Spatiotemporal prediction of unsafe road conditions using weather and infrastructure information

Nirmal Sitaldin*[1], Bart van Arem[1], Maaike Snelder[1], and Shadi Sharif Azadeh[1]

[1]Department of Transport & Planning, Delft University of Technology, The Netherlands

Short summary

This paper presents the spatiotemporal prediction results of unsafe road conditions (URC) across the Dutch highway network, based on sparse, historical multi-source data. Climate change increases the risk of extreme weather-related URCs on road networks, which can cause disruptions and road user safety risks. Hence, the objective of this paper is to develop a data-driven model for the spatiotemporal prediction of URCs for highway networks. We propose a Graph Convolutional Network – Bidirectional Long Short Term Memory Network (GCN-BLSTM) for this goal, and benchmark this model against state-of-the-art deep learning algorithms. The GCN-BLSTM shows overall, higher predictive performance, with road ponding and roadside fires being predicted most accurately among the URCs. This study highlights the predictive potential of a hybrid spatiotemporal model for multi-URC predictions, compared to state-of-the-art data-driven methods, and proposes implications for future research on multi-URC prediction.
Keywords: GCN-BLSTM, Spatiotemporal Prediction, Deep Learning, Extreme Weather, Road Network.

1 Introduction

As climate change is expected to accelerate, so are the chances of extreme weather-induced events across transport systems. For road networks, such events occur as a consequence of both weather factors and infrastructure (built-environment) conditions. These events can increase the likelihood of short-term (e.g. vehicle accidents) and long-term impacts (e.g. decreased network connectivity) Verschuur et al. (2024), affecting road user safety as well Becker et al. (2022). Examples of such unsafe road conditions (URCs) are road ponding, roadside fires, hydroplaning, and infrastructure damage. Given an expected increase in the likelihood of URCs due to climate change, road authorities can benefit from knowing which parts of the network are most susceptible to URCs. With information on prospective susceptibility of links, road authorities can prioritize time, financial, and human resources towards the reinforcement of these links to minimize the financial, road user safety, and operational risks.

Therefore, the objective of this study is to research the extent to which a spatiotemporal learning model can accurately predict road ponding, roadside fires, hydroplaning, and infrastructure damage across the Dutch highway network. The secondary objective is to then identify links in the network that showcase frequent, high likelihoods of URCs, or "hotspots". Though URCs are rising in the Netherlands, these events are still relatively rare compared to non-weather related accidents. Thus, we have obtained a sparse, historical dataset of URC occurrences, including weather and infrastructure features. This data is used to train and validate the proposed model and benchmarks for this study.

We propose a GCN-BLSTM to capture spatial correlations from neighbouring URCs in the network on one hand, and the weather patterns over time on the other hand. The model receives spatiotemporal data on temperature and precipitation rate, and distance of the URC occurrence to the built environment factors (pavement, roadside vegetation, waterworks) as input. The output of the model is a Softmax probability vector of each URC category occurring across the network.

Current literature on extreme weather-driven event (URC) prediction for road infrastructure is growing, but relatively new. A main distinction can be made between simulation-based Ma et al. (2023); Zhang et al. (2021) and data-driven approaches, with an increase observed in the latter approach. We choose for a data-driven approach for its computational efficiency, ability to extrapolate predictions for the entire road network, and ability to learn non-linear patterns from

historical data between the input features and URC target.
This study contributes to literature as follows: First, our prediction framework is among the few to consider spatiotemporal learning for longer term prediction (at least 12 months prediction time-window), making our work more suitable for strategic planning. Previous data-driven frameworks considered either spatial Tariq et al. (2022) or temporal learning models X. Liu et al. (2023); Y. Liu et al. (2022) for short-term predictions, but rarely both. Secondly, we include both weather and infrastructure features as input. Despite the correlation between infrastructure conditions and road user safety (given that this study tackles URCs) Ahmed et al. (2013); Kocur-Bera & Dudzinska (2015), infrastructure features were rarely considered in past research. Lastly, we propose a multi-URC prediction framework for asset management. This is relevant to formulate reinforcement strategies from multiple (URC) risk perspectives for individual links.

2 Methodology

Proposed prediction model

The GCN-BLSTM model is proposed for its ability to capture spatial and temporal correlations for the prediction task at hand. The GCN layer processes the geospatial distance and infrastructure features of neighbouring URCs for the to-be-predicted URC. Aggregated information from surrounding incidents is expected to result in higher accuracy for local URC prediction. The URCs on the network are represented as a weighted, undirected graph $G = \{\mathcal{V}, A\}$, where $\mathcal{V} = \{v_1, ..., v_n\}$ represents the URCs, and $A \in \mathbb{R}^{n \times n}$ the adjacency matrix between these incidents based on infrastructure features and geolocation proximity. Given a multi-layered GCN, feature extraction of neighbouring points is conducted through aggregation, linear transformation, and non-linearity application per layer. Each layer executes these processes, with the layer-wise propagation rule of the GCN is defined as Kipf & Welling (2016):

$$H^{(l+1)} = \sigma(\tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}H^{(l)}W^{(l)}). \tag{1}$$

In Equation 1, $\tilde{A} = A + I_N$ is the weighted, adjacency matrix with self-connections through identity matrix I_N. $W^{(l)}$ is the trainable weight matrix of the previous GCN layer, $H^{(l)}$. σ denotes the non-linear activation function (Rectified Linear Unit). The initial GCN layer, or original URC representations, consists of the feature matrix $H^{(0)} = X$. Given the URC classification task, a two-layer GCN prediction model $f(X, A)$ can be defined as Kipf & Welling (2016):

$$f(X, A) = \text{Softmax}(\hat{A}\,\text{ReLU}(\hat{A}XW^{(0)})W^{(1)}) \tag{2}$$

In Equation 2, $\hat{A} = \tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}$ is the adjacency matrix renormalization trick, or preprocessing, $W^{(0)} \in \mathbb{R}^{d \times h}$ is the input-to-hidden trainable weights, given h hidden layers and d features per node. $W^{(1)} \in \mathbb{R}^{h \times n}$ is the hidden-to-output trainable weights for the n outputs.

The BLSTM is chosen for temporal processing, extracting patterns of URC features over time in both a chronological and reverse time order. More specifically, temporal weather patterns are relevant to process seasonality from precipitation and temperature information. For the URC prediction task, the BLSTM is expected to extract seasonal patterns more effectively than the LSTM, as observed in temporal prediction comparisons in literature (e.g. Roy et al. (2022), Adhikary et al. (2025)). Theoretically, the BLSTM processes two LSTM layers at time t in forward and backward hidden states, respectively notated as $\overrightarrow{h_t}$ and $\overleftarrow{h_t}$. Given T URC occurrences, $\overrightarrow{h_t}$ processes temporal dependencies of weather patterns from $t = 1$ to T, and $\overleftarrow{h_t}$ from $t = T$ to 1. $\overrightarrow{h_t}$ and $\overleftarrow{h_t}$ do not interact during the backpropagation through time (model training) phase, despite processing the same input data x_t. Both directions train separate weight matrices for their hidden states over time, $\overrightarrow{W_{h_t}}, \overleftarrow{W_{h_t}}$. The final output at t of the BLSTM architecture is comprised of the processing from both directions.
Formally, the GCN-BLSTM model is defined as follows: Given a $m-$layer GCN model $f_1(\bar{X}_t, A)$, $f_1(\bar{X}_t, A)$ receives a batch of 30 (bootstrapped), chronologically ordered URC datapoints X_t, where $\bar{X}_t = [X_1, ..., X_{29}, X_{30}]$. Alongside $\bar{X}_t$, $f_1(\bar{X}_t, A)$ considers the global adjacency matrix A as input for the entire road network, accounting for the spatial feature embeddings. The output of the final GCN layer can be noted as $\hat{X}_t$, which are the URC node embeddings after going through m convolutional filters. $\hat{X}_t$ is directly used as input by a $k-$layer BLSTM model $f_2(\hat{X}_t)$ to learn how the URC embeddings for all features evolve over time through forward and backward layers. The

BLSTM operations per LSTM cell for the forward processing forget $\overrightarrow{f_t}$, input $\overrightarrow{i_t}$, new candidate values $\overrightarrow{\tilde{c}_t}$, updated cell state $\overrightarrow{c_t}$, output gate $\overrightarrow{o_t}$, and cell output state $\overrightarrow{h_t}$ of the target URC at time t can be defined as:

$$\overrightarrow{f_t} = \sigma(\overrightarrow{W_f}[\overrightarrow{h_{(t-1)}}, f_1(\bar{X}_t, A)] + \overrightarrow{b_f}), \tag{3}$$

$$\overrightarrow{i_t} = \sigma(\overrightarrow{W_i}[\overrightarrow{h_{(t-1)}}, f_1(\bar{X}_t, A)] + \overrightarrow{b_i}), \tag{4}$$

$$\overrightarrow{\tilde{c}_t} = \tanh(\overrightarrow{W_{\tilde{c}_t}}[\overrightarrow{h_{(t-1)}}, f_1(\bar{X}_t, A)] + \overrightarrow{b_{\tilde{c}_t}}), \tag{5}$$

$$\overrightarrow{c_t} = \overrightarrow{f_t} \cdot \overrightarrow{c_{(t-1)}} + \overrightarrow{i_t} \cdot \overrightarrow{\tilde{c}_t}, \tag{6}$$

$$\overrightarrow{o_t} = \sigma(\overrightarrow{W_o}[\overrightarrow{h_{(t-1)}}, f_1(\bar{X}_t, A)] + \overrightarrow{b_o}), \tag{7}$$

$$\overrightarrow{h_t} = \overrightarrow{o_t} \cdot \tanh(\overrightarrow{c_t}). \tag{8}$$

The same operations are set for the backward layer variants. The k^{th} layer of $f_2(\hat{X}_t)$ outputs the concatenated forward and backward hidden states and bias of $\hat{X}_t$ based on $[X_1, ..., X_{30}]$, termed Y. The objective is to predict the 31^{st} URC datapoint at X_{30+1}. As our task requires URC classification (C classes), Y is passed to a fully-connected output layer D, combined with Softmax activation function, to output Softmax probabilities of each URC category for the target instance. D outputs C linear transformations of Y, or logits, denoted as $\tilde{z} = [z_1, ..., z_c]$. These are transformed by the Softmax activation function as follows Aggarwal et al. (2018):

$$Softmax(\tilde{z})_i = \frac{\exp z_i}{\sum_{j=1}^{C} \exp z_j}, \forall i \in \{1, ..., C\}. \tag{9}$$

The GCN-BLSTM model can be noted as $F = f_2(f_1(\bar{X}_t, A))$, a multi-class URC classification model that considers spatiotemporal dependencies. The objective of F during training is to minimize the categorical cross-entropy (CCE) loss value for each class prediction, defined as:

$$\min\ L(y; \hat{y}) = -\sum_{j=1}^{C} y \log \hat{y}. \tag{10}$$

In Equation 10, y is the true, one-hot encoded URC class, expressed as a binary value $[0, 1]$ for if the observation is actually the specified class ($y = 1$) or not ($y = 0$). $\hat{y}$ is the output of Equation 9, the Softmax probability of the prediction. The objective of F is to minimize Equation 10 per training iteration for all C URC classes, as a low CCE score implies effective discernment between classes by F.

Benchmarking and evaluation metrics

The proposed model is benchmarked against state-of-the-art spatial and temporal algorithms, namely LSTM, BLSTM, and GCN. To ensure a fair comparison, all models are experimented with the same hyperparameters and trained in 100 epochs or iterations. As the dataset in our context expresses URC as different categories of events, the models deal with a classification task. Hence, the evaluation metrics measure the models' ability to predict the True Positives and True Negative on a validation dataset. The metrics for this objective are Categorical Accuracy (Equation 11), Precision (Equation 12), and Recall (Equation 13). $\mathcal{TP}, \mathcal{TN}, \mathcal{FP}, \mathcal{FN}$ stand for respectively True Positives, True Negatives, False Positives, and False Negatives.

$$\text{Accuracy} = \frac{\mathcal{TP} + \mathcal{TN}}{\mathcal{TP} + \mathcal{TN} + \mathcal{FP} + \mathcal{FN}} \tag{11}$$

$\mathcal{TP}, \mathcal{FP}, \mathcal{TN}$ and $\mathcal{FN}$ in Equation 11 are aggregated across all the URC categories.

$$\text{Precision} = \frac{\mathcal{TP}}{\mathcal{TP} + \mathcal{FP}} \tag{12}$$

$$\text{Recall} = \frac{\mathcal{TP}}{\mathcal{TP} + \mathcal{FN}} \tag{13}$$

3 Results and discussion

Data description

The dataset is provided by Rijkswaterstaat (RWS), the asset manager of the Dutch highway network, and contains URC incidents spanning from 2014 till 2022. Each datapoint represents a significant disturbance caused by a URC (e.g. vehicle accident, congestion). The target variable to predict is the URC category, which are road ponding, roadside fire, infrastructure damage,, hydroplaning, and "other" (unspecified weather-driven cause). Precipitation and temperature data are included at the time of incident. Built environment (infrastructure) conditions are represented by the distance of infrastructure factors (vegetation, waterworks, and pavement) to the URC location. After data preprocessing, the final dataset contained 4658 datapoints.

Clustering and bootstrapping

Density-Based Spatial Clustering of Applications with Noise (DBSCAN) is applied to segment the network into subnetworks based on the geographical distribution of URC occurrences. This is done to train the models and provide predictions even more locally, in addition to the GCN spatial correlation process. The clustering results in five subclusters of the network on which the models are trained and validated. As data-driven methods require sufficiently large amounts of data, bootstrapping with replacement is applied on each cluster. Table 1 provides an overview of the five cluster distributions and (bootstrapped) data sizes.

Table 1: Regional spread of clusters

Cluster	**Size (bootstrapped)**	**Regions**
Cluster 1	760 (10000)	Alkmaar, Flevoland, Apeldoorn, Breda, Tilburg, Ede, Eindhoven s-Hertogenbosch
Cluster 2	1060 (10000)	Amsterdam, Haarlem, Utrecht, Almere, Hilversum Purmerend, Amersfoort
Cluster 3	1281 (10000)	Leiden, Den Haag, Delft, Rotterdam, Spijkenisse Dordrecht, Barendrecht
Cluster 4	1235 (8000)	Goes, Schouwen, Venlo, Nijmegen, Arnhem, Enschede, Valkenswaard, Eindhoven (partially) Ommen, Lelystad (partially), Emmen, Leeuwarden, Groningen, Zwolle, Schagen
Cluster 5	322 (5000)	Limburg, Roermond, Maastricht

Data splitting

Each bootstrapped cluster is split into a training, tuning, and validation dataset based on temporal slicing: The training set for all clusters ranges from 01-01-2014 to 01-07-2020, hyperparameter tuning set ranges from 01-07-2020 - 01-07-2021, and the validation dataset for predictions covers the period from 01-07-2021 till 31-12-2022. Results of the validation set are presented in the remainder of this section.

URC prediction

The general model accuracy scores on the validation set can be viewed in Table 2. Analyzing Table 2 implies that the GCN-BLSTM outperforms the other algorithms in clusters 2, 3, 4, and 5, tying with the BLSTM in cluster 4. The BLSTM marginally outperforms the GCN-BLSTM in cluster 1. The LSTM and BLSTM also outperform the GCN in all clusters except the Limburg region

(cluster 5).

Table 2: Overall model categorical accuracy

	LSTM	**BLSTM**	**GCN**	**GCN-BLSTM**
Suburban region (Cluster 1)	0.53	**0.56**	0.37	0.53
Amsterdam - Utrecht (Cluster 2)	0.48	0.48	0.37	**0.53**
Randstad (Cluster 3)	0.55	0.49	0.52	**0.59**
Outskirts (Cluster 4)	0.47	**0.49**	0.37	**0.49**
Limburg (Cluster 5)	0.53	0.53	0.56	**0.62**

To obtain more information on the extent to which the model predictions differ between the various URCs on the validation set, Table 3 provides the Precision Scores (PS) per URC for each cluster. The PS measure how well the models are able to predict the true occurrences (TP) of each URC to the FP predictions. The Recall table (Table 4) presents the True Positive Rate (TPR), or ground truth predictions (TP) in the context of the total ground truth observations. Both PS and TPR can take a value between 0 and 1, with a higher score indicating better predictive ability
Considering the overall PS patterns, all models score the highest precision for road ponding and roadside fires, followed by infrastructural damage across all clusters. PS scores for these URCs range from $0.30 - 0.72$, $0.24 - 0.75$, and $0.11 - 0.56$ respectively. Regarding the TPR, the same observation is made: The models are able to correctly predict road ponding and roadside fires the best, from all observed instances of these URCs (FN + TP) in the validation dataset. The TPR for road ponding and roadside fires range between respectively $0.41 - 0.94$ and $0.33 - 0.87$ across all models and clusters. Slippery pavement, to some extent infrastructure damage in clusters 3 and 4, and other or unknown URC incidents are less accurately predicted in terms of PS and TPR.
From these results, it is concluded that road ponding and roadside fires are the best predicted URCs by the models. An explanation for the less accurate predictions of slippery pavement and infrastructure damage is the relevance of input features: We expect that other features are more relevant predictors for these URCs, such as pavement material type and road age for infrastructure damage, and road friction index and driver speed for slippery pavement related incidents. Additionally, these URCs were less frequent in the dataset, compared to road ponding and roadside fires. Overall, the GCN-BLSTM outperforms the other models in URC predictions. This could be expected from the training and tuning observations as well. The lower performance of the GCN implies that spatial correlations among neighbouring datapoints do not contribute to effective learning in this context. When combined with temporal learning, the spatial information marginally ($\sim 5\%$) improves prediction. From a geographical perspective, clusters 1, 2, and 5 show the highest accuracy for URC predictions. This is unsurprising, as clusters 1 and 2 make up the busiest roads in the highway network (the Randstad region, Amsterdam, Utrecht). These regions are thus represented more in the dataset. Interestingly, URC predictions in Limburg are comparatively accurate.

4 Conclusions

The objective of this study was to assess the extent to which data-driven methods can accurately predict URCs based on sparse, multi-source historical data. To achieve this, we proposed a hybrid GCN-BLSTM model for spatiotemporal prediction and benchmarked this with LSTM, BLSTM, and GCN. Unlike previous work, we focused on long-term prediction, included weather and infrastructure input, and conducted spatiotemporal learning.
The findings of this study show that overall, spatiotemporal learning yields more accurate URC predictions compared to exclusively spatial or temporal learning. The GCN-BLSTM outperformed the benchmarks, with road ponding and roadside fires being the most accurately predicted URCs. Prediction accuracy also differed per region, depending on the incident density caused by how busy the subregional networks are. Hence, predictions for the Randstad, Amsterdam, and Utrecht were more accurate than for the suburban and rural parts of the networks.
Despite the application novelty, future research should validate the relevance of built-environment surroundings for URC predictions. In addition, building spatiotemporal models with more relevant input features for the targeted URCs is important. It would be interesting to see if applying one

Table 3: Precision scores for each URC across clusters

Region	URC Type	LSTM	BLSTM	GCN	GCN-BLSTM
Suburban region (Cluster 1)	Infrastructure damage	**0.69**	0.51	0	0.28
	Other	0	0	0	**0.09**
	Road ponding	0.44	**0.59**	0	**0.59**
	Roadside fire	0.68	**0.69**	0.37	0.67
	Slippery pavement	0.02	0.01	0	**0.22**
Amsterdam - Utrecht (Cluster 2)	Infrastructure damage	0.39	0.43	0	**0.56**
	Other	0	**0.31**	0	0.16
	Road ponding	0.54	0.52	**0.62**	0.58
	Roadside fire	0.44	**0.54**	0.24	**0.54**
	Slippery pavement	0.45	0.41	0	**0.49**
Randstad (Cluster 3)	Infrastructure damage	0.13	0.12	0	**0.31**
	Other	0.06	0.21	0	**0.28**
	Road ponding	0.60	0.59	0.60	**0.64**
	Roadside fire	0.58	0.52	0.39	**0.64**
	Slippery pavement	0	0	0	**0.42**
Outskirts (Cluster 4)	Infrastructure damage	0	**0.51**	0	**0.50**
	Other	0	0	0	**0.13**
	Road ponding	0.41	0.43	**0.57**	0.48
	Roadside fire	**0.67**	0.62	0.30	0.54
	Slippery pavement	0.09	0.21	0	**0.36**
Limburg (Cluster 5)	Infrastructure damage	0.14	**0.22**	0	0.11
	Other	0	**0.44**	0	0.29
	Road ponding	0.56	0.69	0.56	**0.72**
	Roadside fire	0.75	0.52	0	**0.75**
	Slippery pavement	0	0	0	0

Table 4: Recall or True Positive Rate scores for each URC across clusters

Region	URC Type	LSTM	BLSTM	GCN	GCN-BLSTM
Suburban region (Cluster 1)	Infrastructure damage	0.19	**0.41**	0	0.17
	Other	0	0	0	**0.06**
	Road ponding	0.57	0.57	0	**0.58**
	Roadside fire	0.68	0.8	1.00	**0.81**
	Slippery pavement	0.02	0.04	0	**0.47**
Amsterdam - Utrecht (Cluster 2)	Infrastructure damage	0.10	0.22	0	**0.46**
	Other	0	**0.12**	0	0.06
	Road ponding	0.54	0.74	0.75	**0.76**
	Roadside fire	0.44	0.68	**0.9**	0.6
	Slippery pavement	0.45	0.41	0	**0.56**
Randstad (Cluster 3)	Infrastructure damage	0.02	0.09	0	**0.11**
	Other	0.05	0.17	0	**0.29**
	Road ponding	**0.87**	0.77	**0.87**	**0.87**
	Roadside fire	0.67	0.54	0.58	**0.68**
	Slippery pavement	0	**0.11**	0	0.06
Outskirts (Cluster 4)	Infrastructure damage	0	0.23	0	**0.34**
	Other	0	0	0	**0.03**
	Road ponding	0.41	**0.70**	0.54	0.67
	Roadside fire	0.67	**0.82**	1.00	0.67
	Slippery pavement	0.09	0.11	0	**0.29**
Limburg (Cluster 5)	Infrastructure damage	0.07	**0.18**	0	0.08
	Other	0	**0.22**	0	0.18
	Road ponding	**0.94**	0.74	1.00	0.86
	Roadside fire	0.33	0.63	0	**0.80**
	Slippery pavement	0	0	0	0

model for multi-URC prediction in different case studies, leads to even higher accuracy among all URCs.

References

Adhikary, A., Palas, M. R., Das, S. C., & Hasib, A. (2025). Enhancing maritime safety through blstm-driven vessel trajectory prediction with advanced data reconstruction. In *2025 international conference on quantum photonics, artificial intelligence, and networking (qpain)* (p. 1-6). doi: 10.1109/QPAIN66474.2025.11171664

Aggarwal, C. C., et al. (2018). *Neural networks and deep learning* (Vol. 10) (No. 978). Springer.

Ahmed, I., et al. (2013). Road infrastructure and road safety. *Transport and Communications Bulletin for Asia and the Pacific*, *83*, 19–25.

Becker, N., Rust, H. W., & Ulbrich, U. (2022). Weather impacts on various types of road crashes: a quantitative analysis using generalized additive models. *European transport research review*, *14*(1), 37.

Kipf, T. N., & Welling, M. (2016). Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.

Kocur-Bera, K., & Dudzinska, M. (2015). Roadside vegetation—the impact on safety. *Proc. Intl. Sci. Con.(13)*, 594–600.

Liu, X., Chen, F., Xiao, Y., Wu, X., Zhang, J., & Wang, C. (2023). Urban waterlogged road detection under rainfall scenarios. In *2023 ieee 14th international conference on software engineering and service science (icsess)* (p. 285-289). doi: 10.1109/ICSESS58500.2023.10293026

Liu, Y., Zhang, W., Yan, Y., Li, Z., Xia, Y., & Song, S. (2022). An effective rainfall–ponding multi-step prediction model based on lstm for urban waterlogging points. *Applied Sciences*, *12*(23). Retrieved from `https://www.mdpi.com/2076-3417/12/23/12334` doi: 10.3390/app122312334

Ma, F., Ao, Y., Wang, X., He, H., Liu, Q., Yang, D., & Gou, H. (2023). Assessing and enhancing urban road network resilience under rainstorm waterlogging disasters. *Transportation Research Part D: Transport and Environment*, *123*, 103928. Retrieved from `https://www.sciencedirect.com/science/article/pii/S1361920923003255` doi: https://doi.org/10.1016/j.trd.2023.103928

Roy, D., Ishizaka, T., Mohan, C. K., & Fukuda, A. (2022). Detection of collision-prone vehicle behavior at intersections using siamese interaction lstm. *IEEE Transactions on Intelligent Transportation Systems*, *23*(4), 3137-3147. doi: 10.1109/TITS.2020.3031984

Tariq, A., Shu, H., Siddiqui, S., Munir, I., Sharifi, A., Li, Q., & Lu, L. (2022). Spatio-temporal analysis of forest fire events in the margalla hills, islamabad, pakistan using socio-economic and environmental variable data with machine learning methods. *Journal of Forestry Research*, *33*(1), 183–194.

Verschuur, J., Fernández-Pérez, A., Mühlhofer, E., Nirandjan, S., Borgomeo, E., Becher, O., ... Hall, J. W. (2024, 04). Quantifying climate risks to infrastructure systems: A comparative review of developments across infrastructure sectors. *PLOS Climate*, *3*(4), 1-21. Retrieved from `https://doi.org/10.1371/journal.pclm.0000331` doi: 10.1371/journal.pclm.0000331

Zhang, M., Xu, M., Wang, Z., & Lai, C. (2021). Assessment of the vulnerability of road networks to urban waterlogging based on a coupled hydrodynamic model. *Journal of Hydrology*, *603*, 127105. Retrieved from `https://www.sciencedirect.com/science/article/pii/S0022169421011550` doi: https://doi.org/10.1016/j.jhydrol.2021.127105