Automatic Accident Detection, Segmentation and Duration Prediction Using Machine Learning

Artur Grigorev, Adriana-Simona Mihăiţă, *Senior Member, IEEE*, Khaled Saleh, *Member, IEEE* and Fang Chen

***Abstract*— Traffic accidents are often inaccurately reported, with incorrect location and disruption duration due to various external factors. This can result in imprecise predictions and inaccurate decision-making in data-driven models. To address these challenges, our study presents a comprehensive framework for traffic disruption segmentation from traffic speed data (obtained from Caltrans Performance Measurements system) in the time-space proximity of reported accidents (from Countrywide Traffic Accident dataset). Furthermore, we evaluate multiple machine learning models on reported, estimated, and manually marked disruption intervals, and demonstrate that our enhanced modelling approach reduces the root mean squared error (RMSE) of traffic accident duration prediction while providing higher similarity with disruptions observed in traffic speed. Our algorithm yields higher disruption detection precision than reported accident timelines. Although using multiple segments offers a slight decrease in the quality of results, it highlights more disruptions. Future research could explore expanding the algorithm's complexity and applying it to improve traffic incident impact predictions.**

***Index Terms*— Traffic management, traffic operations, traffic safety, accidents, accident detection, performance evaluation, traffic simulation, level of services, machine learning.**

I. Introduction

TRAFFIC accidents are a significant concern worldwide, causing fatalities, injuries, and economic losses. The number of vehicles has been substantially increasing during the past decades, which currently leads to an increase in the number of traffic accidents [1]. The National Highway Traffic Safety Administration (NHTSA) reported more than 5 million traffic accidents happening in the United States during the year 2013 [2]. Traffic Management Agencies usually rely on Traffic Incident Management Systems (TIMS) to collect data on traffic accidents, including information on various accidents, traffic states and environmental conditions. Accurately predicting the total duration of an incident shortly after it is being finished, will help in improving the effectiveness of accident response by providing important information to decide the required resources to be allocated (response team size, equipment, traffic control measures) [3]. A traffic accident is a rare event with stochastic nature. The effect of the accident can be observed as an anomalous state in the time series of traffic flow [4].

Various terms and concepts are employed in the field of traffic accident duration prediction. Key terms include the Incident duration - The time between the occurrence of an incident and its clearance [5] and Predictive modelling - the process of developing data-driven models to forecast future outcomes, such as accident duration [6]. Important road safety concepts encompass that related to traffic incident duration prediction are the following: 1) Human factors: Elements related to driver behavior, such as attention, fatigue which affect decision-making [7], 2) Vehicle factors: aspects related to the vehicle itself, including design, maintenance, and safety features [8], 3) Infrastructure factors [9]: The design, construction, and maintenance of roads and their surroundings, 4) Traffic management [10]: Measures and strategies implemented to improve the efficiency, safety, and sustainability of road networks. In our research, we focus on a possible contribution to the field of traffic management by employing traffic speed disruption detection and performing traffic incident duration prediction with higher accuracy.

Challenges: The traffic accident analysis may be a challenging task due to incorrect or incomplete accident reports, including the set and the quality of the accident characteristics that have been reported. Accident reports can contain user-input errors related to the accident duration such as: 1) an approximate reporting of accident's start and end time 2) reporting of the accident start time could have been done after the incident finished in reality 3) a 'placeholder' accident duration reporting (filling report with the approximate duration value due to unavailability of data by the moment of reporting). In our previous research [11], [12] we found that timeline-related errors are present in accident reports across three different data sets from both Australia and the United States of America, which creates the possibility of observing that such errors can occur in other data sets from around the world as well, due to multiple human and technical factors that can arise. To forecast the accident impact it is crucial to have an accurate and correct data regarding the observed disruption timeline. We emphasise that disruptions observed in a recorded traffic state can be automatically segmented and associated with a reported accident at the same time and location as when the accident occurred, which

Manuscript received 22 September 2022; revised 28 April 2023 and 18 July 2023; accepted 7 September 2023. This work was supported in part by the Australian Research Council (ARC) Linkage Project under Grant LP180100114; and in part by iMOVE Co-operative Research Centre (iMOVE CRC) funded by the Cooperative Research Centers Program, an Australian Government Initiative. The Associate Editor for this article was Y. Kamarianakis. *(Corresponding author: Artur Grigorev.)*

The authors are with the Faculty of Engineering and IT, University of Technology Sydney, Sydney, NSW 2007, Australia (e-mail: Artur.Grigorev@ieee.org; khaled.saleh@ieee.org; fang.chen@uts.edu.au).

Digital Object Identifier 10.1109/TITS.2023.3323636

allows to eliminate user-input errors from reports and improve the accident duration prediction performance in many traffic management centres around the world. To help address this issue, in our paper we propose various methods for a correct traffic disruption segmentation, the method for an association between vehicle detector stations and accident reports.

Another important challenge is that many incident data sets around the world are private and not shared for public investigation; for those open data sets, there are several missing information fields, or even worse, incomplete information regarding the traffic conditions in the vicinity of the accidents. Even often publish crash data sets are limited in size as well and contain a very small number of records. This represents a tight constraint when testing one framework over multiple countries with different traffic rules and regulations. For our studies we have oriented our attention towards two big open data sets - CTADS (Countrywise traffic accident data set) which contains 1.5 million accident reports and the Caltrans Performance Measurement System (PeMS) which provides data on traffic flow, traffic occupancy and traffic speed across California. Despite both being extensive data sets, vehicle detector station readings from PeMS are not associated with traffic accident reports from CTADS either by time, location or coverage area. The lack of such association makes it impossible to analyse the relation between accidents and their effects on traffic flow and speed. To address this challenge, in our paper we introduce the following mapping algorithm which will secure several steps such as:

- an association of Vehicle Detection Stations (VDS) with reported accidents in their proximity,
- a segmentation of traffic speed disruptions from detector readings,
- an association of detector stations with reported accidents (we will further show that this step is necessary due to many detected user-input errors in accident reports).

As a result, we obtain traffic disruptions segmented by the traffic speed associated with reported accidents. This association makes it possible to perform various important tasks of the accident analysis: 1) prediction of the traffic accident impact on the traffic speed based on accident reports, 2) prediction of the traffic accident duration derived directly from the effect of disruption on the traffic speed (impact-based duration), 3) analysis of disruption propagation (each detected disruption can be studied for spatial-temporal impact within the traffic network). Through this work, we will focus on the prediction of the impact-based accident duration and lay the foundation for a further research.

Overall, the main contributions (summarised in Figure 1) of our paper are as follows:

1. We propose a fusion methodology of two large data sets (CTADS and PeMS) for a detailed traffic accident analysis. To the best of our knowledge, this is the first research study proposing the methodology for merging of these two large data sets, which allows an association between observed disruptions in traffic flow and the reported accidents.

The research of this nature (fusion of traffic flow and accident reports) has been performed before [13], [14], but our methodology has the following advantages: 1) Our disruption segmentation model can be fine-tuned via hyper-parameter search to find optimal disruption detection rate, 2) The method produces difference estimates proportional to the degree of observed disruption, which allows for control of false positives rate via threshold choice, 3) We evaluate multiple comparison metrics for traffic speed difference estimation, 4) The segmentation algorithm is more complicated and includes pre-processing convolution, test of multiple difference metrics, adjustment to selectivity and cyclic shift for difference window, 5) our methodology is modular, where each logical part can be further refined and studied in a separate research.

2. We propose a novel methodology for the disruption mining using a combination of different metrics (which we further find to have properties important for disruption segmentation): a) the Wesserstein metric, which allows us to measure the disruption severity and b) the Chebyshev metric, which provides a higher selectivity for the disruption mining and a rectangular shape of the disrupted segments, allowing an automated disruption segmentation. We detail all unique properties of both metrics utilized together to allow an accurate disruption segmentation.

3. We perform the estimation of traffic accident disruption duration from traffic speed via the above metrics which allows us to alleviate user-input errors in accident reports.

4. We evaluate multiple machine learning models by comparing both the reported and the estimated accident duration predictions extracted from traffic speed disruptions.

5. We introduce a new modelling approach which focuses on the amount and shape of the the disruption associated with an accident, which allows a further analysis and modelling of accident impact.

In contrast to one of the previous studies [14], which utilized Fuzzy Modelling, Multi-layer Perceptron, Weibull Regression, and Log-logistic Regression, our methodology that offers a higher degree of complexity. We rely on advanced machine learning models with the use of a disruption segmentation algorithm, which relies on multiple hyper-parameters. This design allows fine-tuning to find the optimal disruption detection rate.

Overall, this research forms the foundation for a new early traffic accident disruption detection, traffic disruption speed impact analysis and the use of observed traffic accident durations for correcting errors in user reports. Moreover, this work contributes to our ongoing objective to build a real-time platform for predicting traffic congestion and to evaluate the incident impact (see our previous works published in [12], [13], [14], [15], [16]).

The paper is further organised as follows: Section II discusses related works, Section III-A presents the data sources available for this study, Section III showcases the methodology, Section IV presents the disruption segmentation results, showcases the result of data set fusion, Section V presents the ablation study and Section VII provides conclusions and future perspectives.

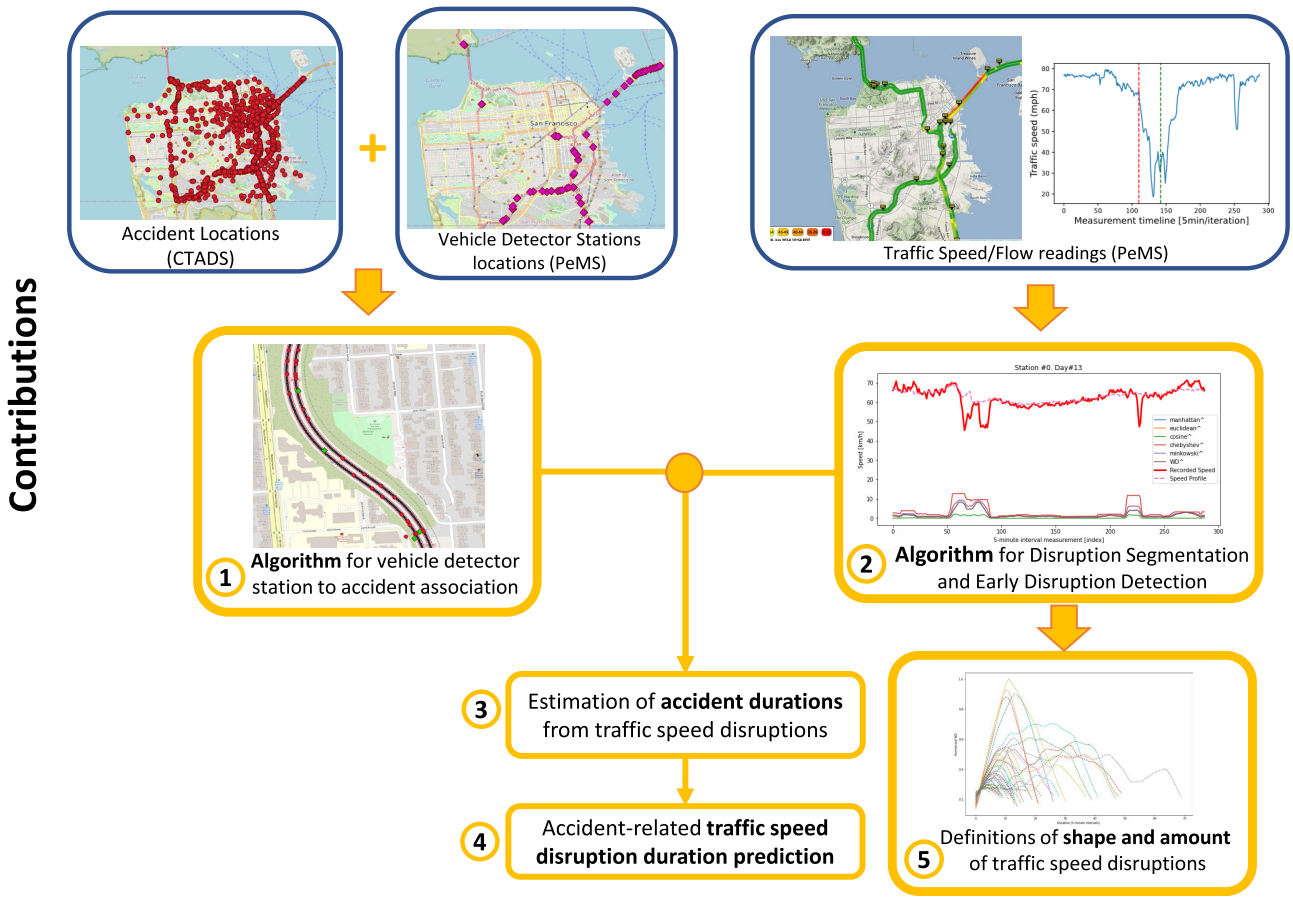

Fig. 1. Contributions and data-flow schema for association of traffic speed readings with accident reports.

II. Related Works

Multiple studies rely on user-input-based incident reports from Traffic Management Centers (TMC) with different machine learning models to predict the traffic incident duration [17]. The use of traffic flow features is found to be rare and mostly specific - incident detection and incident impact prediction by using traffic flow [18]. In other words, traffic flow data is rarely combined with actual incident reports since it requires a higher system complexity and extensive data collection.

A. Anomaly Detection Related Works

There are numerous studies related to the accident detection problem from traffic flow using anomaly detection techniques [19]. Various methods used for anomaly detection in time series are applicable for the task of traffic disruption detection. The ability to perform the detection of an actual disruption, should give us the actual shapes of disruptions and time intervals and allows an in-depth analysis of usual accident statistics, including the effect of the type of accident on the pattern of disruption in traffic flow. By integrating data on the traffic state with accident reports we are able to further connect traffic flow disruption patterns to various accident characteristics (hour of the day, weather conditions, crash type, type of vehicle involved - truck/car [20], the effect of road pavement types [21], the road design and the road operation [22], etc).

Anomaly detection in time series data is a critical problem in various applications, such as finance and transportation. The data generated by many transportation applications (e.g. vehicle trajectory or vehicle loop data acquisition) is a continuous temporal process [23]. The detection of unusual events performed in a time-critical manner, is known as streaming outlier detection. There are two main aspects of the anomaly detection from traffic speed time series: 1) continuity - traffic accident can be characterised by performing an abrupt change (which can be reformulated as a lack of continuity [24]) in traffic speed with steady or also abrupt return of traffic state back to normal condition after the accident elimination, and 2) novelty - traffic accidents can also generate unusual unobserved earlier patterns of change in the traffic speed.

There are multiple approaches for time series anomaly detection: 1) the sliding window technique, enabling continuous monitoring and timely detection of outliers, 2) Offline Outlier Detection (OOD) using predictive and statistical models, which processes data collected and analyzed later. This approach includes: a) ARIMA Models [25] for capturing temporal dependencies, b) Seasonal Hybrid ESD (S-H-ESD) [26], which combines ESD and STL techniques for high accuracy and robustness in detecting anomalies in time series with strong seasonal patterns, and c) LSTM networks [27],

a type of recurrent neural network, for capturing long-range dependencies and estimating miss-prediction costs using moving window predictions.

Offline Outlier Detection using anomaly detection models perform using the following models: a) Isolation Forest [28] which is an unsupervised learning algorithm specifically designed for anomaly detection. It works by recursively partitioning the dataset using randomly selected features and split values, constructing multiple isolation trees in the process. The rationale behind this approach is that anomalies are generally more susceptible to isolation when compared to regular data points. Consequently, the path length from the root node to an anomalous point in the isolation tree is expected to be shorter than that for a regular data point. The average path length across all trees is then used as an anomaly score, with shorter path lengths indicating a higher likelihood of being an outlier. The method was also previously used for time series anomaly detection [29], b) One-Class SVM [30] which is a variant of the Support Vector Machine (SVM) algorithm tailored for unsupervised anomaly detection. It aims to find the smallest hyperplane that separates normal data points from the origin in the feature space, thereby constructing a boundary around the normal data. This is achieved by solving a quadratic optimization problem that maximizes the margin between the data and the origin. Any data point that falls outside the boundary is considered an anomaly.

Streaming outlier detection is important for timely detection of unusual events, such as traffic accidents. Continuity and novelty are the two main aspects of anomaly detection in traffic speed time series. There are multiple approaches to perform anomaly detection from time series, including the sliding window technique, offline outlier detection using predictive and statistical models, and offline outlier detection using anomaly detection models.

B. Incident Duration Prediction Works Using Machine Learning

Various machine learning models are used to solve the task of traffic accident duration prediction [17] including k-nearest neighbours (KNN) and Bayesian networks [31], Recursive Boltsman Machines and Support Vector Machines(SVM) [32] and Random Forests (RF) [33].

XGBoost [34] is a popular gradient boosting framework known for its strong performance on diverse tasks. LightGBM [35] is another gradient boosting framework that focuses on efficiency and scalability, handling large datasets and supporting parallel and GPU learning. CatBoost [36] is a gradient boosting framework designed for handling categorical features effectively, employing an efficient ordered boosting implementation. GBDT (Gradient Boosted Decision Trees) [37] is a technique using an ensemble of decision trees built sequentially, focusing on residual errors of previous trees, resulting in a powerful predictive model. KNN, in the context of traffic accident duration prediction, works by identifying the k most similar historical accidents based on their features and calculating the average duration of these accidents to predict the duration of a new accident. Bayesian network is a probabilistic graphical model that represents a set of variables and their conditional dependencies via a directed acyclic graph. For the traffic accident duration prediction, Bayesian networks model the relationships between various factors such as weather, road conditions, and accident severity, allowing for an estimation of the duration based on the joint probability distribution of these factors. In the context of traffic accident duration prediction, SVM works by mapping the input features into a high-dimensional space and constructing an optimal separating hyperplane to distinguish between different accident duration classes. Random Forest is an ensemble learning method that constructs multiple decision trees and combines their predictions by averaging (regression) or by majority voting (classification). In the traffic accident duration prediction, RF models build decision trees based on various factors such as weather conditions, road type, and accident severity to predict the duration of an accident.

In one of the studies [33], the Random Forest model achieved a mean absolute error (MAE) of 36.652 minutes for the prediction of incidents with duration ranging from 1 to 1,440 minutes; also Random Forest model showed much more stable results than Artificial Neural Network, with only a small error range. The RF built-in variable-importance capability allows to identify the most important variables impacting prediction performance.

In recent study [38], extreme gradient boosting machine algorithm (XGBoost) was used to predict incident clearance time on freeway and analyze the significant factors of clearance time. The XGBoost was used to model the nonlinear data in high-dimensional space and quantify the relative importance of the explanatory variables.

It is also possible to use multiple identified clusters of traffic accidents to train an ensemble of machine learning models, each optimized to perform with a separate cluster with averaging ensemble prediction results [39].

For our scenarios, we can rely on Random Forest or XGBoost to estimate the importance of each model's parameters on the task of predicting the model performance. To perform this prediction, variation of model parameters with estimation of performance is necessary. It will allow to establish a comparative parameter study.

Also, a hybrid deep learning model based on multi-source incomplete data to predict the duration of countrywide traffic incidents in the U.S was also recently proposed [40]. The text data from the natural language description in the model were parsed by the latent Dirichlet allocation (LDA) topic model and input into the bidirectional long short-term memory (Bi-LSTM) and long short-term memory (LSTM) hybrid network together with sensor data for training. We also performed a similar study by incorporating textual accident description and traffic speed/flow encoded using LSTM into incident duration prediction models performed using the same data set (Countrywise Traffic Accident Dataset) [41] Both studies demonstrate an improvement in model performance. Nevertheless, data incorporation is not the focus of our study. After thoroughly studying this data set, we observed a significant amount of misreported incident duration values (up to 40% of all reports just for San-Francisco area), therefore we try to propose a

framework to estimate to real accident duration as observed from the impact on traffic speed.

The definition of traffic incident duration phases is provided in the Highway Capacity Manual [42] and includes the following time-intervals: 1) incident detection - the time interval between the incident occurrence and its reporting, 2) incident response - time between the incident reporting and the arrival of the response team, 3) incident clearance time between the arrival of the response team and the clearance of the incident, 4) incident recovery - the time between the clearance of the incident and the return of traffic state to normal conditions. In this research, we rely on total incident duration - the time between incident occurrence and return of the state to normal conditions. Also, we analyse the subset of traffic incidents - traffic accidents. As we found during the data investigation, traffic accident duration is reported at the time when the incident is cleared by the response team, which doesn't include the duration of the effect that the accident produces on traffic flow. Traffic incident duration prediction studies rely on incident reports without emphasizing on the duration of observed incident effects. In this research we try to solve this issue by proposing the methodology for disruption segmentation from traffic speed.

C. Data Sets for Incident Duration Prediction

Analysis of the effect of traffic incidents has been performed previously using Caltrans PeMS data, where the measure of incident impact was represented as a cumulative travel time delay [43], which is an aggregated value. However, traffic state recovery from disruptions is not necessarily following a single pattern - it may be slowly dissipating, we may observe secondary crashes, it may have a high or low impact, etc. Traffic accident duration prediction methodology relies on reported traffic accidents, but actual reports may contain user-input errors and be misaligned with the actual shape of disruption produced by the accident. Therefore, the approach for disruption segmentation may provide the accident duration estimated from the actual shape of disruption in traffic flow.

III. Methodology

The new framework we propose in this paper is represented in Figure 1 which we support across some initial definitions for our modelling approach (see next sub-section). First, we associate the road segments with their corresponding Vehicle Detector Stations (VDS) from the Caltrans PeMS data set, as well with the locations of reported accidents (see Algorithms 1 and 2 proposed in sub-section III-D). The main outcome of this algorithm is that traffic accidents will get associated with the traffic flow, speed and occupancy readings from the VDS stations.

Second, we propose a new algorithm for early disruption detection and segmentation, detailed in sub-section III-E. By detecting disruptions that occurred in time-space proximity of reported traffic accidents, we obtain the estimated traffic accident duration. This gives us much more information to include in the model training than just the simple accident duration: 1) the disruption shape in terms of modifications of speed data profiles from the standard patterns 2) the accident duration estimated from the impact on the traffic speed 3) the cumulative accident impact estimation.

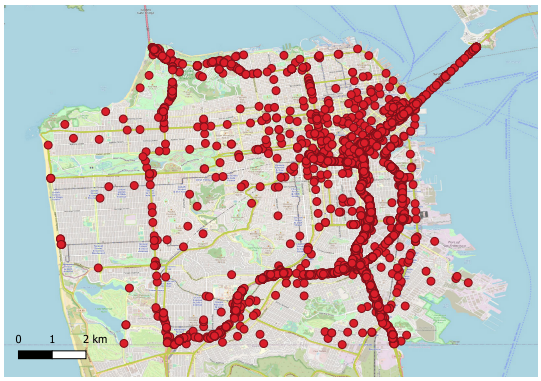

Fig. 2. CTADS reported accidents for San-Francisco.

A. Case Study

Before diving into the methodology, we provide a brief introduction into the data sets in use for showcasing our approach, which helps establishing the modelling base and understanding of the steps taken. We make the observation that the current methodology can be applied on any incident and traffic state data set which can contain a time component, and is not bounded to the chosen data sets for exemplification.

1) CTADS: Accident Reports Data Set: We rely on accident reports from the "Countrywide Traffic Accident Dataset" (CTADS), recently released in 2021 [44], [45], which contains 1.5 million accident reports collected for almost 4.5 years since March 2016, each report containing 49 features obtained from MapQuest and Bing services. We select the area of San-Francisco, U.S.A and extract data for 9,275 accidents (see Figure 2).

The Countrywise Traffic Accident Data Set (CTADS) offers an insight into the extent of recorded accidents. Particularly the Bing data subset has start and end locations of accident extents, CTADS includes accident extents calculated using Havesine distance formula. Properties of extents may help to fine-tune our algorithm. We further rely on the Bing data subset to derive our conclusions regarding the accident extent properties. Figure 3 represents the distribution of these accidents' extents in a histogram, showing how often certain disruption extents occurred within the data. Additionally, Figure 4 depicts the empirical cumulative distribution function (ECDF) for the same data. The ECDF provides a complementary perspective to the histogram, showing the proportion of data points that fall below each value on the x-axis (e.g. around 90% of reported accidents have the road extent below 500m). A summary of important statistics, derived from the data, is provided in Table I. This table includes key measures including the interquartile range (0.25 to 0.75 quantiles). We can safely choose 500m as the maximum accident extent for our association algorithm between vehicle detectors and accident points.

2) PeMS: Traffic Speed and Flow Data Set: We rely on Caltrans Performance Measurement System (PeMS) [46] to collect data on traffic flow and speed. This data set provides

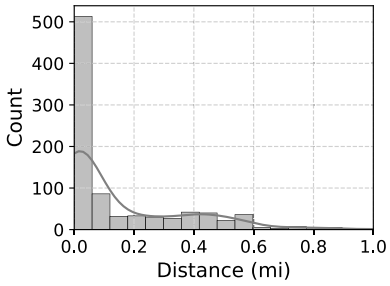

Fig. 3. CTADS: Bing - Histogram for recorded accident extent.

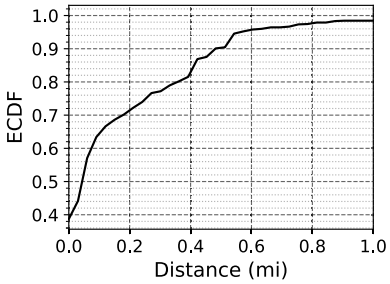

Fig. 4. CTADS: Bing - ECDF for recorded accident extent.

TABLE I
STATISTICS ON ACCIDENT EXTENT FROM CTADS (BING PART) DATA SET

Statistic	Value $[km]$
Mean	0.16
Median	0.04
0.95 Quantile	0.57
0.05 Quantile	0
Standard Deviation	0.27
Variance	0.08
Interquartile Range $[0.25, 0.75]$	0.27

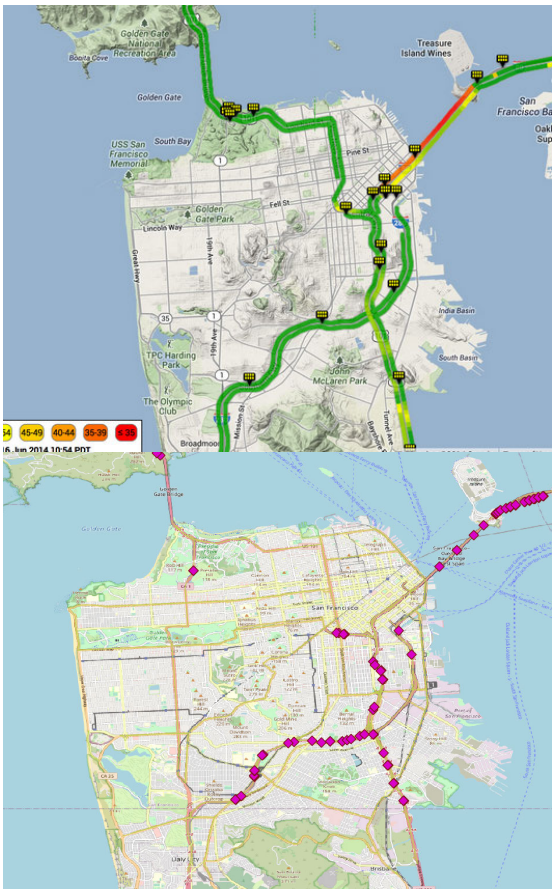

Fig. 5. 1) PeMS data set area coverage for San-Francisco (the map is available at https://pems.dot.ca.gov/) 2) Mapping of the Vehicle Detection Stations from PeMS data set. OpenStreetMap excerpt showing San Francisco. Available at:https://www.openstreetmap.org/#map=12/37.7612/-122.4395.

aggregated 5-minute measurements of traffic flow, speed and occupancy across California. We decided to extract the data for the area of San-Francisco (see Figure 5a), which contains 83 Vehicle Detection Stations (VDS) placed in that area (see 5b), and we try to associate each traffic accident occurred with each of San-Francisco VDS in their 500m proximity using the algorithm detailed in the following section. In total, from 9,275 accidents in the area (extracted from CTADS) we have obtained 1,932 traffic incident reports which we were able to associate with the correct and complete traffic flow and speed readings from a VDS.

B. Speed Difference Estimation Definitions

In the current study we compare the performance of multiple difference metrics that will help us to correctly estimate the impact of an accident and the deviation from the historical speed patterns. These metrics are defined as follows:

a) The Chebyshev difference is a measure of the maximum difference between corresponding elements of two one-dimensional vectors u and v and is expressed as:

$$D_{\text{Cheb}}(u, v) := \int \max_i(|u_i - v_i|) \tag{1}$$

This distance metric is commonly used in data analysis and is named after Pafnuty Chebyshev, who introduced the concept in 1853 [47].

b) The Wasserstein difference, also known as the earth mover's distance, is a measure of the minimum "work" required to transform one probability distribution u into another v. It is expressed as:

$$D_{\text{WD}}(u, v) = \inf_{\pi \in \Gamma(u,v)} \int_{\mathbb{R}\times\mathbb{R}} |x - y| \mathrm{d}\pi(x, y) \tag{2}$$

This metric was introduced by Leonid Kantorovich in 1942 [48] and has found applications in fields such as computer vision, image processing, and natural language processing.

c) The cosine difference, also known as the cosine similarity, is a measure of the similarity between two one-dimensional vectors u and v. It is expressed as:

$$D_{\text{C}}(u, v) = \frac{u \cdot v}{|u|_2 |v|_2}. \tag{3}$$

This metric is commonly used in information retrieval and has also found applications in recommender systems and document clustering [49].

d) The Euclidean difference is a measure of the distance between two one-dimensional arrays u and v in a Euclidean space. It is expressed as:

$$D_{\mathrm{E}}(u, v) = \left(\sum (w_i |(u_i - v_i)|^2)\right)^{1/2} \tag{4}$$

This metric is commonly used in fields such as machine learning, computer vision, and signal processing.

e) The Minkowski difference is a generalization of the Euclidean difference and is a measure of the distance between two one-dimensional arrays u and v in a Minkowski space. It is expressed as:

$$D_{\mathrm{M}}(u, v) = (\sum |u_i - v_i|^p)^{1/p} . \left(\sum w_i (|(u_i - v_i)|^p)\right)^{1/p} . \tag{5}$$

This metric is a generalization of other distance metrics, such as the Manhattan distance (when $p = 1$) and the Euclidean distance (when $p = 2$), and is commonly used in fields such as physics, engineering, and data science [50].

f) The Bray-Curtis difference metric [51] between two vectors $\mathbf{u}$ and $\mathbf{v}$ is given by:

$$D_{\mathrm{BC}}(\mathbf{u}, \mathbf{v}) = \frac{\sum_{i=1}^{n} |u_i - v_i|}{\sum_{i=1}^{n} (u_i + v_i)}, \tag{6}$$

where n is the number of dimensions in the vectors.

g) The Canberra difference metric [52] between two vectors $\mathbf{u}$ and $\mathbf{v}$ is given by:

$$D_{\mathrm{Can}}(\mathbf{u}, \mathbf{v}) = \sum_{i=1}^{n} \frac{|u_i - v_i|}{|u_i| + |v_i|}, \tag{7}$$

where n is the number of dimensions in the vectors.

C. Accident Duration Prediction Task Definitions

Using all available data sets and the incident information, we first denote the matrix of traffic incident features as:

$$X = [x_{ij}]_{i=1..N_i}^{j=1..N_f} \tag{8}$$

where N_i is the total number of traffic incident records used in our modelling and N_f is the total number of features characterising the incident (accident severity, vehicles involved, number of lanes, etc) according to the accident report data set.

Traffic Speed represented as a vector with 5-minute averaged readings from Vehicle Detector Stations:

$$S = [s_i]_{i=1..N} \tag{9}$$

where N is the total amount of traffic speed readings.

Within this research we assess the performance of Machine Learning models on tasks of predicting reported and estimated accident duration. We define the task of accident duration prediction as a regression problem.

The incident duration regression vector (Y_r) is represented as:

$$Y_r = [y_i^r]_{i \in 1..N}, \; y_i^r \in \mathbb{N} \tag{10}$$

and the regression task is to predict the traffic accident duration y_i^r based on the traffic incident features $x_{i,j}$. The regression models go via an 10-fold cross-validation procedure with hyper-parameter tuning.

The estimate the accident duration prediction performance we use the root mean squared error (RMSE):

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (A_t - F_t)^2} \tag{11}$$

where A_t - actual value, F_t - predicted value.

D. Algorithm for Vehicle Detector Station to Accident Association

In order to match correctly what traffic conditions reflect best the effects of each incident, we further define the association procedure between traffic accidents and VDS (Accident-to-VDS), for the San Francisco area. We observe that only a few traffic accidents have VDS stations in their proximity to allow a good traffic speed and flow extraction, as shown in Figures 2 and 5.

In order to find the traffic incidents for which we can have traffic flow and speed data, we develop a mapping algorithm (Accident-to-VDS) which consists of two parts (see Algorithm 1-2), defined by the following steps:

1) We extract primary and secondary road lines from Open Street Map.
2) Road segments are then transformed into points at 2-meters equal distance.
3) Each VDS station and accident are mapped to the closest road point (up to 10m distance).
4) From this step we use the following algorithm to process the point-based representation of VDS, accidents and road segments (see Algorithm 1). The vdsPoints array contains tuple of form (VDS ID, x and y coordinates), each point in accidentPoints contains an array visitedBy (initialized to be empty) to maintain a list of stations in proximity of the accident and assignedVDS as a resulting nearest VDS station to the accident along the road.

The algorithm relies on a recursive function to implement the process of visiting road points (see Algorithm 2). The association part of the algorithm works as follows:

1) We select the current VDS station.
2) We move (jump by points) in all possible directions available from the starting and forthcoming points in a 3m radius. This radius allows us to move along the road jumping between road points. Movement in all possible directions allows to grasp the propagation of the traffic congestion associated with the accident. The maximum available distance is set to 500m (250 jumps) and allows to limit the observable impact distance.

Algorithm 1 Accident-to-VDS: Accident to VDS Mapping Algorithm

```
Input: point
Output: None
Access global arrays:
 roadPoints, accidentPoints, vdsPoints
Function
 visitNearestPoints(VDSID, point, currentHops)
accidents :=
 findNearestAccidents(point, accidentPoints, 10m)
for i = 0 to length(accidents) do
    a := accidents[i]
    a.visitedBy.append([VDSID, currenthops]) ;
     //Recording visits from stations
     to internal accident list
end
if currenthops < 500/2 then
    ;     //Limiting the travel distance
     from VDS
    roadpoints:=findNearestRoadPoints(point,
     roadPoints, 3m) for i = 0 to length(roadpoints)
     do
        rp := roadpoints[i]
        if VDSID not in rp.visitedBy then
            ;     //Preventing the infinite
             recursion
            rp.visitedBy.append(VDSID)
             visitNearestPoints(point, currentHops+
             1)
        end
    end
else
    Return
end
```

3) By moving across points we collect traffic incidents in the 5m proximity of each point and associate them with the current VDS station.

The algorithm is recursive and relies on the list of visited points for each VDS. At the end of the algorithm, we have a subset of traffic accidents with their associated VDS which allows us to extract the traffic flow and speed in the vicinity of the accident. Ideally, all traffic accidents should have associated traffic flow but given their unavailability (due to detector coverage), we select accident reports which have associated traffic flow information currently available from the PeMS data set.

E. Algorithm for Automated Disruption Segmentation (ADS)

Once the accidents have been mapped and associated to their VDS stations which allows us to select the flow/speed that match the day of the incident, etc, we are using the extracted traffic state parameters to propose a new automated disruption segmentation (ADS) method. The algorithm for the segmentation of disruptions via traffic speed works as follows:

Algorithm 2 The Recursive Function for Traveling Across Road Points

```
Input: roadPoints, accidentPoints, vdsPoints
Output: assignedAccidents
for i := 0 to length(vdsPoints) do
    vds: = vdsPoints[i]
    visitNearestPoints(vds, 0)
end
assignedAccidents = []
for i := 0 to length(accidentPoints) do
    accident := accidentPoints[i]
    if length(accident.visitedBy) > 0 then
        accident.assignedVDS =
         sort(accidetn.visitedBy, sortvalue =
         hops)[0] ;    //Choosing closest VDS
         station
        assignedAccidents.append(accident)
    end
end
return assignedAccdents
```

1) A time series pre-processing step prepares all the data for segmentation (see Alg. 3):
   a) Calculate the average monthly profile for daily traffic speed measurements;
   b) Iterate over the traffic speed time series using a moving window of 1-hour time interval (in total there are twelve measurements of 5-minute each)
   c) On each iteration perform a comparison of a 12-unit window between the monthly profile and the current day of measurements. The resulting single value is added to the resulting time series sequence.
   d) Calculated the time series differences (TS) choosing the above defined metrics will be then adjusted by selectivity (using the power function, which will keep values closer to one for the least affected by the function and minor values the most suppressed) and normalized to produce nTS and pTS arrays respectively.
2) The time-series segmentation step (see Algorithm 4):
   a) A first order derivative (dTS) is calculated for the resulting time series of the previous stage (nTS), which returns positive peaks when entering the disruption and negative peaks when exiting the disruption state.
   b) We iteration over resulting derivative time series to record the opening and closing of each disruption in each time series. If two consecutive positive peaks (opening times) are observed then we choose the largest one between the two (we will further debate on this aspect in our future work plans). We repeat the same for consecutive negative peaks.
   c) We then associate the detected disruptions with the accident reports: for each accident report, we extract the traffic speed time series on the day of the accident and if both opening and closing

Fig. 6. The application of dilation operation to an image and time series.

times are recorded, we perform an association of the accident with these times and extract the actual time series sequence for further analysis.

Enhancing selectivity: We use the convolution with the kernel (1,1,1), which attributes to the morphological dilation operation, to facilitate the work of the segmentation algorithm. By applying this convolution we make multiple consequent differences to be accumulated; for example, assuming we have a sequence of 0.3, 0.1, 0.1, 0.2 and 0.2 as differences for each 5-minute step, therefore a total of 0.9 change over 4 iterations. The convolution (1, 1, 1) will produce the values of 0.5, 0.4, and 0.5 by making a sequence of high values from the sequence of small changes (see Figure 6). The dilation operation is primarily used in computer vision tasks to make connected groups from closely placed scattered points to facilitate a further image analysis.

To obtain the monthly profile, the traffic speed measurement sequence was obtained for a duration of 1 month from the VDS before the accident occurred, and was done separately for each accident. This sequence then gets reshaped into a matrix of the form $[number\_of\_days; 288]$, where columns contain the total number of measurements across an entire day (24*12=288). The monthly average was then calculated across axis 1 (number of days) to obtain a vector with 288 values of measurements. This vector gets recalculated for a number of days of observations from each detector to be comparable with the VDS daily measurements.

As an observation, the constants *pThreshold* and *nThreshold* represent thresholds for change that observed in the time series of the metric derivative; they allow us to define a positive and negative change of the difference metric, the selectivity defines power function coefficient to suppress the non-significant and filter the most significant disruptions.

F. Modification of the Algorithm for Automated Real-Time Early Disruption Detection

Since our proposed algorithm doesn't look into the future and calculates different metrics based on the currently observed traffic speed and a few measurements in the past (11 units in the current study, equivalent to 55 minutes from the past), we can perform an early accident detection which will consist in calculating and comparing the first-order differential (FOD) of Chebyshev metric based on the monthly profile. The detection of significant positive peaks (e.g. 0.3-0.5 of normalized difference metric) can identify the amount of disruption in real-time. The end of the disruption can be detected using the same approach in real-time as well by observing a significant negative peak.

Algorithm 3 Algorithm for Automated Disruption Segmentation. Part 1

Input: $monthlyProfile, speedReadings, selectivity, shift$
Output: cTS
; //Accidents array contains a day number, starting and ending index for segmented traffic disruptions
$step := 1$
$windowsize := 12$
$i := windowsize$
$lastDiff = 0$
$DS = []$
while $i < length(speedReadings)$ **do**
  $A := speedReadings[i - windowsize : i]$;
  //Look-back window of readings
  $B := monthlyProfile[i - windowsize : i]$
  $diff := metric(A, B)$
  $DS.append(diff)$
  $lastDiff = diff$
end
for $i = 0$ *to windowsize* **do**
  ; //Padding array with the latest observed value to obtain full-day readings
  $DS.append(lastDiff)$
end
$pTS = power(TS, selectivity)$; //The use of power function to improve selectivity of significant disruptions
$nTS = cyclic_shift(shift)$; //The use of cyclic shift operation
$nTS = normalize(pTS)$
$dTS = derivative(nTS)$; //First order derivative allows to decompose metric results into positive and negative change to the disruption amount
$cTS = convolution(dTS, [1, 1, 1])$
return cTS

IV. Results

A. Data Exploration and Setup

CTADS data set contains traffic accident reports, which after an initial data mining investigation, we found to contain several user-input errors; for example, a lot of traffic accident durations have been rounded to 30 or 360 minutes (see Fig. 7d)); or the incident start time which was reported is unrelated to any disruptions observed by the vehicle detector stations in the proximity - see Figure 7 in which we have provided two different examples of speed recorded during two different accidents A-5198 and A-4490; the red lines indicate the official reported start and end time of the accidents, while in reality the accidents have had a long lag in spreading across the network - see Fig. 7a) or were reported much later that the official speed drop was recorded - see Fig. 7b).

At this step we observed a significant amount of user-input errors in accident reports, which affect the accident

Algorithm 4 Algorithm for Automated Disruption Segmentation. Part 2

Input: $cTS, pThreshold, nThreshold, selectivity$
Output: $Accidents$
; //Accidents array contains a day number, starting and ending index for segmented traffic disruptions
$state := 0$
$Accidents = []$
for $i := 0$ **to** $length(cTS$ **do**
  if $cTS[i] > pThreshold$ **then**
    ; //Significant positive peak identifies the start of disruption
    if $state <> +1$ **then**
      $state = +1$
      $enteridx = i$
    else
      if $cTS[i] > cTS[enteridx]$ **then**
        $enteridx = i;$
        ; //Choosing the largest change from previously observed
    end
  end
  if $cTS[i] < nThreshold$ **then**
    ; //Significant negative peak identifies the end of disruption
    if $state <> -1$ **then**
      $state = -1$ $exitidx = i$
    else
      if $cTS[i] < cTS[enteridx]$ **then**
        $exitidx = i;$
    end
  end
  if $i \ mod \ 288 == 0 \ and \ i > 0$ **then**
    ; //Reset segmentation procedure at the end of each day
    $state = 0$
    $Accident.append([i \ div \ 288, enteridx, exitidx])$
  end
end
return $Accidents$

duration/impact analysis: 1) accidents can be reported earlier or later than its occurrence (observable disruption misalignment in time) 2) a report can be filled with "placeholder" duration values not representing the actual accident duration 3) there may be no observable disruption in traffic speed despite the accident report (due to placement and management of the accident) (false positive) 4) there may be accident-related traffic disruptions not grasped by accident reports (false negative). Therefore, incorrect accident start time, duration and end time, unreported presence or abscence of disruption make it necessary to estimate accident duration characteristics from traffic state data instead of relying on user reports. In this paper our proposed methodology is really meant to solve the user-reporting issues related to traffic accidents and to be applied automatically on any data set, regardless of its nature or geo-location.

The use of PeMS data set allows to estimate the impact of accidents on the traffic states (flow, speed). For our scenarios, we choose the area of San-Francisco with accidents recorded from 2016 to 2020 in the CTADS data set. We then obtain Vehicle Detector Station locations from PeMS, the road network shape from OpenStreetMap and we perform an association of CTADS accident reports with VDS stations along the road within 500m proximity. We then try to segment the disruption time interval occurred on the day of an accident. Further, we associate observed disruptions in the traffic speed series with actual accident reports. The purpose of this step is to reduce user-input errors in accident reports and to enhance the modelling of traffic disruptions with an analysis of traffic speed.

B. Metric Performance Comparison

We apply the difference metrics detailed earlier in Section III to a monthly traffic speed/flow profile (monthly readings averaged to one day) and reading on the day of the traffic accident. There are two approaches to applying the difference calculation: 1) a global difference - when we try to find the difference between the monthly profile and traffic flow/speed readings on the day of the accident; the global approach is too broad and will not allow the actual comparison between disruptions localized in time (metric results can be very similar between the very long subtle disruption and abrupt but impactful one). We measure the amount of difference that occurred within a moving time window (we choose twelve 5-minute time intervals equivalent to one hour). Traffic speed/flow readings from the moving window are taken right before the currently observed value to ensure that the difference estimation algorithm is not looking into the future.

To compare the metric performances we provide an example of speed readings from one of the detector stations. Each difference metric demonstrates its specifics as represented in Figure 8: 1) the Chebyshev metric, which we define as the maximum difference between the monthly profile and the observed readings, produces a noticeably rectangular shape and demonstrates a higher selectivity towards major disruptions than other metrics; the Chebyshev metric will be further used for the automated accident segmentation; 2) the use of Cosine metric allows to detect the change in the traffic state - speed decrease and increase both represented as positive peak values, 3) the Wasserstein difference allows for smooth representation of the amount of disruption (conceptually, it measures the amount of work necessary to change one shape into another, which we can rephrase as the amount of work produced by an accident to deviate the traffic state from the normal operation), 4) the Minkowsky, Euclidean and Manhattan difference metrics show little to no difference to the Wesserstein distance; we choose to use the Wasserstein difference since its connection to physical interpretation.

Examples of applying our proposed algorithm are presented on Figure 9. The 'Disruption Start' and 'Disruption End', which are represented as dashed blue and red vertical lines

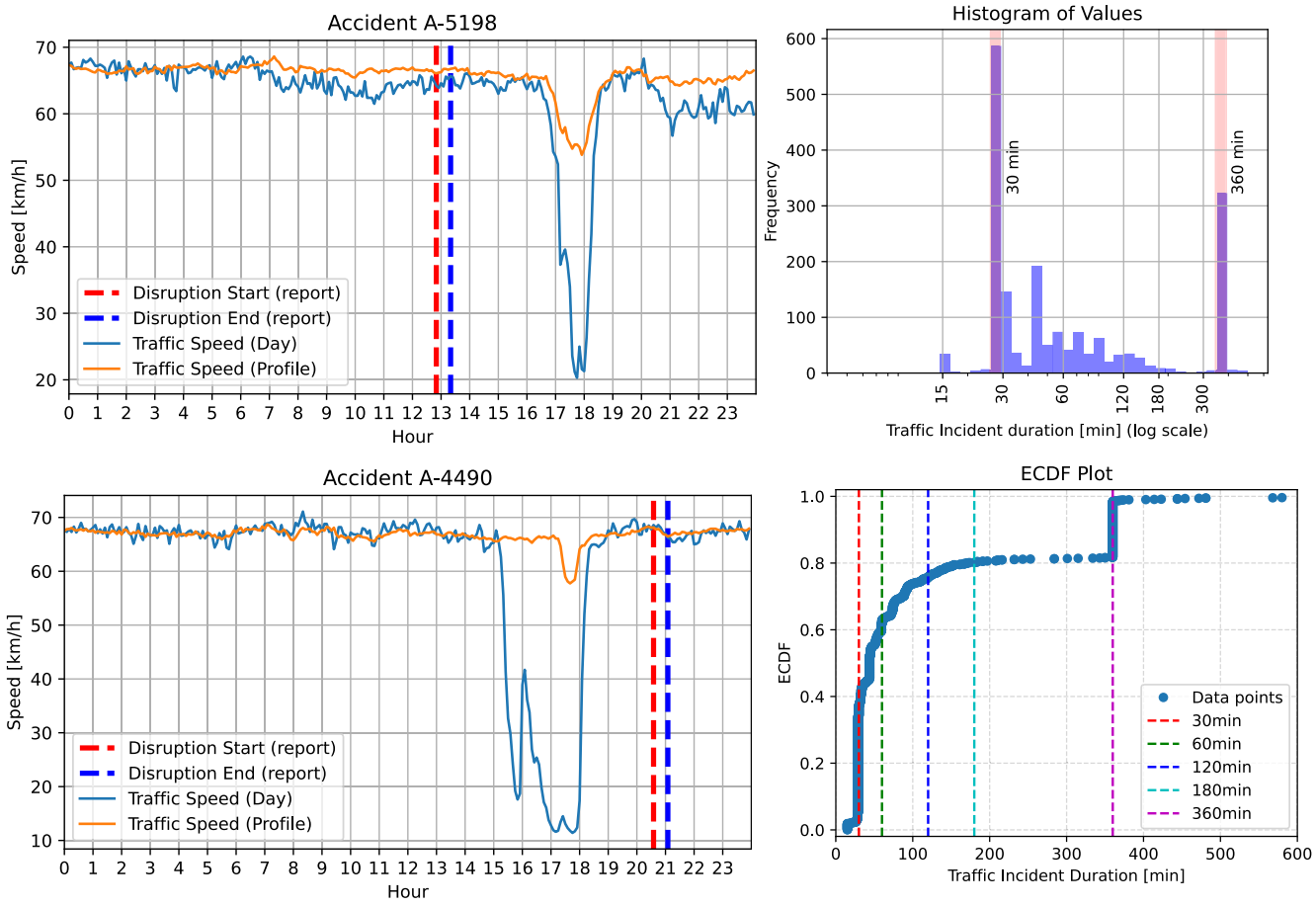

Fig. 7. User input errors located within the CTADS data set.

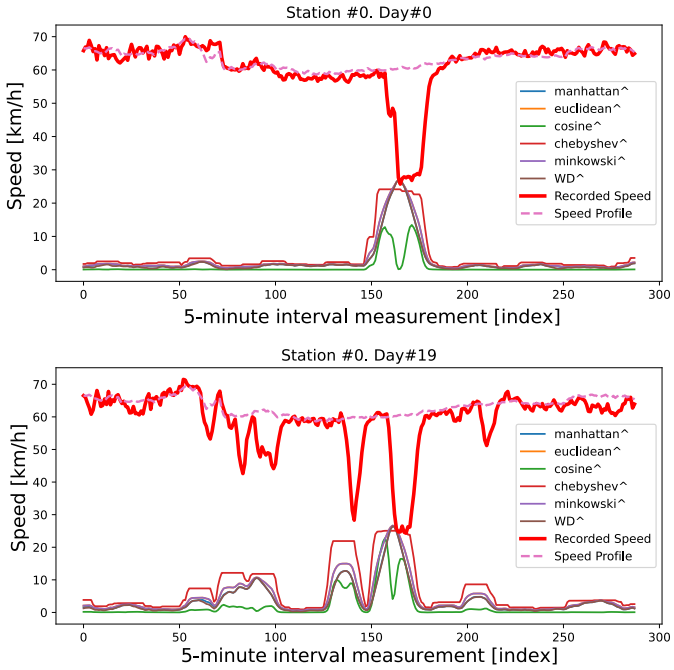

Fig. 8. Various metrics applied to difference between recorded speed and speed profile.

correspondingly show a reported accident timeline. The 'Day' (blue line) represents the traffic speed on the day of the incident and 'Profile' shows the average speed for every 5-minute interval across 14 days of measurements. Application of the 'Wasserstein distance (WD)' shows a gradual measurement of the observed disruption, while the 'Chebyshev' metric shows the nearly rectangular outline of a time interval where disruption is observed. This 'rectangular' result of the 'Chebyshev' metric was the main consideration for the development of the presented algorithm. One of the main observations from the figures as well as from the procedure of manual markup was that accidents were primarily reported 1-2 hours after the return of traffic state to normal conditions (which we define as the end of disruption). Other observation is that accident timeline is often misreported as a 'rounded' value of either 30 or 360 minutes. Application of both metrics shows a clear outline of disruption shape observed in traffic speed.

C. Combination of Our Proposed Methodology With Modern Methods for Accident Scene Segmentation

Image segmentation methods can be utilized to output a degree to which an accident is observed in an image [53], ultimately helping to create an accident timeline. By leveraging the power of semantic segmentation, the method can quantify the extent of the accident by assigning scores or probabilities to different elements within the scene. Here's how this can be done: 1) Accident-related object detection (Spatial analysis): Semantic segmentation can identify accident-related objects in the scene, such as damaged vehicles, debris, or injured

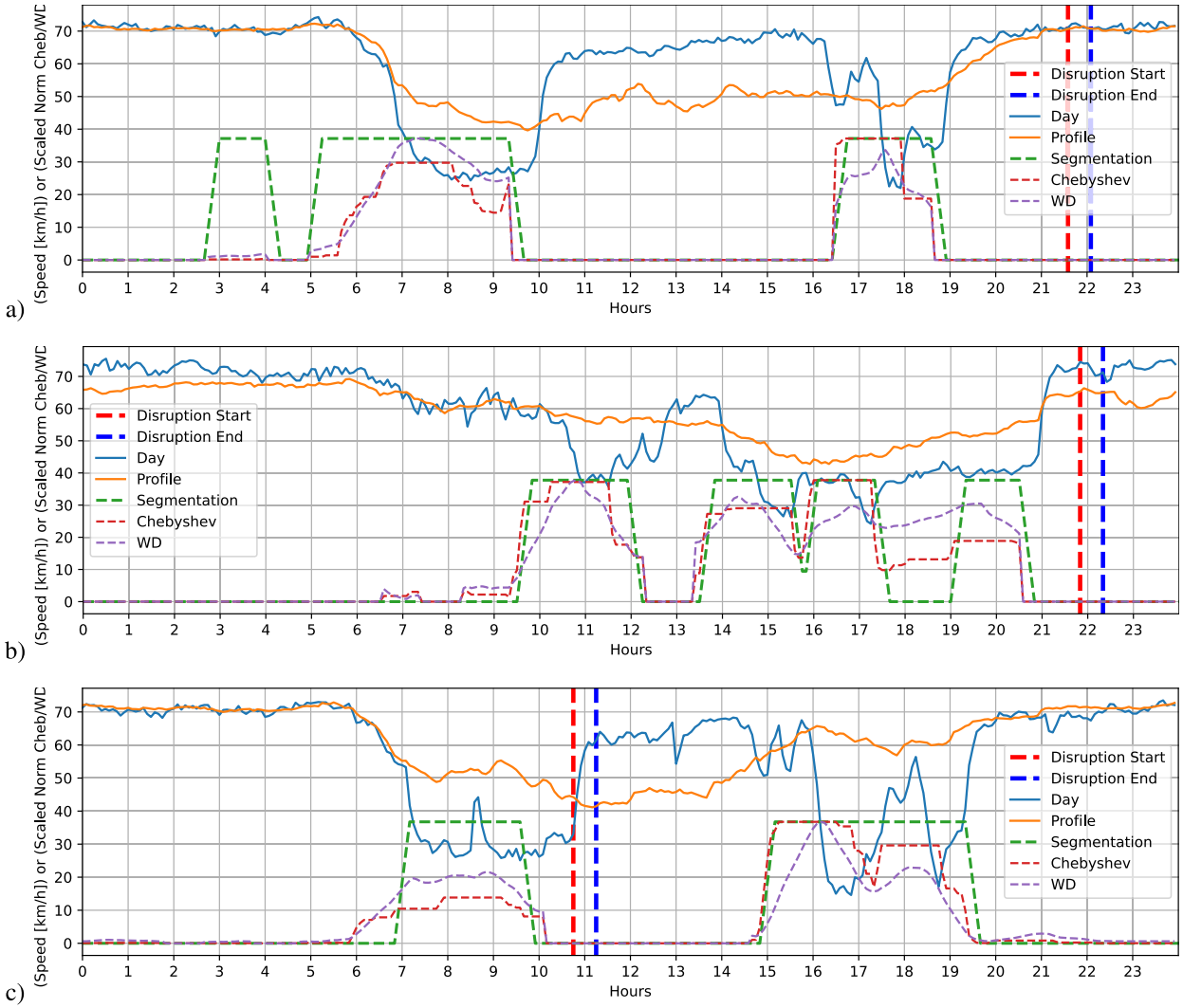

Fig. 9. Results disruption segmentation algorithm application for accidents a) A-5764, b) A-8119, c) A-9931.

pedestrians; by calculating the proportion of these objects within the segmented image, it is possible to assign a degree or score that represents the severity or extent of the accident at that specific moment, 2) Temporal analysis: by analyzing the segmented images over time, we can track changes in the accident scene, such as the motion of vehicles or the appearance of new accident-related elements. This enables the creation of a timeline that reflects the progression of the accident and the associated changes in the severity or extent of the event, 3) Probability-based analysis: advanced segmentation methods can output probability maps that indicate the likelihood of each pixel belonging to a specific class or label; by analyzing these probability maps, it is possible to compute a score that represents the degree of accident occurrence within the scene over the timeline, 4) Accident phase classification: The degree to which an accident is observed can also be used to classify distinct phases of the accident, such as pre-collision, impact, and post-collision. By evaluating the changes in accident-related object proportions or scores over time, the segmentation method can identify critical moments or transitions between different accident phases. This information can be used to construct a detailed accident timeline that highlights the key events and their corresponding degrees of severity.

There is potential to connect our proposed methodology with accident scene segmentation research approaches to create a more comprehensive and accurate framework for analyzing traffic accidents and predicting disruption durations. Here's how the two research approaches can be integrated: 1) Improved incident duration prediction: The segmentation output from the first research can be used as input for the early detection and disruption segmentation algorithm in the second research. This would allow for a more accurate identification of critical moments in the accident timeline and better prediction of incident durations, 2) Integration of mathematical metrics: the Wasserstein and Chebyshev metrics proposed in the second research can be used to refine the segmentation results obtained from the accident scene segmentation over timeline. This would help to improve the performance of the accident scene segmentation and contribute to a more accurate incident duration prediction, 3) Joint machine learning model: The speed disruption segmentation from our research can be

combined with the semantic segmentation methods to create a joint machine learning model. This integrated model could leverage both the event-driven dynamic context and the mathematical metrics for segmentation to improve its predictions for incident duration and severity. By connecting these two research approaches, a more comprehensive framework for analyzing traffic accidents and predicting disruption durations can be developed. This integrated approach would benefit from the strengths of both methods, enabling more accurate and reliable predictions for incident durations. Ultimately, this could lead to improvements in road safety, emergency response, and traffic management.

In conclusion, image segmentation methods can be employed to not only segment the accident scene but also to quantify the degree to which an accident is observed in an image. This information can be used to create an accident timeline that reflects the progression of the accident, the severity of the event, and the critical moments when interventions or safety measures could have been taken. This approach in combination with our proposed disruption segmentation method can potentially contribute to better accident analysis, road safety improvements, and more effective emergency response strategies.

D. Automated Disruption Segmentation Results

Figure 9 presents the results obtained from our algorithm for the automated disruption segmentation. The segmentation line (dotted blue) represents the estimated disruption intervals represented as 0 and 1 to perform our visualisation investigation better. Figure 9a) shows that there may be multiple observed disruptions in a 300*5=1500 time interval. Due to errors in accident reports regarding the starting time and the duration of the accident, it is non-trivial to determine which disruption is associated with the accident. The situation may be easier in the case when only one disruption is observed during the day. According to our algorithm, we select the largest disruption on the day the accident was reported. Figures 9b) and 9c) highlight additional specific situations which need to be considered: 1) higher traffic speed at the end of the day than observed from the monthly profile, 2) unstable traffic speed approaching normal traffic conditions with high frequency, 3) slight misalignment of disruption intervals with the visually observed disruption intervals. All these problems can be addressed by using manual segmentation with deployment of Deep Learning models since there are advanced computer vision methods proposed in recent years (e.g. autoencoders for segmentation).

E. Comparison of Estimated, Reported and Manual Markup of Accident Durations

There is a significant difference between the estimated and the reported accident durations that we would like to highlight: 1) the reported accident durations contain a large amount of 30 and 360 minutes duration values (nearly 40% of data - see Figure 10a)) while the estimated accident durations using our approach have an average duration of 58 minutes, while the reported is 108 minutes (which is by assumption skewed due to 360 placeholder values), 3) the estimated accident durations are distributed between 90 and 355 minutes (0.10 and 0.90 quantiles correspondingly) (see 11b)), while the reported durations are distributed between 29 and 360 minutes (see 11a) and manually detected disruptions distributed between 75 and 440 minutes), which highlights that disruptions observed from traffic speed are much shorter than reported in the original data set, 4) There is no noticeable correlation between observed and reported durations with high amount of horizontal anomalies in reported accident durations (see Figure 11). Traffic accident duration is most common to follow log-normal or log-logistic distribution [17] and on resulting plots, we see that accident reports are found to represent log-normal distribution to less extent than manual markup or estimated accident duration.

To perform the ablation study, we perform a manual markup of disruptions observed in traffic speed for 800 accidents, which will be discussed in the corresponding section.

F. Extraction of Disruption Shapes

In previous subsections we applied a Chebyshev metric to perform segmentation of disruptions. To analyse the disruption impact we apply the Wasserstein difference between monthly speed profile and daily traffic speeds and extract the corresponding disruption intervals. Wasserstein difference, originally named an Earth Mover distance, has an intuitive physical interpretation - the minimum “cost” of altering one pile of earth into the other, which is assumed to be the amount of earth that needs to be moved times the mean distance it has to be moved. In application to traffic state, it is the minimum amount of work necessary to alter the traffic state to disrupted condition, or in other words - the amount of disruption. We compare normalized metric values since every at every vehicle detector station there is a different average traffic speed. As in our proposed algorithm, we use a 12-units moving window (one hour) to estimate the Wasserstein difference between traffic speed measurements and provide the plot for the first 40 segmented disruptions, which allows for shape analysis of traffic disruption amount (see Figure 12): 1) We observe the similarity between multiple disruptions - they have a ‘hill’ shape, 2) there are secondary (double ‘hill’) and long-lasting disruptions. The observed shapes can be defined through the parametric equation to perform the classification of disruption effects and facilitate the prediction of disruption impact timeline since we observe that high-peak fast-ascending disruptions have a probability to end sooner than slowly ascending ones. The analysis of the speed of ascendance has potential to perform the early classification of disruptions, which is planned for further research.

G. Accident Duration Prediction

We further compare a regression model prediction performance on the CTADS data set by using on the training data set both our estimated versus the reported accident durations. We report results of a 10-fold cross-validation over 820 accident reports for which we performed a Vehicle Detector Station association and manual markup of traffic disruptions from traffic speed for ablation study. Firstly, we need to

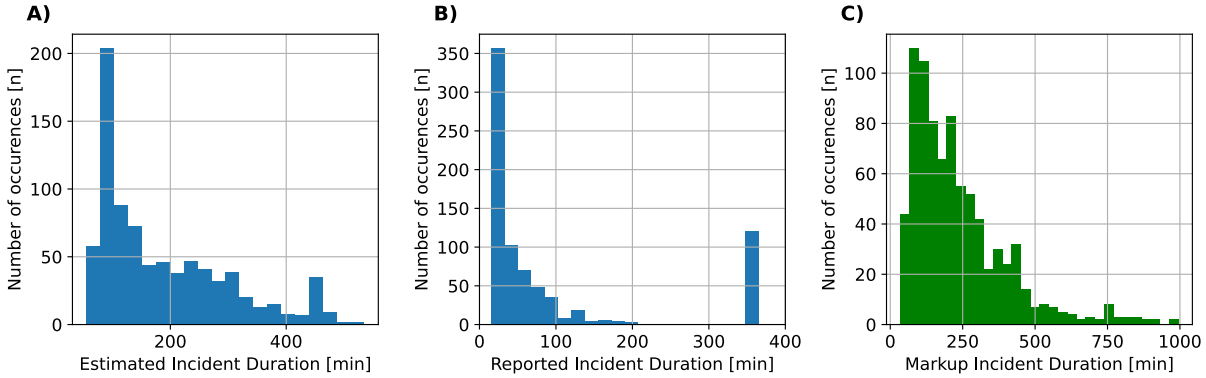

Fig. 10. Distribution of accident durations for a) estimated, b) reported accident durations for the area of San Francisco, c) results of manual markup of disruptions observed in traffic speed.

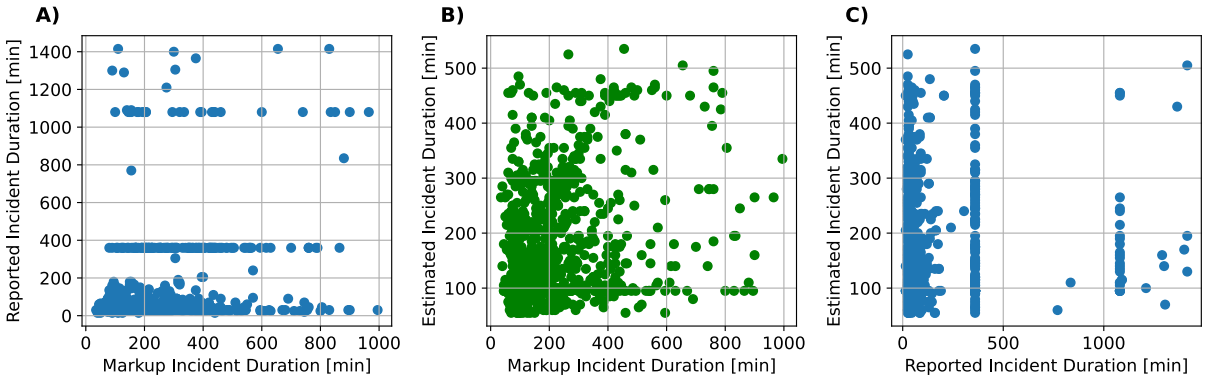

Fig. 11. Scatter plot for a) estimated and b) reported accident durations for the area of San Francisco, c) results of manual markup of disruptions observed in traffic speed.

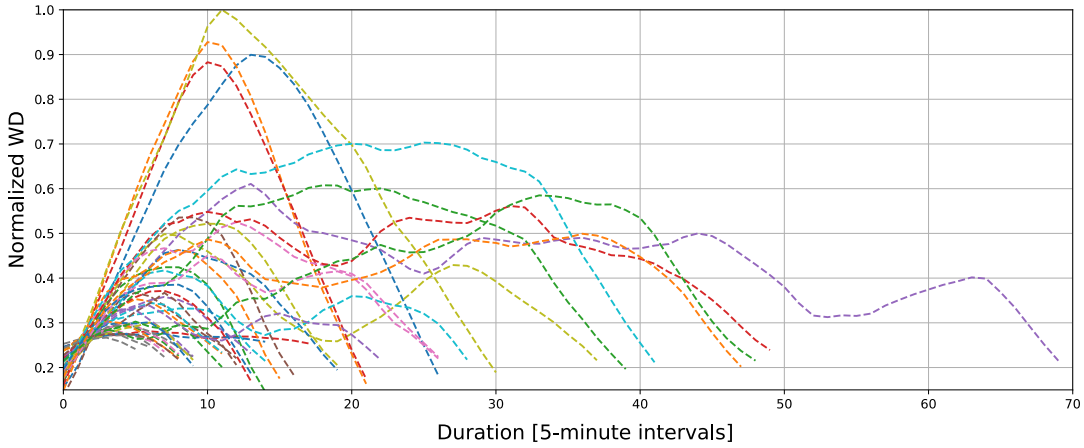

Fig. 12. Normalized Wasserstein distance plot for disruption shapes extracted for segmented intervals.

consider that the performance using reported durations from CTADS can be affected because of the presence of user-input errors in the form of placeholder values. Secondly, the nature of estimated accident durations is different since accident response teams usually report the end of the accident at the moment they finished the accident clearence, without estimating the time for the traffic to return to a normal condition, which would require additional presence, calculations and access to measurements.

We have further extended the current results by adding newtables with several machine learning models on the task of predicting a target variable.

Table II shows the Mean Absolute Error (MAE) results. The model with the lowest MAE is the CatBoost model, with an estimated MAE of 17.55, followed by the Ridge Regression with an estimated MAE of 17.87. The highest MAE is reported by the Linear Regression model (76.76). The CatBoost model outperforms all the other models by a significant margin, with the next best model (Ridge Regression) having an estimated MAE that is only slightly lower.

Table III shows the Root Mean Squared Error (RMSE) results. Here, the CatBoost model also has the lowest RMSE, with an estimated value of 22.55. The next best model is the Ridge Regression with an estimated RMSE of 22.21.

TABLE II
MEAN ABSOLUTE ERROR (MAE) RESULTS

Model	Reported	Manual	Estimated
RandomForest [33]	26.52	21.89	17.21
XGBoost [34]	24.22	23.06	18.29
GBDT [37]	26.50	22.37	17.46
CatBoost [36]	23.96	21.58	17.55
LightGBM [35]	36.57	22.43	18.26
KNN [31]	44.11	26.22	19.73
LinearRegression	76.76	24.12	17.82
SVM [32]	84.82	23.70	17.55
NeuralNetwork [55]	55.34	24.33	19.27
RidgeRegression [56]	84.72	24.26	17.87
Target	(Reported)	(Manual)	(Estimated)

TABLE III
ROOT MEAN SQUARED ERROR (RMSE) RESULTS

Model	Reported	Manual	Estimated
GBDT [37]	73.14	30.46	22.11
CatBoost [36]	73.05	29.64	22.55
Random_Forest [33]	93.73	29.94	21.79
XGBoost [34]	82.67	31.75	23.61
LightGBM [35]	99.77	30.55	23.58
KNN [31]	142.97	35.27	24.45
Linear_Regression	117.53	32.54	22.35
SVM [32]	208.29	34.23	23.66
Neural_Network [55]	124.21	33.38	23.18
Ridge_Regression [56]	134.71	32.48	22.21
Target	(Reported)	(Manual)	(Estimated)

The highest RMSE is reported by the SVM model, with an estimated value of 208.29. As with the MAE results, the CatBoost model outperforms all the other models by a significant margin. All the methods use default parameters as they are presented in Scikit-learn [54] and corresponding modules.

When we are using accident reports to predict the estimated accident duration, we obtain a better performance using the RMSE metric across all the regression models, which may be connected to the lower amount of long accident durations than reported.

These best-performing models are all complex tree methods, which utilize multiple learners (via ensembles and boosting) to gain better predictive performance. They work well with mixed types of data (numeric and categorical), can capture non-linear relationships, and are less prone to overfitting. On the contrary, Linear Regression assumes a linear relationship between the input variables and the single output variable, KNN assumes that similar instances are near to each other, and SVM assumes that the data is linearly separable by a hyperplane in a feature space. Low performance of these methods shows that these assumptions may not align well with the data in case of traffic accident reports.

The reported duration, as provided directly from the source or via some other form of direct measurement is subject to more variability due to factors such as measurement errors (incorrectly reported duration), reporting biases ("rounded" 30 and 360 minute durations), or other uncontrolled external influences (late accident detection, disruption effects mis-aligned to reported accident timeline). We expect that correct estimation of the incident duration contributes to reduction in modelling complexity due to reduced effect of outliers, bias and errors on prediction performance.

In contrast, the manual and estimated durations are derived using more controlled processes and algorithms. The manual duration calculated by a consistent procedure, minimizing the room for error. The estimated duration, relies on parametric model, would also tend to have less variation due to the model fine-tuning to minimize prediction error based on the available data.

Overall, the CatBoost model consistently outperforms all the other models across all metrics.

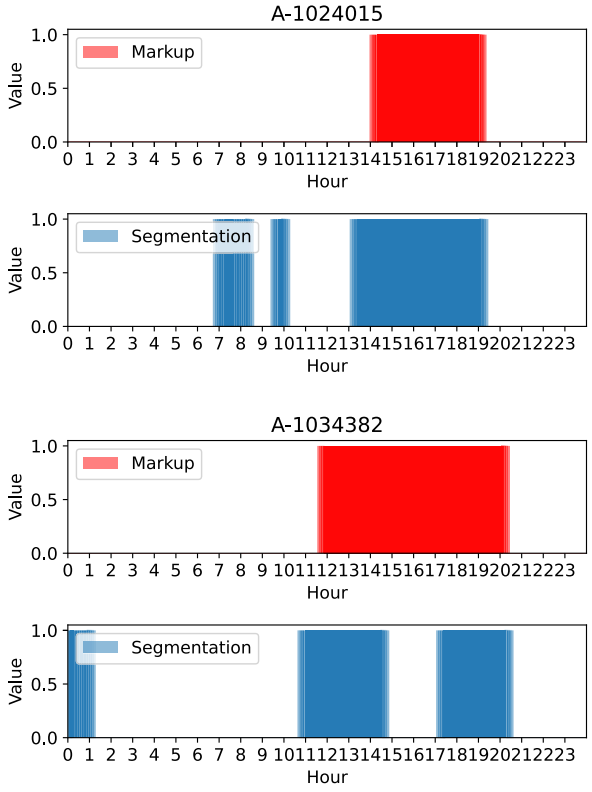

Fig. 13. Manual markup and algorithm segmentation comparison. Time series segments represented as binary values of 0 and 1.

V. ABLATION STUDY

In this paper, we propose using the F1 score to estimate the quality of time interval segmentation in binary time series (see Figure 13) in which we provide two different examples of different stations with both manual markups of the incidents - red markups- and our segmentation algorithms - blue markups- that is more efficient at detecting multiple incidents throughout the 24h time period and not only one single isolated event. The value on Y-axis shows a positive 1.0 value if the interval contains the disruption. Examples are provided for Accidents with ID A-1024015 and A-1034382 from CTADS data set.

Given a ground truth dataset with original reported accident duration, we perform a manual labelling of segments and obtain a set of predicted segments obtained from our automated segmentation algorithm, we compute the precision

and the recall of the algorithm, and then combine them into a single F1 score.

F1-score is a popular metric used to evaluate the quality of binary classification models defined as follows:

$$\text{precision} = \frac{\text{true positives}}{\text{true positives} + \text{false positives}}$$
$$\text{recall} = \frac{\text{true positives}}{\text{true positives} + \text{false negatives}}$$

where true positives are the number of correctly classified positive instances, false positives are the number of negative instances classified as positive, and false negatives are the number of positive instances classified as negative.

F1-score is defined as the harmonic mean of precision and recall, given by:

$$\text{F1-score} = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$$

F1-score ranges from 0 to 1, with higher values indicating a better classification performance.

In the case where a time series is represented as a series of points with values of 1 for segmented intervals and 0 for intervals with no segments, F1-score can be applied to estimate the quality of the time interval segmentation.

To apply the F1-score, we need a ground truth dataset with manually labelled segments (and we obtain this manual markup for 820 accidents), and a set of predicted segments obtained from our automated segmentation algorithm. We can use these two sets to compute the precision and recall of the segmentation algorithm, and then combine them into a single F1-score.

Precision measures the proportion of true positives among all the predicted positives. In the context of time interval segmentation, the precision measures the accuracy of the algorithm in detecting the true segments. The Recall measures the proportion of true positives among all the actual positives. In the context of time interval segmentation, the recall measures the completeness of the algorithm in detecting all the true segments.

To apply the F1-score to estimate the quality of time interval segmentation, we can compute the precision and recall for each segment, and then compute the overall F1-score as the weighted average of precision and recall, weighted by the number of segments. This provides a single metric that reflects the quality of the time interval segmentation.

As a result (see Figure 14), the official reported incident segmentation is found to be very off (with a mean F1-score of 0.29 - Figure 14a)); next, the segmentation done by the algorithm while selecting only the interval closest to the reported timeline yields the highest average F1-score of 0.51 - Figure 14c)) with a peak at 0.3; lastly, when considering multiple segmented incident intervals detected from our algorithm, it produced a slightly lower F1 score of 0.47 - Figure 14b)), but more evenly distributed. Overall, the algorithm performance that we propose in this paper yields a higher precision in detecting disruptions from time series of traffic speeds than from the reported accident timeline. The use of multiple segments produced by the algorithm can highlight multiple disruptions while producing just a slight decrease in the quality of results. The error for multiple intervals segmentation increases because more additional intervals are considered in the evaluation of the metric, which may lay outside of originally marked intervals (see Figures 13 and 9).

A. False Positives Rate Analysis

The issue of false alarms in the incident detection task can be significant. Traffic authorities may need the control over incident detection specificity. Since our segmentation algorithm provides real values after applying a difference metric, the value of false positives can be controlled by selecting an appropriate threshold of binarization. We provide a receiver operating characteristic curve (ROC) curve for comparison across total merged timeline of incidents and represent manual and estimated segmentation procedures as a binary classification problem. Parameters like granularity and binarization threshold can be fine-tuned according to specific metric (e.g. Area under ROC curve, F1-score or heuristics of metrics) to increase the amount of true positives while reducing the amount of false positives. We utilized F1-score as it able to grasp both of these values in a single formula. As shown on Figure 15, our proposed methodology, even without the tuning of hyper-parameters, allows to maintain a high detection rate while keeping the false alarm rate low.

We further look into specifics of disruption detection for various accidents (see Figures 16 and 17). For some accidents, high detection rate cannot be achieved without increasing the false positives rate. It is important to note that the selection of the binarization threshold plays a crucial role in controlling algorithm performance. A lower threshold might increase the sensitivity, thereby escalating the detection rate, but at the cost of specificity, leading to more false positives. Conversely, a higher threshold might reduce false alarms but may also miss some real incidents, thus lowering the detection rate. Therefore, the end users can fine-tune the parameters according to their specific needs, demonstrating the flexibility and adaptability of our proposed methodology.

B. Parameter Importance Study

For our model, we have the following variables and their intervals of variation:

- **gran**: Granularity, an integer value controlling the level of detail (moving window size) in the metric estimation function. In the provided search space, the range of gran is [2, 40] with a step of 1. Default value is 12.
- **kernel_size**: A list of float values used as weights in the dilation convolution operation. The search space for the kernel is the size of the convolution [1, . . . 4], float values primarily intended to implement pre-processing operation for the day time-series. Default value is 3.
- **selectivity**: A float value between 0.01 and 4.0 that determines the power coefficient in post-processing difference estimations Default value is 2.0.
- **shift**: An integer value between -32 and +32 that represents a cyclic shift of the resulting time series to attribute

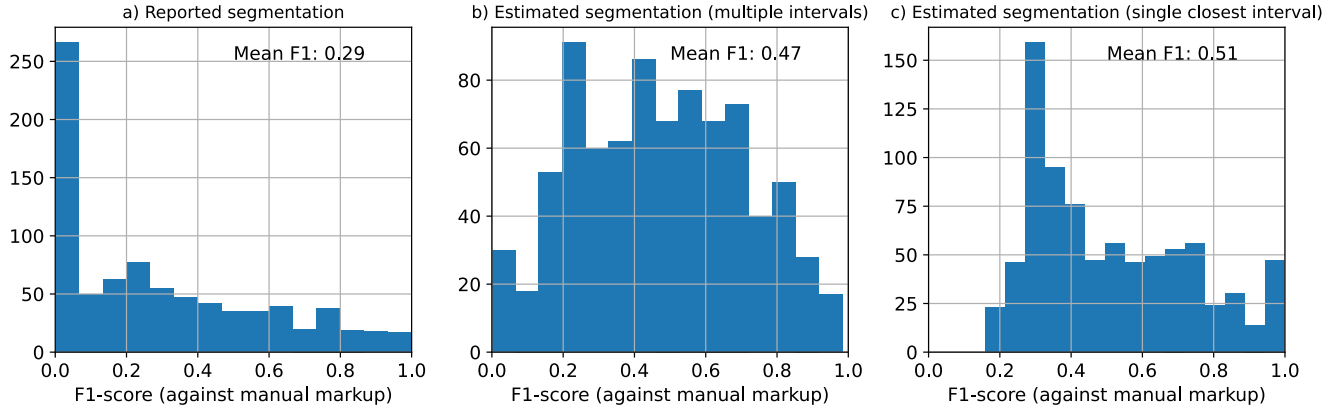

Fig. 14. Histogram of F1-score against manual markup for a) reported accident time interval and b) estimated segmentation when algorithm detecting multiple disruption intervals c) estimated segmentation for the single closest interval to reported incident occurrence time.

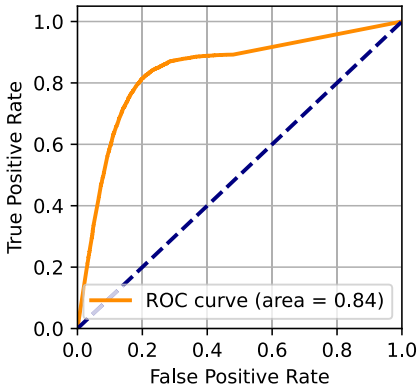

Fig. 15. Receiver operating characteristic for all accidents. Markup vs Estimation.

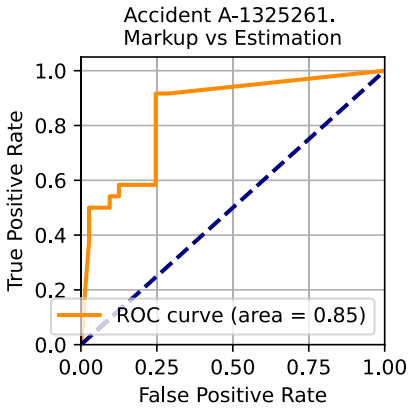

Fig. 16. Receiver operating characteristic for accident A-1325261. Markup vs Estimation.

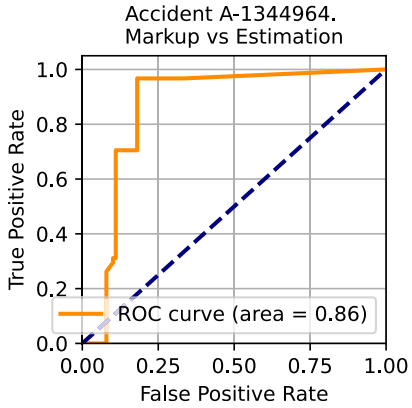

Fig. 17. Receiver operating characteristic for accident A-1344964. Markup vs Estimation.

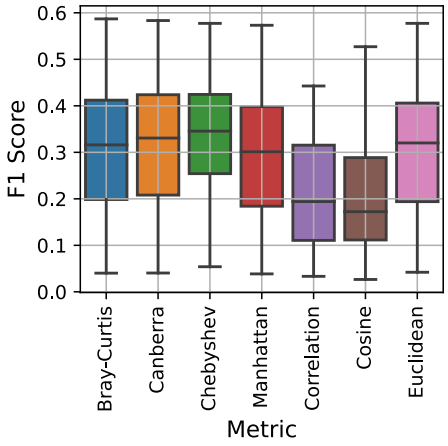

Fig. 18. Connection between metric and F1 score.

to a shift in convolution operation and facilitate to overall adaptation to the target segmentation. Default value is 0.

- **threshold**: A float value that serves as a threshold in the interval processing function, which is used to perform the binarization of the normalized output array by disruption degree. In the provided search space, the range of the search space for the threshold is [0.01, 0.99]. Default value is 0.15.

At the begininning we perform a hyper-parameter search across all the mentioned parameters but also include a search among metric list (Bray-Curtis, Canberra, Chebyshev, Manhattan, Correlation, Cosine, Euclidean, Minkowski difference metrics) to determine the best performing difference metric for our algorithm. By performing search across 3,000 iterations we then estimate the avrage f1 score obtained when using each metric (see Figure 18). The Chebyshev metric yields higher f1 score than other metrics, possibly due to the structure and interpretation of the metric: high difference between

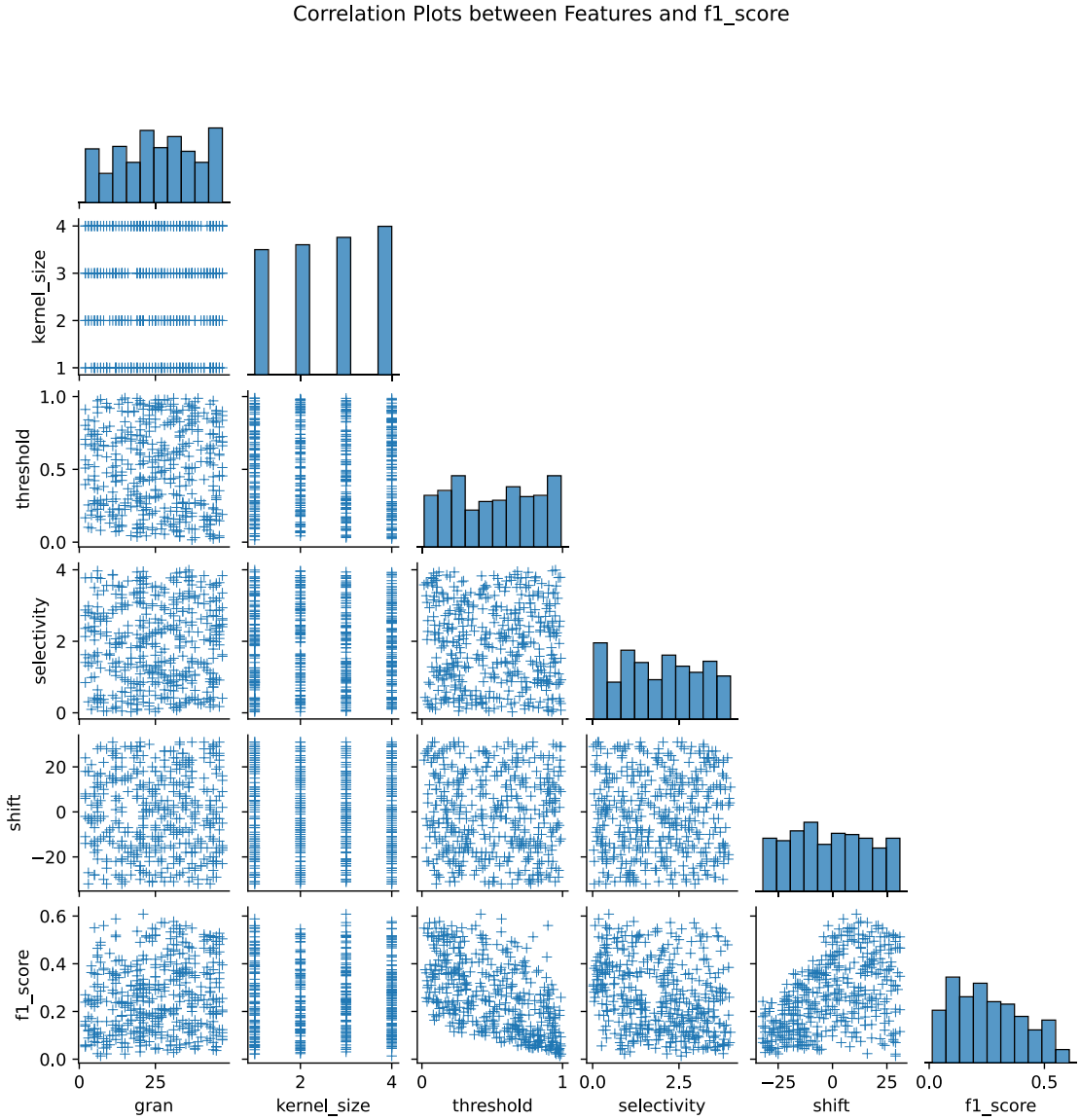

Fig. 19. Scatter plots between model parameters and F1 score.

maximum and minimum traffic speed measurements within a time window can indicate the presence of the disruption.

Our next step is to perform a hyper-parameter search for the Chebyshev difference metric only for 1,000 iterations. We obtained a significant improvement in the average f1 score for multiple interval comparison - 0.62 (a significant improvement from 0.52). As can be seen from scatter plots (see Figure 19), there are noticeable positive (kernel size vs f1 score), negative (binarization threshold vs f1 score) and peaking trends (shift vs f1 score) observed in results. Optimal values for the binarization threshold are located at lower values (between 0.01 and 0.4). Overall, the algorithm requires a positive shift in the post-processing function, which contributes to a substantial increase from 0.52 to 0.62 in f1 score when considering the positive shift of the resulting array.

We further provide a Correlation heatmap between algorithm parameters (see Figure 20) and the resulting f1 score: 1) The highest Pearson correlation values are with variables threshold (-0.55), shift (0.49), followed by selectivity (-0.32). There are no significant correlations between model parameters themselves.

In conclusion, the hyper-parameter search led to the selection of the Chebyshev metric, which demonstrated the highest average F1 score. Fine-tuning the disruption segmentation algorithm hyper-parameters significantly improved the average F1 score. Trends and optimal parameter values were identified, and the correlation heatmap showed that threshold, shift, and selectivity had the highest Pearson correlation with the F1 score.

VI. Application of the Methodology

We further publish the code and describe the functionality of the toolkit which can be described as a novel approach for associating geospatial and traffic data to traffic incidents, using data from three different sources such as OpenStreetMap (OSM), the Countrywise Traffic Accident Data Set (CTADS), and the Performance Measurement System (PeMS). Data preprocessing includes various steps, such as filtering data by

TABLE IV
INPUTS, OUTPUTS, AND USE CASES OF AAA TOOLKIT CODE PARTS

Code Part	Input	Output	Use Case
Download OSM data	Geographic coordinates or region name	Raw OSM data file	-
Download CTADS data	Geographic coordinates or region name, CTADS database access	Raw CTADS data file	-
Filter CTADS and OSM by area	Raw OSM and CTADS data files	Filtered OSM and CTADS data files	-
Convert OSM to points	Filtered OSM data file	Point-based OSM data	Simplifies the geographical data to allow point-based algorithm processing.
Get VDS points from PEMS	Geographic coordinates or region name, PEMS database access	VDS data file	-
Apply road alignment	Point-based OSM data, VDS data, and CTADS data	Aligned VDS and CTADS data with OSM road points	Initial stage for the association of traffic flow/speed and accident data with specific roads, enabling more accurate accident modelling. Allows algorithms application from road point of view.
Apply CTADS2VDS algorithm	Aligned VDS and CTADS data	Associated VDS points with accident points	Links traffic speed/flow detector ID and accident report data.
Download traffic speed data (CTADS2TS)	CTADS data with accident dates, PEMS database access	Traffic speed data for the incident day and the two weeks prior	Used to investigate the impact of traffic accidents on traffic speed, leading to more informed speed limit policies.
Apply segmentation algorithm	Associated VDS points with accident points, traffic speed data	Time series of disruption degree, disruption intervals associated with incidents	Enables the time series analysis assiciated with the disruption.

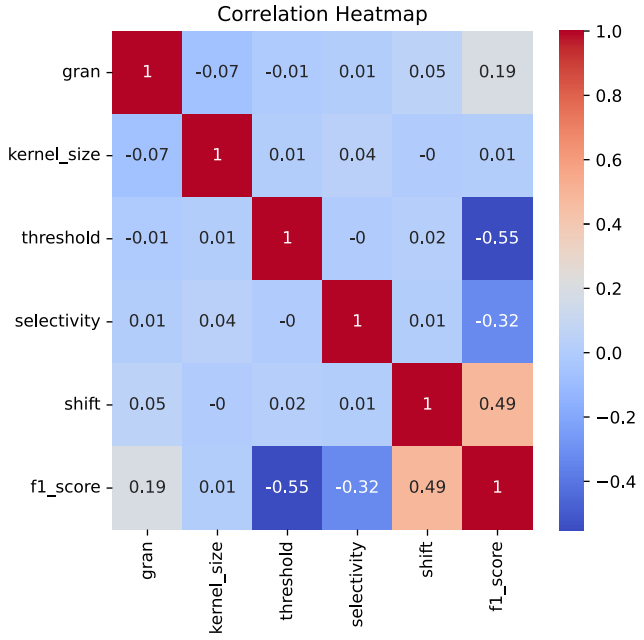

Fig. 20. Scatter plot between model parameters and F1 score.

geographic area, converting road data to point data, and aligning different datasets based on corresponding road points, the toolkit manages to link traffic conditions with incident locations. A specialized algorithm, referred to as CTADS2VDS, is applied to form these associations. Furthermore, the toolkit retrieves traffic speed data for each day of an incident and for the preceding two-week period, offering insights into traffic conditions leading up to the accident.

For authorities, such a toolkit can be incredibly valuable. Not only can it provide comprehensive information about the conditions of traffic accidents, it can also help in identifying patterns or trends related to accident progression and the traffic conditions leading up to these incidents. This can be useful for traffic management, road infrastructure planning, and the design of traffic safety measures. The data-driven approach used by the toolkit can enable authorities to make informed decisions based on observed traffic speed disruption, rather than relying on reported estimates or assumptions. The methodology followed by this toolkit is modular and versatile. It can be adjusted and optimized based on specific requirements or challenges encountered in different regions. Alternatively, the toolkit can be used to monitor the effectiveness of traffic safety measures by comparing the rate of observed disruptions before and after the implementation of these measures. The capability to analyze accident duration or severity in relation to traffic flow data can also assist authorities in prioritizing their efforts to improve road safety.

There are several potential obstacles related to the code and its adaptation for real-world scenarios, including: 1) Scalability: Handling big data environments, especially in larger urban networks, requires optimized algorithms capable of distributed computing. Two main resource-intensive components are traffic speed/flow data retrieval and CTADS2VDS point mapping, both of which can be improved by implementing parallel versions of algorithms, 2) Interoperability: The original code must be able to parse and handle OSM map and CTADS incident report data formats (e.g., CSV, OSM) and connect to the PeMS database. In the case of alternative data sources and formats, it requires implementing additional data preprocessing steps, 3) Algorithm Precision: performance and limitations of disruption segmentation algorithms explored in the results section. These issues can be mitigated by the following properties of our methodology: 1) Algorithm Fine-Tuning:

Fine-tuning of the disruption segmentation algorithm can be performed automatically using hyper-parameter search for best performance on alternative sources of data, 2) Sensitivity-specificity control: maintaining high incident detection rates while minimizing false alarms is a key challenge. The disruption segmentation algorithm allows us to estimate the "degree" of disruption before applying the binarization threshold. This property allows for false-alarm control using fine-tuning of the detection threshold. Balancing sensitivity (identifying real incidents) and specificity (avoiding false alarms) often involves trade-offs and can be fine-tuned to specific data set.

In urban networks with hundreds of measurement locations, the data retrieval is a bottleneck, since each accident report will require a request for daily and fortnight measurements at specific detectors. Depending on the speed of VDS data retrieval, the amount of data that can be obtained in an acceptable amount of time can be limited.

The code for the paper can be found by the following link: https://github.com/Future-Mobility-Lab/AAA-toolkit/tree/main.

The Table IV presented below provides a summary of the key parts involved in the Accident Analysis& Association (AAA) Toolkit codebase. Each row represents a specific segment of the code, outlining the corresponding inputs required and outputs produced for each segment. The sequence of code parts indicates the flow of data and the transformation processes that occur from acquiring the initial raw data to ultimately applying segmentation algorithms on the compiled information.

VII. Conclusion

Our methodology aims to automatically detect, segment, and extract traffic disruptions and accidents using distance metrics. This approach improves incident prediction accuracy across multiple machine learning models and provides better fit to manual markup of observed traffic speed disruptions. By obtaining the intervals and shapes of traffic disruptions, we can model the impact of accidents with greater precision, using traffic state measurements rather than just reported parameters (duration, start time, etc). This approach provides more data on the accident and allows us to study accident impacts in greater detail.

A. Relevance of This Work Can be Summarized in Following Points

1) Enhancement of Traffic Management Systems: Integrate the proposed early detection and disruption segmentation algorithm into existing traffic management systems to improve and automate incident detection and corresponding data collection. This will help to minimize congestion and the overall impact of incidents on traffic flow, 2) highlight of reporting errors to standardize data reporting: Establish standardized guidelines and protocols for reporting traffic incidents, including the accurate reporting of the location, start and end times, number of lanes affected, and other relevant details; this will ensure that data-driven models can accurately predict incident severity and disruption length, 3) highlight the necessity of creating of data standards policies across countries for collecting necessary traffic accident information, 4) development of Incident Response Strategies by utilizing the improved incident prediction models to develop data-driven incident response strategies, including the dynamic traffic rerouting and real-time traffic guidance; this will help to mitigate the impact of traffic incidents on road users and reduce the risk of secondary incidents; 5) Data Fusion for a better traffic accident analysis: due to observed improvement in the quality of prediction arising from data fusion, traffic Authorities can consider integrating data sets from private companies for jointly analysing traffic datasets of various types to improve traffic safety by improving accuracy of traffic incident duration prediction.

B. Future Research in This Area

1) Algorithm's complexity can be expanded by incorporating custom kernels, which can be found using hyper-parameter search, 2) Disruption measurements obtained over time can enable the prediction of traffic incident impact propagation with greater accuracy than relying solely on reported values, 3) The proposed methodology can be extended to include disruptions beyond accidents, such as construction or road closures, which can improve the accuracy of impact prediction, 4) Further improvement can also be achieved by performing data fusion and incorporating external data sources, such as weather and events, into the incident impact prediction models. We are currently modelling the cascading effect on traffic disruptions and how these can be automatically identified based on multiple incoming traffic state streams; the main challenge of detecting subsequent incidents lie in the time-span duration of the first incident which is normally stochastic in nature.

C. Limitations of This Work

The current modelling approach has been applied to a San Francisco data set due to its public availability and easiness to access. However, we would like to test the approach on multiple other countries and incident databases across the globe; the main challenge is the lack of both traffic states and traffic accidents logs to be released with synchronised timelines.

Authors and Contribution Statement

The authors confirm their contribution to the paper as follows: study conception and design: Artur Grigorev, Adriana-Simona Mihăiţă, and Fang Chen; data collection: Artur Grigorev; analysis and interpretation of results: Artur Grigorev and Khaled Saleh; draft manuscript preparation: Artur Grigorev and Adriana-Simona Mihăiţă. All authors reviewed the results and approved the final version of the manuscript.

Acknowledgment

The authors would like to thank the support of Transport for NSW, Australia.

REFERENCES

[1] *Global Status Report on Road Safety 2015*, World Health Organization, Geneva, Switzerland, 2015.

[2] *Traffic Safety Facts 2013, U.S. Department of Transportation*, Nat. Highway Traffic Saf. Admin., Washington, DC, USA, 2013.

[3] W. Kim and G.-L. Chang, "Development of a hybrid prediction model for freeway incident duration: A case study in Maryland," *Int. J. Intell. Transp. Syst. Res.*, vol. 10, no. 1, pp. 22–33, 2011, doi: 10.1007/s13177-011-0039-8.

[4] A. Theofilatos, G. Yannis, P. Kopelias, and F. Papadimitriou, "Predicting road accidents: A rare-events modeling approach," *Transp. Res. Proc.*, vol. 14, pp. 3399–3405, Jan. 2016, doi: 10.1016/j.trpro.2016.05.293. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S235214651630299X

[5] A. J. Khattak, J. Liu, B. Wali, X. Li, and M. Ng, "Modeling traffic incident duration using quantile regression," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 2554, no. 1, pp. 139–148, Jan. 2016.

[6] A. Mehdizadeh et al., "A review of data analytic applications in road traffic safety. Part 1: Descriptive and predictive modeling," *Sensors*, vol. 20, no. 4, p. 1107, Feb. 2020.

[7] J. Wang, K. Li, and X.-Y. Lu, "Chapter 5. Effect of human factors on driver behavior," in *Advances in Intelligent Vehicles*. London, U.K.: Academic, 2013, pp. 111–155, doi: 10.1016/B978-0-12-397199-9.00005-7.

[8] M. S. Horswill and M. E. Coster, "The effect of vehicle characteristics on drivers' risk-taking behaviour," *Ergonomics*, vol. 45, no. 2, pp. 85–104, Feb. 2002, doi: 10.1080/00140130110115345.

[9] R. B. Noland and L. Oh, "The effect of infrastructure and demographic change on traffic-related fatalities and crashes: A case study of Illinois county-level data," *Accident Anal. Prevention*, vol. 36, no. 4, pp. 525–532, Jul. 2004.

[10] A. Winder, M. Brackstone, and P. Site, "Traffic management for land transport: Research to increase the capacity, efficiency, sustainability and safety of road, rail and urban transport networks," Transp. Res. Knowl. Centre (TRKC), Tech. Rep., 2023.

[11] A. Grigorev, A.-S. Mihaita, S. Lee, and F. Chen, "Incident duration prediction using a bi-level machine learning framework with outlier removal and intra–extra joint optimisation," *Transp. Res. C, Emerg. Technol.*, vol. 141, Aug. 2022, Art. no. 103721, doi: 10.1016/j.trc.2022.103721. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0968090X22001589

[12] A. S. Mihaita, Z. Liu, C. Cai, and M. Rizoiu, "Arterial incident duration prediction using a bi-level framework of extreme gradient-tree boosting," 2019, *arXiv:1905.12254*.

[13] B. Ghosh, M. T. Asif, J. Dauwels, U. Fastenrath, and H. Guo, "Dynamic prediction of the incident duration using adaptive feature set," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 11, pp. 4019–4031, Nov. 2019.

[14] L. Dimitriou and E. I. Vlahogianni, "Fuzzy modeling of freeway accident duration with rainfall and traffic flow interactions," *Analytic Methods Accident Res.*, vol. 5, pp. 59–71, Jan. 2015.

[15] S. Shafiei, A. Mihaita, H. Nguyen, C. Bentley, and C. Cai, "Short-term traffic prediction under non-recurrent incident conditions integrating data-driven models and traffic simulation," in *Proc. Transp. Res. Board 99th Annu. Meeting*, 2020, pp. 1–24.

[16] T. Mao, A.-S. Mihaita, F. Chen, and H. L. Vu, "Boosted genetic algorithm using machine learning for traffic control optimization," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 7, pp. 112–7141, Jul. 2022.

[17] R. Li, F. C. Pereira, and M. E. Ben-Akiva, "Overview of traffic incident duration analysis and prediction," *Eur. Transp. Res. Rev.*, vol. 10, no. 2, pp. 1–13, Jun. 2018.

[18] S. Fukuda, H. Uchida, H. Fujii, and T. Yamada, "Short-term prediction of traffic flow under incident conditions using graph convolutional recurrent neural network and traffic simulation," *IET Intell. Transp. Syst.*, vol. 14, no. 8, pp. 936–946, Aug. 2020.

[19] A. B. Parsa, H. Taghipour, S. Derrible, and A. Mohammadian, "Real-time accident detection: Coping with imbalanced data," *Accident Anal. Prevention*, vol. 129, pp. 202–210, Aug. 2019.

[20] L. Eboli, C. Forciniti, and G. Mazzulla, "Factors influencing accident severity: An analysis by road accident type," *Transp. Res. Proc.*, vol. 47, pp. 449–456, Jan. 2020, doi: 10.1016/j.trpro.2020.03.120. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2352146520303197

[21] T. Tsubota, C. Fernando, T. Yoshii, and H. Shirayanagi, "Effect of road pavement types and ages on traffic accident risks," *Transp. Res. Proc.*, vol. 34, pp. 211–218, Jan. 2018, doi: 10.1016/j.trpro.2018.11.034. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2352146518303235

[22] G. Yannis et al., "Use of accident prediction models in road safety management—An international inquiry," *Transp. Res. Proc.*, vol. 14, pp. 4257–4266, 2016, doi: 10.1016/j.trpro.2016.05.397. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2352146516304033

[23] C. C. Aggarwal and C. C. Aggarwal, "Probabilistic and statistical models for outlier detection," in *Outlier Analysis*. Cham, Switzerland: Springer, 2013, pp. 41–74.

[24] J. Takeuchi and K. Yamanishi, "A unifying framework for detecting outliers and change points from time series," *IEEE Trans. Knowl. Data Eng.*, vol. 18, no. 4, pp. 482–492, Apr. 2006.

[25] G. E. P. Box and D. A. Pierce, "Distribution of residual autocorrelations in autoregressive-integrated moving average time series models," *J. Amer. Stat. Assoc.*, vol. 65, no. 332, pp. 1509–1526, Dec. 1970.

[26] R. G. Vieira, M. A. L. Filho, and R. Semolini, "An enhanced seasonal-hybrid esd technique for robust anomaly detection on time series," in *Proc. Anais do 36th Simp. Brasileiro de Redes de Computadores e Sistemas Distribuídos*, 2018, pp. 281–294.

[27] P. Malhotra, L. Vig, G. Shroff, and P. Agarwal, "Long short term memory networks for anomaly detection in time series," in *Proc. ESANN*, 2015, p. 89.

[28] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation forest," in *Proc. 8th IEEE Int. Conf. Data Mining*, Dec. 2008, pp. 413–422.

[29] S. Zhong, S. Fu, L. Lin, X. Fu, Z. Cui, and R. Wang, "A novel unsupervised anomaly detection for gas turbine using isolation forest," in *Proc. IEEE Int. Conf. Prognostics Health Manag. (ICPHM)*, Jun. 2019, pp. 1–6.

[30] L. M. Manevitz and M. Yousef, "One-class SVMs for document classification," *J. Mach. Learn. Res.*, vol. 2, pp. 139–154, Dec. 2001.

[31] L. Kuang, H. Yan, Y. Zhu, S. Tu, and X. Fan, "Predicting duration of traffic accidents based on cost-sensitive Bayesian network and weighted K-nearest neighbor," *J. Intell. Transp. Syst.*, vol. 23, no. 2, pp. 161–174, Mar. 2019.

[32] S. Xiao, "Traffic accident duration prediction based on natural language processing and a hybrid neural network architecture," *Proc. SPIE*, vol. 11933, pp. 194–202, Oct. 2021.

[33] K. Hamad, R. Al-Ruzouq, W. Zeiada, S. Abu Dabous, and M. A. Khalil, "Predicting incident duration using random forests," *Transportmetrica A, Transp. Sci.*, vol. 16, no. 3, pp. 1269–1293, Jan. 2020, doi: 10.1080/23249935.2020.1733132

[34] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2016, pp. 785–794.

[35] G. Ke et al., "LightGBM: A highly efficient gradient boosting decision tree," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 3146–3154.

[36] A. V. Dorogush, V. Ershov, and A. Gulin, "CatBoost: Gradient boosting with categorical features support," 2018, *arXiv:1810.11363*.

[37] J. Ye, J.-H. Chow, J. Chen, and Z. Zheng, "Stochastic gradient boosted distributed decision trees," in *Proc. 18th ACM Conf. Inf. Knowl. Manag.*, Nov. 2009, pp. 2061–2064.

[38] J. Tang, L. Zheng, C. Han, F. Liu, and J. Cai, "Traffic incident clearance time prediction and influencing factor analysis using extreme gradient boosting model," *J. Adv. Transp.*, vol. 2020, pp. 1–12, Jun. 2020, doi: 10.1155/2020/6401082.

[39] H. Zhao, W. Gunardi, Y. Liu, C. Kiew, T.-H. Teng, and X. B. Yang, "Prediction of traffic incident duration using clustering-based ensemble learning method," *J. Transp. Eng., A, Syst.*, vol. 148, no. 7, Jul. 2022, Art. no. 04022044, doi: 10.1061/JTEPBS.0000688.

[40] Q. Shang, T. Xie, and Y. Yu, "Prediction of duration of traffic incidents by hybrid deep learning based on multi-source incomplete data," *Int. J. Environ. Res. Public Health*, vol. 19, no. 17, p. 10903, Sep. 2022, doi: 10.3390/ijerph191710903. [Online]. Available: https://www.mdpi.com/1660-4601/19/17/10903

[41] A. Grigorev, M. Adriana Simona, K. Saleh, and M. Piccardi, "Traffic incident duration prediction via a deep learning framework for text description encoding," in *Proc. IEEE 25th Int. Conf. Intell. Transp. Syst. (ITSC)*, Sep. 2022, pp. 1770–1777, doi: 10.13140/RG.2.2.12647.32164.

[42] A. M. S. Alkaabi, D. Dissanayake, and R. Bird, "Analyzing clearance time of urban traffic accidents in Abu Dhabi, United Arab Emirates, with hazard-based duration modeling method," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 2229, no. 1, pp. 46–54, Jan. 2011.

[43] M. Miller and C. Gupta, "Mining traffic incidents to forecast impact," in *Proc. ACM SIGKDD Int. Workshop Urban Comput.*, Aug. 2012, pp. 33–40.

[44] S. Moosavi, M. H. Samavatian, S. Parthasarathy, R. Teodorescu, and R. Ramnath, "Accident risk prediction based on heterogeneous sparse data: New dataset and insights," in *Proc. 27th ACM SIGSPATIAL Int. Conf. Adv. Geographic Inf. Syst.*, Nov. 2019, pp. 33–42.
[45] S. Moosavi, M. H. Samavatian, S. Parthasarathy, and R. Ramnath, "A countrywide traffic accident dataset," 2019, *arXiv:1906.05409*.
[46] C. Chen, K. Petty, A. Skabardonis, P. Varaiya, and Z. Jia, "Freeway performance measurement system: Mining loop detector data," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 1748, no. 1, pp. 96–102, Jan. 2001.
[47] P. Chebyshev, "Memoire sur les nombres premiers," *J. de Mathématiques Pures et Appliquées*, vol. 17, no. 1, pp. 366–390, 1853.
[48] L. Kantorovitch, "On the translocation of masses," *Manag. Sci.*, vol. 5, no. 1, pp. 1–4, Oct. 1958.
[49] G. Salton and M. J. McGill, *Introduction to Modern Information Retrieval*. New York, NY, USA: McGraw-Hill, 1988.
[50] H. Minkowski, *Raum Und Zeit*. Stutgart, Germany: S. Hirzel Verlag GmbH, 1909.
[51] J. R. Bray and J. T. Curtis, "An ordination of the upland forest communities of southern Wisconsin," *Ecological Monographs*, vol. 27, no. 4, pp. 325–349, Oct. 1957.
[52] D. Ivanković and M. K. Tiljak, "Comparison of two or more samples of quantitative data," *Acta medica Croatica, Casopis Hravatske Akademije Medicinskih Znanosti*, vol. 60, pp. 37–46, Jan. 2006.
[53] J. Zhang, K. Yang, and R. Stiefelhagen, "Exploring event-driven dynamic context for accident scene segmentation," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 3, pp. 2606–2622, Mar. 2022.
[54] F. Pedregosa et al., "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, Nov. 2011.
[55] S. I. Gallant, "Perceptron-based learning algorithms," *IEEE Trans. Neural Netw.*, vol. 1, no. 2, pp. 179–191, Jun. 1990.
[56] G. C. McDonald, "Ridge regression," *Wiley Interdiscipl. Rev., Comput. Statist.*, vol. 1, no. 1, pp. 93–100, 2009.

Artur Grigorev received the bachelor's degree in computer science and engineering from ITMO University, Russia, and the master's degree (Hons.) in applied mathematics and computer science (urban supercomputing). He is currently pursuing the Ph.D. degree with the University of Technology Sydney, researching traffic incident analysis and impact prediction under the supervision of Dr. Adriana- Simona Mihăiţă. His research interests include computer vision methods for traffic analysis.

Adriana-Simona Mihăiţă (Senior Member, IEEE) is currently a Senior Lecturer with the University of Technology Sydney, where she leads the Future Mobility Research Laboratory. Her research interests include traffic simulation and optimization using AI and machine learning, particularly for predicting traffic accidents and their urban impact, and smart analytics for connected and autonomous vehicles in a smart city environment.

Khaled Saleh (Member, IEEE) received the Ph.D. degree in computer science from Deakin University in 2019. He is an experienced AI and machine learning researcher, with over seven years of experience in both academia and industry. Previously, he was an AI Research Fellow with Deakin University, developing socially aware AI models for intent understanding of vulnerable road users.

Fang Chen is currently a distinguished professor. She is also an internationally recognized leader in artificial intelligence and data science. She is also the Executive Director of Data Science with the University of Technology Sydney (UTS) and the Leader of the UTS Data Science Institute. With over 300 publications in science and engineering, including several books, and more than 30 patents filed globally, she has a wealth of experience in developing innovative solutions for complex real-life problems in various sectors.