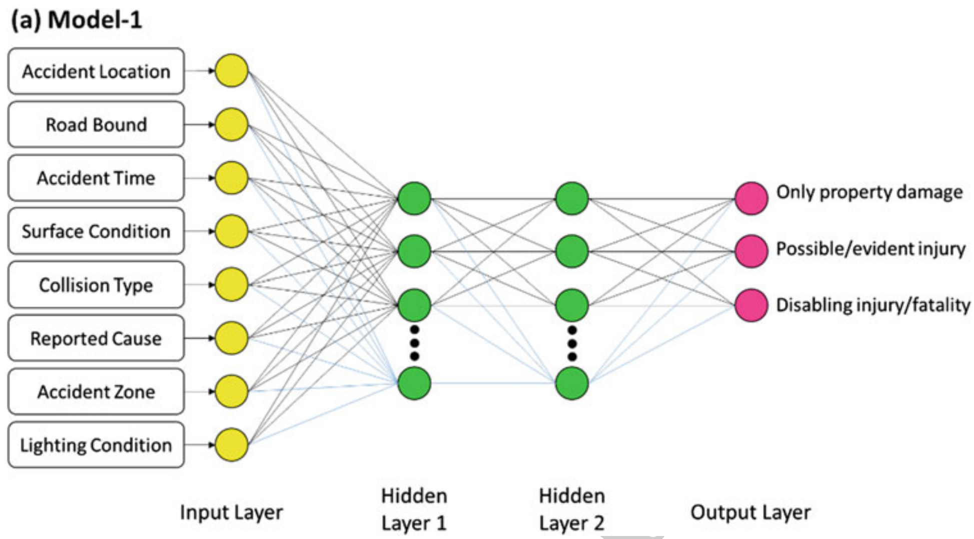

Fig. 11.3 Proposed NN model for predicting the injury severity of traffic accidents

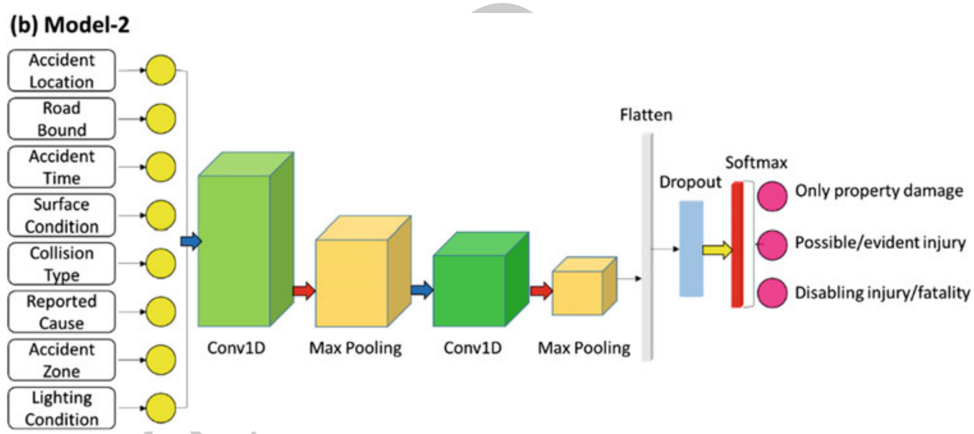

Fig. 11.4 Proposed CNN model for predicting the injury severity of traffic accidents

into a new feature representation by using convolution and pooling operations. A 1D convolution operation is applied to handle the sequence accident data. Maximum pooling operations are applied to abstract the rectified features. Then, the features are flattened to be used for classification. A dropout layer is added to avoid overfitting. Finally, the injury severity of traffic accidents is predicted by the softmax layer. The total number of parameters of the CNN model is 4739.

The third model is based on a RNN, which is designed for sequence problems. A RNN can be considered the addition of loops to the architecture. For example, in a given layer, each neuron may pass its signal later to forward to the next layer. The output of the network may feedback as an input to the network with the next input vector. Figure 11.5 shows the proposed network architecture based on the RNN model. Similarly, the network considers a vector of eight variables as inputs and produces probabilities for three severity classes of traffic accidents, namely only property damage, possible/evident injury, or disabling injury/fatality. The network consists of a LSTM layer with 100 hidden units, two fully connected layers, a dropout layer, and a softmax layer. The total number of parameters of the RNN model is 82,755. AQ2

A dropout layer is applied to avoid overfitting of the CNN and RNN models. NNs do not perform well when they are overfitted and when they are given new examples because they learn their weights from the training dataset. The dropout layer set randomly selected activations to zero. By doing so, the overfitting problem can be alleviated. This method is only used during training time and not during testing time. The number of dropout activations is controlled by the parameter known as keep probability.

11.4 Experimental Results

The models were implemented in Python using the open-source TensorFlow deep learning framework recently developed by Google (Abadi et al. 2016). TensorFlow

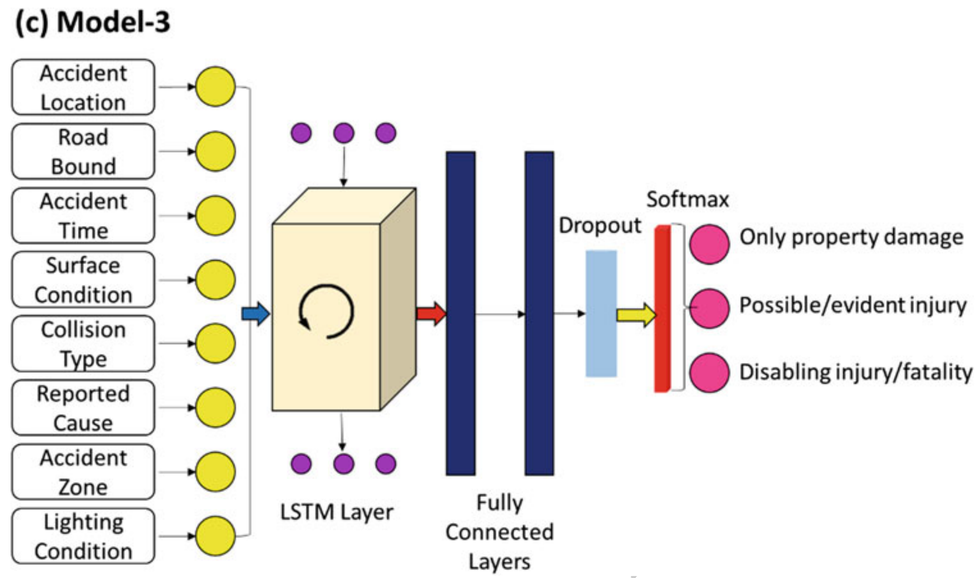

Fig. 11.5 Proposed RNN model for predicting the injury severity of traffic accidents

features automatic differentiation and parameter sharing capabilities, which allow a broad range of architectures to be easily defined and executed (Abadi et al. 2016).

11.4.1 Dataset

The 2009–2015 traffic accident data for the North–South Expressway (NSE), Malaysia were used in this study. NSE is the longest expressway (772 km) operated by Projek Lebuhraya Usaha Sama (PLUS) Berhad (i.e., the largest expressway operator in Malaysia) and links many major cities and towns in Peninsular Malaysia. The data were obtained from the databases of PLUS accident. The files used in this study were accident frequency and accident severity files in the form of Excel spreadsheet. The accident frequency file contains the positional and descriptive accident location and the number of accidents in each road segment of 100 m. The accident records were separated according to the road-bound (south and north). By contrast, the accident severity file contains the general accident characteristics, such as accident time, road surface and lighting conditions, collision type, and reported accident cause. Unique identity field (accident number) was used to link the two files.

The section of NSE employed in this study has a length of 15 km running from Ayer Keroh (210 km) to Pedas Linggi (225 km) (Fig. 11.6). The accident severity data show that the last section (220–225 km) of the NSE experienced the highest number of accidents that resulted in serious injury (82) compared with the other sections (Table 11.1). Most accidents occurred on the main route and southbound of the expressway. Actual accident causes were documented during accidents. The data show that lost control, brake failure, and obstacles are the main accident causes on NSE. Regarding lighting and surface conditions, most accidents occurred in daylight condition and dry road surface. The main collision types in the accident records are out of control and rear collision. In addition, the accident time factor shows that 91.68% of the accidents occurred during daytime. The data demonstrate that two-car accidents, single heavy car with an object and motorcycle with objects are mostly involved in the recorded crashes on NSE.

11.4.2 Model Performance

The proposed models were tested by a tenfold cross-validation method on the testing dataset. Table 11.2 and Fig. 11.7 show the accuracy of NN, CNN, and RNN models. The RNN model achieved the best accuracy (73.76%) compared with the CNN (70.30%) and NN (68.79%) models. However, the CNN model slightly outperformed the RNN model in terms of accuracy stability across different folds of the testing dataset. The standard deviation of the accuracies achieved by the CNN and RNN models are 0.53 and 1.24%, respectively. This result indicates the high stability of the CNN and the RNN models in predicting the injury severity of traffic accidents. The NN model exhibits low accuracy stability with a standard deviation of 2.21%.

11.4.3 Optimization and Sensitivity Analysis

Rather than simply using standard parameters, the network architecture should be optimized because deep learning models largely depend on data type and processing task. The

Layout: T3_Grey | Book ID: 454584_1_En | Book ISBN: 978-3-030-10373-6
Chapter No.: 11 | Date: 4-3-2019 Time: 2:32 pm | Page: 132/137

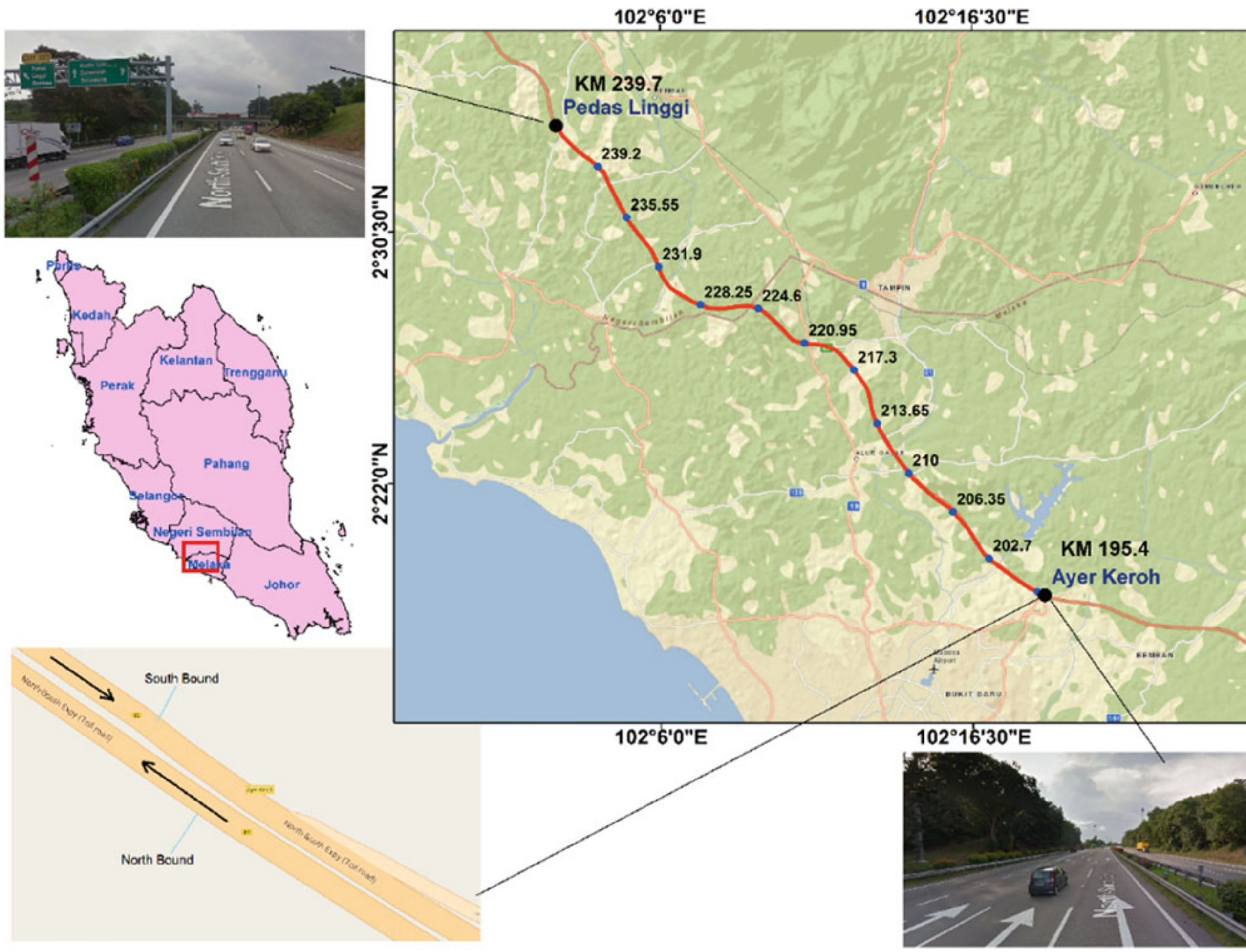

Fig. 11.6 Location of the NSE section analyzed in this study

data can vary in size, complexity, and the type of relationships between the predictors and dependent variable. Therefore, in this study, the architecture of the NN, CNN, and RNN was optimized via a grid search implemented in SciPy. This technique explores and finds the correct combinations of hyperparameters that can best predict injury severity of traffic accidents from the eight predictors given to the networks as inputs.

Nadam is found to be the best algorithm for the optimizer for all three models. The Nadam optimizer with its default parameters achieved the accuracy of 0.66, 0.70, and 0.73, for the NN, CNN, and RNN models, respectively (Fig. 11.8). Adam and Rmsprop optimizers also performed well for all the models. Although Adamax and Adadelta methods achieved relatively good accuracy in the CNN model, their performance was poor in the NN and RNN models. Overall, the Nadam algorithm is suggested to be used for analyzing traffic accident data.

Batch size refers to the number of training examples over which optimization update is computed. It has remarkable effects on the accuracy of the models. Figure 11.9 shows how the accuracy of NN, CNN, and RNN models changes when the batch size increases from 2 to 64. The performance of the NN model decreases with the increase of batch size, and the best accuracy was obtained when the batch size = 4. Similarly, the batch size of 4 achieved the best accuracy in the RNN model. However, the CNN model performed best with the batch size of 8.

In addition, due to a high number of parameters in the CNN and RNN models, dropout plays a major role in avoiding overfitting. Figure 11.10 shows the sensitivity analysis of the use of dropout layer with different keep probability parameters in the CNN and RNN models. The analysis shows that the best dropout rates for the CNN and RNN models are 0.2 and 0.5, respectively. This keep probability should be selected for each dataset and processing task by a grid search because selecting dropout rate has an important effect on the accuracy of deep learning models, and it is widely dependent on the number of parameters in those models.

Layout: T3_Grey Book ID: 454584_1_En Book ISBN: 978-3-030-10373-6
Chapter No.: 11 Date: 4-3-2019 Time: 2:32 pm Page: 133/137

Table 11.1 Driver injury severity distribution according to accident-related factors

Factor	Property damage only	Evident injury	Disabling injury	Total
Location				
210–214	185	172	58	415
215–219	234	47	56	337
220–225	238	58	82	378
Road-bound				
South	453	99	139	691
North	287	73	79	439
Accident zone				
Interchange	14	3	0	17
Junction				
Lay-by	2	0	1	3
Main route	666	155	209	1030
Northbound entry ramp	8	2	0	10
Northbound exit ramp	4	2	0	6
Rest and service area	21	4	2	27
Southbound entry ramp	2	0	1	3
Southbound exit ramp	7	1	3	11
Toll plaza	16	5	2	23
Reported accident cause				
Poor pavement condition	0	1	0	1
Brake failure	6	2	1	9
Bump–bump	37	12	27	76
Dangerous pedestrian behavior	0	0	1	1
Drunk	0	0	1	1
Loss of wheel	1	0	2	3
Lost control	75	18	22	115
Mechanical	5	1	0	6
Mechanical/Electrical Failure	11	0	1	12
Obstacle	43	12	6	61
Other poor driving	15	1	4	20
Other human factor/overload/overheight	3	0	0	3
Overspeeding	345	61	91	497
Parked vehicle	4	4	10	18
Skidding	1	0	0	1
Sleepy driver	134	44	42	220
Stray animal	13	1	2	16
Tire burst	47	15	8	70
Lighting condition				
Dark and with streetlight	47	6	8	61
Dark and without streetlight	225	74	89	388
Dawn/dusk	35	9	9	53
Daylight	433	83	112	628

(continued)

Table 11.1 (continued)

Factor	Property damage only	Evident injury	Disabling injury	Total
Surface condition				
Dry	460	146	190	796
Wet	280	26	28	334
Collision type				
Angular collision	9	2	0	11
Broken windscreen	2	0	0	2
Cross-direction	2	0	1	3
Head-on collision	0	1	4	5
Hitting animal	12	1	2	15
Hitting object on road	44	12	7	63
Others	20	0	6	26
Out of control	457	92	107	656
Overturned	33	11	7	51
Rear collision	137	48	81	266
Right-angle side collision	11	1	1	13
Sideswipe	13	4	2	19
Accident time				
Daytime	677	156	203	1036
Nighttime	63	16	15	94
Vehicle type				
Car–Bus	7	3	6	16
Car–Car	499	68	60	627
Car–Heavy car	51	11	14	76
Car–Motorcycle	4	7	22	33
Heavy car	131	23	25	179
Heavy car–Bus	2	3	3	8
Heavy car–Heavy car	24	9	15	48
Heavy car–Motorcycle	0	1	6	7
Heavy car–Taxi	2	0	0	2
Motorcycle	11	42	60	113
Motorcycle–Taxi	0	1	1	2
Motorcycle–Van	0	0	2	2
Taxi	1	0	1	2
Van	8	4	3	15

Table 11.2 Average cross-validation accuracy of the proposed models

Model	Tenfold cross-validation accuracy (%)
Model-1 Feedforward	68.79 (± 2.21)
Model-2 Convolutional	70.30 (± 0.53)
Model-3 Recurrent	73.76 (± 1.24)

Layout: T3_Grey Book ID: 454584_1_En Book ISBN: 978-3-030-10373-6
Chapter No.: 11 Date: 4-3-2019 Time: 2:32 pm Page: 135/137

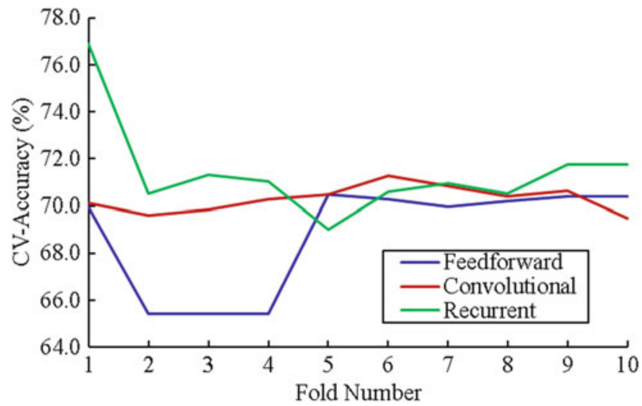

Fig. 11.7 Accuracy of the models tested by a tenfold cross-validation method

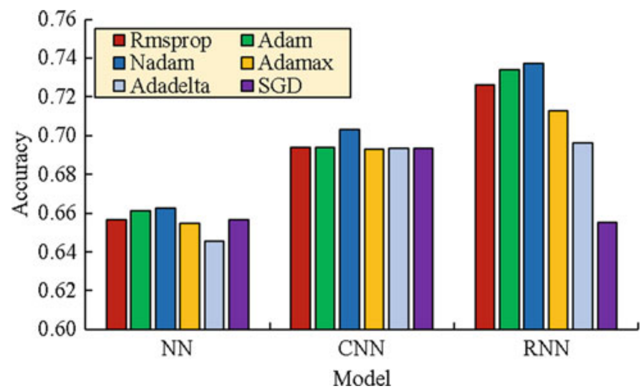

Fig. 11.8 Effects of optimization algorithm on the accuracy of the proposed models

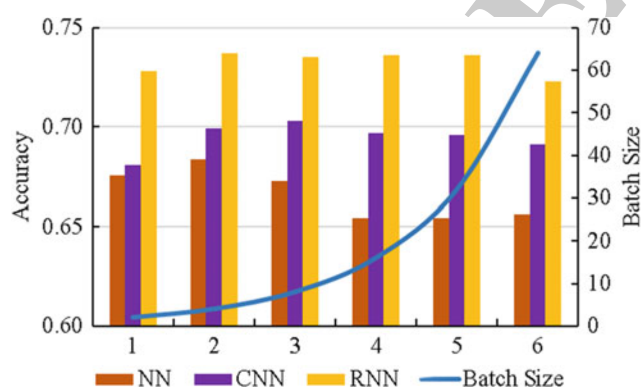

Fig. 11.9 Effects of batch size on the accuracy of the proposed models

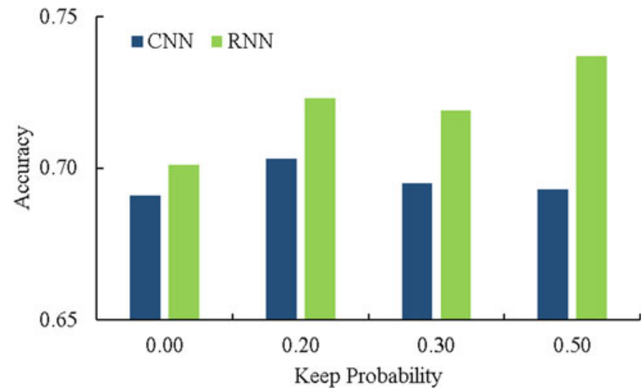

Fig. 11.10 Effects of the dropout rate on the CNN and RNN models

11.5 Discussion

In general, CNN models learn data representations to recognize patterns across space. That is, a CNN model is best used to recognize the components of an image or 2D array of numbers (e.g., line, curve, objects). However, when traffic accident features transformed from 1D vector into a 2D array, the CNN model can learn the spatial relationships among the features of different observations (accident events). Thus, this model is expected to perform better than the traditional NN model, which does not consider this additional feature in the data.

RNN models learn data representations to recognize patterns across time. Therefore, comparing the CNN and RNN models can reveal the critical component (space or time) for forecasting traffic accidents. In this study, the two models are found to be better than the NN model in terms of prediction accuracy; that is, the inclusion of additional spatial and time features can improve model performance than when they are not used at all. Having the accuracy of RNN found higher than that achieved by the CNN model shows that the temporal component of accident data is greater than the spatial structure of the data because variations in traffic, weather and driving conditions (traffic volume, the speed of vehicles, raining status) appear at different periods.

Considering that RNN models have memories where computations are derived from the earlier input are fed back into the network, they can find relationships among the accident events that are difficult to obtain when traditional methods or experts are used. Thus, these models automate

Layout: T3_Grey Book ID: 454584_1_En Book ISBN: 978-3-030-10373-6
Chapter No.: 11 Date: 4-3-2019 Time: 2:32 pm Page: 136/137

the feature identification and representation processes, which provide advantages to accident-forecasting models. However, RNN models require more complex training algorithms than those required by the CNN models, which often limit their applications, especially with limited datasets or data without temporal features (e.g., accident time).

11.6 Conclusion

Predicting injury severity of traffic accidents with high accuracy can improve the ability to manage roads efficiently and provide safe roads for drivers. This study explored the accuracy performance of three deep learning architectures (feedforward NN, CNN, and RNN). An initial network architecture was selected by an empirical analysis, and then the network hyperparameters were optimized through grid search method. Parameter sensitivity was analyzed to understand their effects on model accuracy.

The findings suggested that the RNN model performs best with an average accuracy of 73.76% compared with the CNN and NN models, which accuracies of 70.30 and 68.79%, respectively. The sensitivity analysis showed that Nadam is the best optimization technique for all the three models. The best batch sizes ranged from 4 to 8, and a dropout with 0.2 and 0.5 keep probability was found necessary for the CNN and RNN models, respectively.

Although this research shows that the deep learning models, such as RNN and CNN, can be promising tools for road safety assessment, few points need to be analyzed in future works. Firstly, the universal optimization of the networks needs to be established. Secondly, assessing the networks on large datasets can help transform deep learning models to the industry to be used in practice. Finally, models, such as RNN and CNN, can be integrated into a unified deep learning framework to help other applications of road safety assessment.

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., & Ghemawat, S. (2016). Tensorflow: Large-scale machine learning on heterogeneous distributed systems. arXiv preprint arXiv:1603.04467.

Al-Ghamdi, A. S. (2002). Using logistic regression to estimate the influence of accident factors on accident severity. *Accident Analysis and Prevention, 34*(6), 729–741.

Baczyński, D., & Parol, M. (2004). Influence of artificial neural network structure on quality of short-term electric energy consumption forecast. *IEE Proceedings-Generation, Transmission and Distribution, 151*(2), 241–245.

Delen, D., Sharda, R., & Bessonov, M. (2006). Identifying significant predictors of injury severity in traffic accidents using a series of artificial neural networks. *Accident Analysis and Prevention, 38*(3), 434–444.

Donahue, J., Anne Hendricks, L., Guadarrama, S., Rohrbach, M., Venugopalan, S., Saenko, K., & Darrell, T. (2015). Long-term recurrent convolutional networks for visual recognition and description. In *Proceedings of the IEEE Conference on Computer Vision and Pattern recognition* (pp. 2625–2634).

Gardner, M. W., & Dorling, S. R. (1998). Artificial neural networks (the multilayer perceptron)—A review of applications in the atmospheric sciences. *Atmospheric Environment, 32*(14), 2627–2636.

Graves, A., Mohamed, A. R., & Hinton, G. (2013, May). Speech recognition with deep recurrent neural networks. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 6645–6649).

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation, 9*(8), 1735–1780.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems* (pp. 1097–1105).

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature, 521*(7553), 436–444.

Li, X., Lord, D., Zhang, Y., & Xie, Y. (2008). Predicting motor vehicle crashes using support vector machine models. *Accident Analysis and Prevention, 40*(4), 1611–1618.

Li, Z., Liu, P., Wang, W., & Xu, C. (2012). Using support vector machine models for crash injury severity analysis. *Accident Analysis and Prevention, 45,* 478–486.

Lv, Y., Tang, S., & Zhao, H. (2009, April). Real-time highway traffic accident prediction based on the k-nearest neighbor method. In *ICMTMA'09. IEEE International Conference on Measuring Technology and Mechatronics Automation, 2009* (Vol. 3, pp. 547–550).

Ma, X., Yu, H., Wang, Y., & Wang, Y. (2015). Large-scale transportation network congestion evolution prediction using deep learning theory. *PLoS ONE, 10*(3), e0119044.

Mia, M. M. A., Biswas, S. K., Urmi, M. C., & Siddique, A. (2015). An algorithm for training multilayer perceptron (MLP) for Image reconstruction using neural network without overfitting. *International Journal of Scientific & Technology Research, 4*(2), 271–275.

Moghaddam, F. R., Afandizadeh, S., & Ziyadi, M. (2011). Prediction of accident severity using artificial neural networks. *International Journal of Civil Engineering, 9*(1), 41.

Mokhtarzade, M., & Zoej, M. V. (2007). Road detection from high-resolution satellite images using artificial neural networks. *International Journal of Applied Earth Observation and Geoinformation, 9*(1), 32–40.

Sameen, M. I., & Pradhan, B. (2017a). A two-stage optimization strategy for fuzzy object-based analysis using airborne LiDAR and high-resolution orthophotos for urban road extraction. *Journal of Sensors*.

Sameen, M. I., & Pradhan, B. (2017b). Assessment of the effects of expressway geometric design features on the frequency of accident crash rates using high-resolution laser scanning data and GIS. *Geomatics, Natural Hazards and Risk, 8*(2), 733–747. https://doi.org/10.1080/19475705.2016.1265012.

Sameen, M. I., & Pradhan, B. (2017c). A simplified semi-automatic technique for highway extraction from high-resolution airborne LiDAR data and orthophotos. *Journal of the Indian Society of Remote Sensing, 45*(3), 395–405. https://doi.org/10.1007/s12524-016-0610-5.

Sameen, M. I., Pradhan, B., Shafri, H. Z. M., Mezaal, M. R., & Hamid, H. (2016). Integration of ant colony optimization and object-based analysis for LiDAR data classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 10*(5), 2055–2066. https://doi.org/10.1109/JSTARS.2017.2650956.

Sarikaya, R., Hinton, G. E., & Deoras, A. (2014). Application of deep belief networks for natural language understanding. *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP), 22*(4), 778–784.

Schindler, K., & Van Gool, L. (2008, June). Action snippets: How many frames does human action recognition require?. In *IEEE Conference on Computer Vision and Pattern Recognition, 2008. CVPR 2008* (pp. 1–8).

Yang, G. Y. C. (1995). *Geological mapping from multi-source data using neural networks*. Calgary: Geomatics Engineering, University of Calgary.

Author Query Form

Book ID :	**454584_1_En**
Chapter No :	**11**

Please ensure you fill out your response to the queries raised below and return this form along with your corrections.

Dear Author,
During the process of typesetting your chapter, the following queries have arisen. Please check your typeset proof carefully against the queries listed below and mark the necessary changes either directly on the proof/online grid or in the 'Author's response' area provided below

Query Refs.	Details Required	Author's Response
AQ1	Please check whether the edit made in the sentence 'Max, average…available' conveys the intended meaning.	
AQ2	Please check whether the edit made in the sentence 'In this model…pooling operations' conveys the intended meaning.	