M. Shafik (Ed.)

doi:10.3233/ATDE240021

AI-Driven Real-Time Incident Detection for Intelligent Transportation Systems

Georgia Gkioka[a,1], Monica Dominguez [b], Athina Tympakianaki[b] and Gregoris Mentzas[c]
[a] *Researcher, Information Management Unit, Institute of Communication and Computer Systems (ICCS), National Technical University of Athens (NTUA)*
[b] *Scientific Researcher, Aimsun, Spain*
[c] *Professor, Information Management Unit, Institute of Communication and Computer Systems (ICCS), National Technical University of Athens (NTUA)*

Abstract. Efficient automatic detection of incidents is a well-known problem in the field of transportation. Non-recurring incidents, such as traffic accidents, car breakdowns, and unusual congestion, can have a significant impact on journey times, safety, and the environment, leading to socio-economic consequences. To detect these traffic incidents, we propose a framework that leverages big data in transportation and data-driven Artificial Intelligence (AI)-based approaches. This paper presents the proposed methodology, conceptual and technical architecture in addition to the current implementation. Moreover, a comparison of data-driven approaches is presented, the findings from experiments to explore the task using real-world datasets are examined, while highlighting limitations of our work and identified challenges in the mobility sector and finally suggesting future directions.

Keywords. incident detection, artificial intelligence, big data, machine learning, deep learning, intelligent transportation system.

1. Introduction

Automatic incident detection in intelligent transport systems (ITS) refers to the process of identifying incidents such as accidents, congestion, or road hazards in real-time using advanced technologies and data analysis techniques. Incidents are referring to “any non- recurring event that causes a reduction of roadway capacity or an abnormal increase in demand” [1]. Incident detection thus constitutes an essential component of ITS, given that if non-recurrent incidents could be detected in a timely manner, preventive measures and appropriate actions to respond to the incident could be rapidly taken. ITS incident detection systems typically use data from sources such as traffic cameras, sensors, GPS devices, and social media feeds to monitor traffic conditions and identify anomalies or patterns that indicate the occurrence of an incident. These systems can also use Machine Learning (ML) and Deep Learning (DL) algorithms to learn from past incidents and improve the accuracy of the detection.

[1] Georgia Gkioka, Institute of Communication and Computer Systems (ICCS), Patission 42, 10682 Athens,; E-mail: georgiagkioka@mail.ntua.gr.

The present paper introduces the conceptual and technical framework, methodology and work undertaken in developing a real-time automatic incident detection system. Moreover, the background and related works are mentioned, while outlining the identified research gaps which our work aims to fulfill. Then, the methodology and initial findings from preliminary experiments are discussed. Finally, the paper outlines the limitations of our work in addition to the challenges encountered and proposes future research directions.

2. Background and literature review

2.1. Incident detection task

When an incident occurs, the traffic undergoes a transition from an uncongested state to a congested state, leading to changes in vehicle speed and flow. Essentially, this transition creates a ripple effect known as a shockwave, which results in the formation of a queue after the bottleneck, typically at the location of the incident. This phenomenon is often visually represented in a space-time diagram, where the speed-time graph exhibits a cyclic pattern of acceleration and deceleration, as depicted in **Figure 1**. To effectively detect or predict an incident, it is crucial to consider anomalies that indicate the formation of queues both upstream and downstream from the incident location. Therefore, it is necessary to combine the time series data obtained from loop sensors with the spatial information of the traffic network.

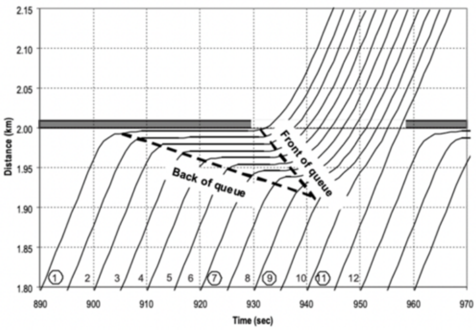

Figure 1. A time–space diagram for typical temporary capacity reduction (i.e., traffic accident) [2]

2.2. Relevant works

In recent years, research efforts have been proposed to deploy Automatic Incident Detection (AID) Systems onto urban roads, as effective incident detection in an automatic data-driven manner in ITS can improve traffic flow, reduce congestion and delays. Automatic Incident Detection Algorithms (AIDA), have been studied widely and are categorized by many systematic reviews (e.g., [3] [4] [5]). One such review categorizes these algorithms based on their data processing and methods used into four categories: comparative, statistical, artificial intelligence-based and video–image processing algorithms. [6]

In the type of comparative algorithms, the difference in the traffic flow parameters between two adjacent fixed detectors is calculated and compared. On the other hand, the statistical type of methods commonly employs the temporal characteristics of traffic flow data to build models based on the given statistical theory. While traditional approaches (comparative and statistical algorithms) have been widely used in the past because of their simplicity and are considered effective in detecting incidents which cause significant changes in the traffic flow, a shift has been observed towards AI-based approaches, because of their flexibility and their superior performance. Moreover, being data-driven, they can fit a large quantity of traffic flow data to mine their intrinsic patterns [7] and are able to capture both temporal and spatial information of traffic flow, a point proven important for improving the accuracy in incident detection [8]. Recently, a rise has been observed in research works which make use of video footage from CCTV cameras to identify incidents. Examples of such works which demonstrate the potential of using video streams and highlight the effectiveness of deep learning techniques in incident detection, are mentioned in various systematic reviews (e.g., [9] [10]). Based on several reviews of the domain, it is observed that despite progress being made, AI-based and generally data-driven incident detection algorithms are found to still have outstanding limitations and thus research gaps emerge. These gaps include obtaining a richer set of historical traffic incident data to train and test the models; constructing balanced datasets in which the number of incident samples equals the number of non-incident samples; improving the real-time capability of the models; and effectively extracting the spatial and temporal correlations to improve performance. [11].

3. Architecture

3.1. Conceptual architecture

In **Figure 2**, the conceptual architecture of the real-time incident detection is demonstrated. The input to our system comprises of network traffic data (i.e., speed, occupancy, flow) as measurements captured by Inductive Loop Detectors (ILD). The system then processes the data through a dedicated Machine Learning pipeline and is able to produce captured anomalies in both space and time dimensions, and thus to identify incidents in the network.

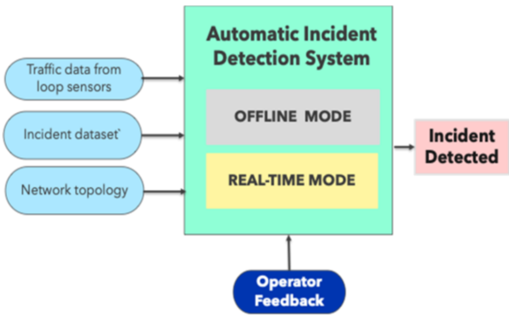

Figure 2. Conceptual architecture of our system.

The system consists of two main components to detect incidents specifically for roadways and highways: an offline training component and a real-time module. The main data source which is used as input to our system are measurements from inductive loop detectors, whose purpose is to continuously monitor traffic, and which are commonly used because of their reliability, cost-effectiveness, and easy integration into existing infrastructure. In the context of our current work, the data obtained from our system is analyzed using data-driven approaches to detect patterns and anomalies within these traffic observations.

3.2. Technical architecture

In Figure 3, the technical architecture of the offline training module of our system is presented. The Data Layer currently contains inductive loop detector (historical and real-time) measurements for speed, occupancy and flow in conjunction with the corresponding incident dataset. More information about these data sources and their pre-processing and feature engineering is included in Section 4. As part of the ML/DL module, a suite of Machine Learning algorithms has been implemented for incident detection. These include both Supervised and Unsupervised approaches and are discussed in Section 4. Leveraging techniques such as Grid Search and Random Search, we conducted comprehensive hyperparameter tuning to optimize the performance all those algorithms. We would like to emphasize the adoption of two sophisticated cross-validation techniques tailored for our system. Firstly, we employed Time-Series Cross-Validation, an essential validation strategy for incident detection given the inherent temporal nature of our data. Unlike traditional K-fold or stratified cross-validation, this method respects the chronological order of the data, mitigating the risk of future data leakage into the training setThe temporal sequence of the data is strictly maintained, ensuring each training set is only constituted by data points preceding those in the validation set. Secondly, to further fortify the robustness of our performance estimates, we utilized Repeated Cross-Validation. This technique entailed repeating the entire cross-validation process numerous times, with different random seed settings for each iteration. The primary advantage of this method lies in its capacity to minimize the variability of performance estimates, thus providing a more reliable representation of the model's prospective performance on unseen data. The combination of these techniques underscores a comprehensive and tailored validation approach, significantly enhancing the reliability of our incident detection model during its offline training phase.

Regarding the technical details, Python has been used for the development of the system. Moreover, several libraries, such as pandas, numpy, Tensorflow, Keras, scikit-learn have been utilized for data manipulation, model training, testing and evaluation, and seaborn and matplotlib for visualization purposes (e.g., as part of the exploratory data analysis (EDA)). The architecture of the system is designed to be scalable and flexible in terms of type of measurements derived from loop detector data, allowing for easy expansion and customization to meet the needs of different roadways and networks.

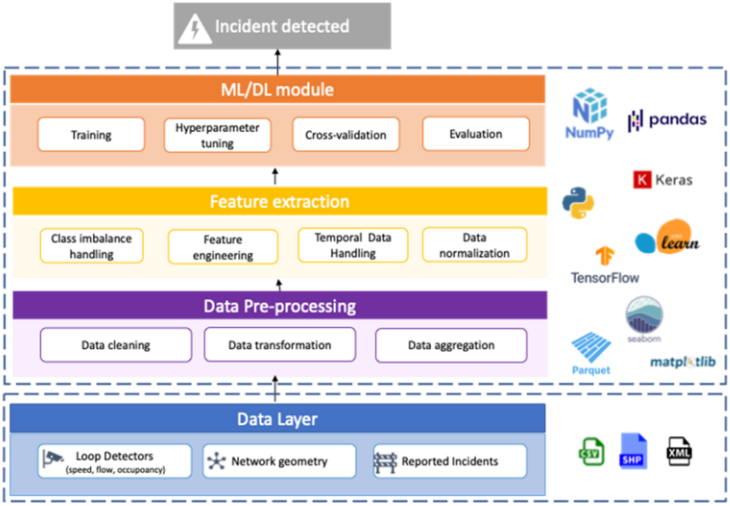

Figure 3. Technical architecture for the offline mode.

4. Implementation

4.1. Offline training mode

4.1.1. Inductive Loop detectors (ILD) dataset

As mentioned in the previous section, our system uses primarily traffic observations for ILDs. For the city of Athens, a corridor of Attiki Odos (a modern motorway extending along 70 km and constituting the ring road of the greater metropolitan area of Athens) extending from the Athens airport to the suburb of Metamorfosi has been used as a study area in evaluating the efficiency of our developed system. Loop detector data from 591 units were gathered from October 2020 to end of September 2021. These detectors have registered flow, occupancy, and speed in the original raw dataset. However, preliminary analysis of this data indicates that, as we will explain in more detail below, the raw data obtained from the sensors show many inconsistencies in measurements.

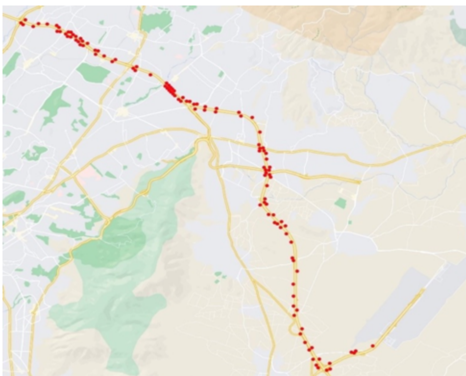

Figure 4. The network of our study area.

A detailed analysis of flow, occupancy and speed readings yield very low reliability for occupancy and speed. This estimation is based on statistical analysis of the time-series, unknown values (NANs), zeros, negative values and outliers. Out of the total 591 detectors provided, only 196 are regarded as reliable enough to be used as part of the experiments conducted. This led us to establish a direct communication with the data providers and have managed to obtain a list of the most reliable sensors, which are the ones included in our experiments and training. From the total amount of 26,331,086 readings provided (one every minute from the selected period), several filters were applied to remove detectors which were not in the station aggregation file, flow reliability outliers, flow-occupancy-speed mismatches, detectors with more than 50% not-a-number entries (NaNs), stuck values (constanxt readings across time), isolated values, and atypical profiles. Several types of imputation of missing/unreliable data were carried out on approximately 35% of the readings, namely: polynomial, time k-nearest neighbor (KNN), free-flow speed imputation, spatial KNN, PPCA-based imputation, and weekday-based imputation. After vigorous analysis, due to the low reliability in terms of occupancy and speed values, the variable selected to be used for experiments was the flow. Finally, the data have been transformed and stored in parquet files, each of which contains the monthly observations of one of the traffic characteristics.

4.1.2. Labeled incidents dataset

In addition to the Inductive Loop Detectors dataset, which comprises of the measurements of network-related attributes (i.e., speed, occupancy and flow), the labelled incidents dataset provided to us by Attikes Diadromes, the operator of our study area, plays a pivotal role in the experiments conducted for the purpose of automatic incident detection. This dataset, comprising 34,652 incident occurrences in total and 34 feature columns, serves as a critical resource for evaluating the performance of our models, as it represents the ground truth against which our models will be assessed. By leveraging this dataset, we can measure the accuracy and effectiveness of our incident detection techniques, enabling us to make informed decisions and ensure the quality of the obtained predictions.

The feature columns of this dataset include information regarding *'timestamp', 'source', 'start_time', 'end_time', 'direction', 'intersection', 'toll_station', 'branch', 'position_(pk)', 'type', 'subcategory', 'outcome', 'deaths', 'injured', 'queue_start_time', 'queue_end_time', 'queue_length_cars', 'queue_length_time', 'weather'* among others. However, it is worth noting that certain inconsistencies were identified within the dataset, based on a conducted Exploratory Data Analysis. Specifically, incidents that had no discernible impact on traffic were still labeled as incidents. To ensure fairness in our experiments, a filtering process has been implemented to remove such instances, thus maintaining consistency in the type of loop detector input data used for analysis, based on the following:

- Notably, it was observed that two specific branches of the highway recorded the highest number of incidents, with 13,829 and 13,757 incidents respectively. Since the majority of the incidents occurred on the main branches of the highway, a decision was made to exclusively focus on those.
- Moreover, a filtering process was applied to include only specific incident types for the scope of our experiments. Specifically, the labelled incidents dataset

exclusively encompasses incidents categorized as Traffic Congestion and Traffic Accident, as they are the primary focus of our investigation.

- Finally, the incidents were further filtered based on the observed queue length of cars. In collaboration with stakeholders, we obtained valuable feedback recommending a reduction in the threshold for queue length to 50 meters, as opposed to our initial proposal of 200 meters. This adjustment was made based on their expertise and supported by the understanding that queues of 200 meters are exceptionally uncommon in the specific highway, even in the event of an unplanned incident.

To summarize, after this filtering process, the dataset used primarily originates from a closed-circuit television (CCTV) system, encompassing a total of 1,786 incident occurrences for the two main branches and more specifically 763 reported incidents for the same time period as the traffic measurements. Following data cleaning and filtering, it was necessary to transform the dataset into a format suitable for utilization by Machine Learning algorithms, either for training purposes in the Supervised approaches or for evaluation purposes in the Unsupervised methods.

We would like to acknowledge the existence of certain limitations in the steps outlined, particularly concerning data filtering, as well as inherent limitations associated with the collected data itself. Several factors contribute to these limitations, which are discussed herein. Firstly, some incidents may not have been captured and registered within the dataset. Although our analysis indicates that all incidents were recorded manually, with most being identified through CCTV cameras, the potential for incomplete incident registration remains. Secondly, there is a possibility of inaccurate timing in the recorded incidents. It is feasible that an event occurring at a specific timestamp could be recorded or logged at a later timestamp. Such inaccuracies have notable repercussions on the evaluation of our algorithms. Lastly, the filtering process we employed is not immune to errors. While the selection criteria were based on the expertise of stakeholders and dataset characteristics, there is a chance that some incidents with significant traffic implications may have been inadvertently overlooked and not accounted for in our analysis.

4.1.3. Method Selection and Experiments

For the evaluation of the algorithms selected in our research, a one-month dataset from May 2021 was utilized. Our experiments encompassed both supervised and unsupervised techniques. The supervised techniques involved the utilization of labelled data for both training and testing, specifically employing Support Vector Machines (SVM). On the other hand, unsupervised techniques employed labelled data solely for testing purposes, and the algorithms evaluated included Isolation Forest, Convolutional Neural Network (CNN), Bayesian Convolutional Neural Network (BCNN), Wavelet Neural Network (WNN), Bidirectional Long Short-Term Memory (LSTM) in addition to the Aimsun Live's Incident Detection Module (IDM)[2]. Aimsun's IDM was used for benchmarking our experiments against a proprietary data-driven module which is integrated in the Aimsun Live solution for real-time transportation management and described in [12].

[2] Aimsun Live IDM is a component of a larger system that combines simulation and data-driven prediction modules. This component only uses data-driven techniques without any simulation.

The selection of input features for the Machine Learning algorithms was based on an extensive literature review of AI-based approaches and a thorough analysis of the available data features. Notably, flow emerged as the most reliable feature, as supported by rigorous analysis explained in Section 3. Building upon this finding, we have conducted experiments involving different combinations of temporal and spatial features, tailored to our use case, and after careful consideration, we selected the following features to serve as inputs for the Machine Learning algorithms: *Flow, Upstream and downstream flow for adjacent detectors, Mean upstream and downstream flow of detector {5, 10, 15} minutes before, Mean upstream and downstream flow of detector {5, 10, 15} minutes after, Mean upstream and downstream flow of adjacent detectors {5, 10, 15} minutes before, Mean upstream and downstream flow of adjacent detectors {5, 10, 15} minutes after*. For the deep learning algorithms, we have chosen five-time steps to make the sequences to be used as input.

Regarding the AI models which constitute the focus of our research work, we have carefully selected those algorithms based on insights from relevant studies and their demonstrated effectiveness in the field of traffic incident detection. Support Vector Machines (SVM) were chosen due to their documented advantages, including lower misclassification rates, higher correct detection rates, lower false alarm rates, and relatively faster detection times compared to other models [13]. Isolation Forests were incorporated into our approach, drawing from numerous works in the field of AID. They offer notable benefits, such as their efficient time complexity, low memory requirements, and the ability to effectively handle multi-dimensional feature spaces, mitigating the computational costs associated with distance calculation in various distance and density-based methods [14]. To capture time series patterns with time-varying period and intensity [15], we employed Wavelet transformation (WCNN), leveraging the PyWavelets library, an open-source wavelet transform software for Python, which allows the decomposition of the time series into a sum of frequency components, effectively capturing temporal dependencies and seasonalities. Incorporating Bayesian deep learning models can capture uncertainty over weights and activations using probabilistic layers trained through Bayesian inference and offer a straightforward extension of traditional DL models to better account for uncertainty and capture more nuanced relationships in the data [16]. Autoencoders, being unsupervised Machine Learning models, extract nonlinear features of traffic flow data and are generally used in anomaly detection tasks, since they try to minimize the reconstruction error as part of their training, thus can detect anomalies by checking the magnitude of the reconstruction loss [17]. Finally, the bidirectional Long Short-Term Memory (LSTM) model has been deemed suitable due to its ability to capture temporal autocorrelation within the data [14]. This architecture, trained on historical data, enables estimation of future values, and facilitates the classification of anomalous behavior using a threshold learnt from the data for loss values and comparing actual traffic data with corresponding patterns.

It is worth mentioning that for the case of Supervised Learning, since we have an extremely unbalanced dataset, we have made use of oversampling and under sampling techniques. In practice, the availability of traffic event samples of our dataset is considerably fewer compared to those of nonevent type, resulting in an imbalanced distribution between the two types of samples. Consequently, traffic incident detection can be viewed as a classification problem involving imbalanced data. The challenge of working with imbalanced datasets is that most machine learning techniques will ignore, and in turn have poor performance on, the minority class, although it is the performance on the minority class that is mostly important. To address this issue, the Synthetic Minority Oversampling Technique (SMOTE) [18] and Tomek link [19] is frequently employed. In our experiments, we have chosen to combine SMOTE with Tomek links technique, as it has been shown that this method is much superior compared with that of using only one of the two [20] [21].

To comprehensively evaluate and compare the performance of the aforementioned techniques, we employed commonly used metrics in AI-related works, namely precision, recall, and F1-score. These metrics have been widely adopted in the field due to their effectiveness in assessing algorithm performance [22]. It is important to note the interdependence of these metrics, where improvements in one metric may be accompanied by a degradation in others [23]. Table 1 shows results of the standard set of metrics used in our AI experiments, namely precision, recall and F1-score.

Table 1. Evaluation metrics.

Algorithm	Precision	Recall	F1-score
SVM	**0.58**	**0.97**	**0.64**
Isolation Forest	0.01	0.44	0.02
CNN	0.01	0.94	0.02
WNN	0.05	0.96	0.09
Autoencoder	0.03	0.49	0.05
Bidirectional LSTM	**0.19**	**0.43**	**0.26**
Aimsun IDM	0.08	0.50	0.14

The results obtained from our analysis exhibit some fluctuations across the employed techniques, highlighting the need for further investigation and analysis. It is important to remember that labelled incidents were limited to visible areas of the network, therefore the false positive rate is seriously affected. There is no certainty that false positives might be due to an invisible event to the data supplier (therefore, there was an event, but it was not labelled) or truly a faulty prediction by the algorithm. Moreover, it is crucial to consider that the evaluation dataset has been formatted in 5 minutes intervals to feed the algorithms, therefore when computing precision there is a true positive when the event is detected at exactly the annotated timestamp. However, in real scenarios an ideal incident detection algorithm should be able to spot an event ideally before it happens or at least within a reasonable time margin. In this light, among the algorithms tested, the Support Vector Machine (SVM) emerged as the top-performing method, demonstrating high precision and recall. This finding aligns with existing literature [24] [6], which suggests that SVM performs exceptionally well when a labelled incidents dataset is available, given its supervised nature. However, we should acknowledge that this approach may suffer from overfitting issues and limited generalizability to unseen samples. On the other hand, the Isolation Forest algorithm has managed to achieve a satisfactory recall but exhibited very low precision, resulting in a significant number of false positives and subsequently impacting the obtained F1-score. The Convolutional Neural Network (CNN) and Wavelet Neural Network (WNN) demonstrated high recall rates but struggled to achieve satisfactory precision. Nevertheless, the wavelet transformation exhibited slightly better performance than the CNN, confirming findings from literature where wavelet transformations successfully applied to time-series datasets [25]. For the Autoencoder, the results were consistent with other deep learning methods tested, indicating that its performance is comparable. Even though Aimsun's

IDM and Bidirectional LSTM showquite low performance per timestamp, when analyzing results on an event-based rationale (within a time margin of 15 minutes around the labelled event), we have been able to detect 11 events out of a total of 15 in the analyzed period (May 2021) yielding a recall of 73% which is an acceptable level of performance for non-recurrent events. However, one of the limitations is the fact that these techniques are bound to produce a high number of false positives as shown by the precision results.

4.2. Real-time operation

After having performed the training of the AI data-driven model in the offline mode as explained in the previous section, our system is able to operate in real-time to raise alerts. Figure 4 displays the process flow of the online module of our system. As soon as new data become available, the online module of our system captures it. In our case, the data is refreshed every minute, thus, the respective information is collected, stored locally and then aggregated in five-minute intervals to be fed in the pre-processing and data cleaning stage of the pipeline. The specific procedures for pre-processing remain consistent with those outlined in the offline mode of operation, maintaining uniformity in the approach to data preparation and cleaning. Then, the data are transformed in the required format to be fed in the step of model prediction. Should the entry contain anomalies (represented as "1"), then feedback is requested from operators, to confirm the identified incident. This human-in-the-loop concept is crucial, since it assists in creating a refined incident dataset and ensures that the system's performance could increase over time, given that it is retrained on this evolving dataset. It is worth mentioning that stakeholders can enhance the quality and accuracy of the reported incidents, by creating manual entries of identified incidents. Finally, in the case that the system has identified an anomaly in the data and labels it as incident, it then produces as output an entity of type "***Incident***" with the location and time attributes of the incident.

Moving forward, we're focusing on enhancing our incident detection model through Online and Batch Learning. Online Learning allows the model to learn incrementally with each new incident, offering a dynamic solution for handling class imbalance. However, we must account for potential instability due to data sequencing. Batch Learning, on the other hand, involves accumulating data before model retraining, and while it helps manage class imbalance, its slower adaptation to new incidents poses a challenge. Furthermore, the feedback loop which we have implemented to compare model predictions with actual outcomes is key to our continued improvement. Any discrepancies detected can then be harnessed to optimize the model via online learning or by being added to the next batch of training data.Regardless of the approach, implementing robust validation strategies and a feedback loop for comparing model predictions to actual outcomes is crucial.

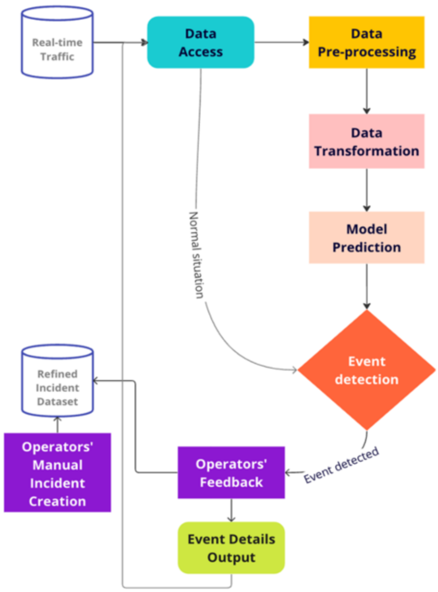

Figure 5. Process flow for the online mode.

5. Conclusion and Future steps

This work presents the conceptual and technical framework, methodology in addition to the implementation and preliminary findings in developing an automatic incident detection system. Our background research and our work has led us to identify several challenges regarding data-driven approaches in automatic incident detection. Firstly, the quality of the input data is crucial in AI-based approaches and integrating vigorous pre-processing techniques, while tackling missing values, needs to be taken care of to avoid inaccurate results. This is in line with literature in the domain of Machine Learning and data-driven studies and our findings confirm that. Moreover, we have observed a scarcity of established benchmarks for evaluating the efficacy of the models' performance; the difficulty of obtaining benchmarks at model level is understandable, however we deem that it would be possible to compare similar purpose algorithms which could be a suggested future direction in the field.

Regarding the limitations of our research, the results of our models are completely dependent on the data quality and reliability, and more specifically on the measurements of the detectors, and the incident dataset, the basis of our evaluation. The high number of false positives produced by the models is difficult to assess because maybe it is related to potential blind spots in the network which hinders the detection of incidents. One way to decrease the false alerts generated is to incorporate various types of data, such as CCTV cameras which is also the vision of our work for the future, to train the algorithms. Despite the abundance of available data and the advanced capabilities of machine learning algorithms, only a limited number of studies have effectively utilized the combination of multiple data sources, as stated by the review conducted by Kashinath et al. [26] .To bridge this research gap, as a future direction, we envisage to extend our work by integrating multimodal heterogenous data sources, such as CCTV video streams, potentially GPS data and floating vehicle data to enhance the performance of our system's detection capabilities. Furthermore, we are considering integrating online machine learning methods to update the models continuously and to ensure that they follow the patterns and trends of the most recent data. Our future work also entails incorporating supplementary real-world datasets which make use of inductive loop detectors in various use cases and perform more experiments to test our algorithms. Under this light, further processing would be required to extend our work for datasets either stemming from proprietary sources or openly available datasets (e.g., METR-LA, PEMS), and subsequently validation and further optimization of our models is expected.

In conclusion, the findings of our research highlight the significant potential of machine learning in automating the incident detection process, providing real-time alerts and insights to improve emergency response and public safety. By harnessing the power of artificial intelligence and advanced data analytics, organizations and authorities can proactively identify and respond to incidents, reducing response times and mitigating potential risks. Despite the certain limitations and challenges discussed as part of this paper, both our ongoing work and the future directions outlined as part of our research aim to address these limitations and ultimately push the boundaries of automatic incident detection, paving the way for a safer, more reliable, and efficient future of mobility.

Acknowledgment

This work is partly funded by the European Union's Horizon 2020 project under grant agreement No 955317. The work presented here reflects only the authors' view and the European Commission is not responsible for any use that may be made of the information it contains.

Bibliography

[1] P. Farradyne, "Traffic incident management handbook. Federal Highway Administration Office of Travel Management.," 2000.

[2] S. Hireche and A. Dennai, "Machine Learning Techniques for Road Traffic Automatic Incident Detection Systems: A Review," Springer Nature Switzerland AG 2020 M. Hatti (Ed.): ICAIRES 2019, LNNS 102, pp. 60–69., 2020.

[3] L. Li, Y. Lin, B. Du, F. Yang and B. Ran, "Real-time traffic incident detection based on a hybrid deep learning model,," Transportmetrica A: Transport Science, 18:1, 78-98, 2022.

[4] J. Evans and A. V. Rhys, "Improving road incident detection algorithm performance with contextual data," 2020.

[5] O. ElSahly and A. Abdelfatah, "Systematic Review of Traffic Incident Detection Algorithms.," Sustainability, vol. 14, p. 14859, 2022.

[6] T. Huang, S. Wang and A. Sharma, "Highway crash detection and risk estimation using deep learning," Accident Analysis & Prevention, vol. 135, p. 105392, 2020.

[7] T. L. Pan, A. Sumalee, R. X. Zhong and N. Indra-payoong, "Short-Term Traffic State Prediction Based on Temporal–Spatial Correlation," IEEE Transactions on Intelligent Transportation Systems, vol. 14, no. 3, pp. 1242-1254.

[8] X. Ma, Z. Tao, Y. Wang, H. Yu and Y. Wang, "Long short-term memory neural network for traffic speed prediction using remote microwave sensor data," Transportation Research Part C: Emerging Technologies, vol. 54, pp. 187-197, 2015.

[9] R. Nayak, C. U. Pati and K. S. Das, "A comprehensive review on deep learning-based methods for video anomaly detection," Image and Vision Computing, vol. 106, 2021.

[10] F. Dion, R. H. and Y. Kang, " Comparison of delay estimates at under-saturated and over-saturated pre-timed signalized intersections," . Transportation Research Part B: Methodological, vol. 38, no. 2, p. 99–122, 2004.

[11] M. Akhtar and S. Moridpour, "A Review of Traffic Congestion Prediction Using Artificial Intelligence," Journal of Advanced Transportation, 2021.

[12] F. Torrent-Fontbona, M. Dominguez, J. Fernandez-Frade and J. Casas, "Towards Efficient Incident Detection in Real-time Traffic Management.," in Proceedings of the 4th Symposium on Management of Future Motorway and Urban Traffic Systems (MFTS 2022) , Dresden, 2022.

[13] F. Yuan and R. L. Cheu, "Incident detection using support vector machines, Transportation Research Part C: Emerging Technologies, Volume 11, Issues 3–4, Pages 309-328,," 2003.

[14] Y. Sun, T. Mallick, P. Balaprakash and J. Macfarlane, "A data-centric weak supervised learning for highway traffic incident detection," Accident Analysis & Prevention, vol. 176, 2022.

[15] S. Agarwal, P. Kachroo and E. Regentova, "A hybrid model using logistic regression and wavelet transformation to detect traffic incidents," Iatss Research, vol. 40, no. 1, pp. 56-63, 2016.

[16] G. Liu, H. Jin, J. Li, X. Hu and J. Li, "A Bayesian deep learning method for freeway incident detection with uncertainty quantification.," Accident Analysis & Prevention, vol. 176, p. 106796, 2022.

[17] H. Yang, Y. Wang, H. Zhao, J. Zhu and D. Wang, "Real-time Traffic Incident Detection Using an Autoencoder Model.," 2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC), pp. 1-6, 2020.

[18] N. V. Chawla, K. W. Bowyer, L. O. Hall and W. P. Kegelmeyer, "SMOTE: Synthetic Minority Over-sampling Technique," 2002.

[19] I. Tomek, "Two Modifications of CNN. IEEE Transactions on Systems, Man, and Cybernetics, vol. 6, no. 11, pp. 769–772.," 1976.

[20] M. Zeng, B. Zou, F. Wei, X. Liu and L. Wang, "Effective prediction of three common diseases by combining SMOTE with Tomek links technique for imbalanced medical data," 2016.

[21] E. F. Swana, W. Doorsamy and P. Bokoro, "settings Open AccessArticle Tomek Link and SMOTE Approaches for Machine Fault Classification with an Imbalanced Dataset," 2022.

[22] J. Zhou, A. Gandomi, F. Chen and A. Holzinger, "Evaluating the Quality of Machine Learning Explanations: A Survey on Methods and Metrics," Electronics, vol. 10, no. 5, p. 593, 2021.

[23] B. Ghosh and D. P. Smith, "Customization of automatic incident detection algorithms for signalized urban arterials," Journal of Intelligent Transportation Systems, vol. 18, no. 4, pp. 426-441, 2014.

[24] F. C. R. Yuan, "Incident Detection Using Support Vector Machines," Transportation Research Part C: Emerging Technologies, vol. 11, p. 309–328, 2003.

[25] Z. Zheng, S. Ahn, D. Chen and J. & Laval, "Applications of wavelet transform for analysis of freeway traffic: Bottlenecks, transient traffic, and traffic oscillations," Transportation Research Part B: Methodological, vol. 45, no. 2, pp. 372-384, 2011.

[26] S. Kashinath, S. Mostafa, A. Mustapha, H. Mahdin, D. &. M. M. Lim, M. Mohammed, B. Al-rimy, M. F. Md Fudzee and T. Yang, "Review of data fusion methods for Real-time and Multi-Sensor Traffic Flow Analysis," IEEE Access, 2021.