Expert Systems With Applications 244 (2024) 122813

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Spatiotemporal features of traffic help reduce automatic accident detection time☆

Pablo Moriano [a,*], Andy Berres [b,c], Haowen Xu [c], Jibonananda Sanyal [b]

[a] Computer Science and Mathematics Division, Oak Ridge National Laboratory, Oak Ridge, TN 37830, USA
[b] Hybrid Energy Systems Group, National Renewable Energy Laboratory, Golden, CO 80401, USA
[c] Computational Sciences and Engineering Division, Oak Ridge National Laboratory, Oak Ridge, TN 37830, USA

ARTICLE INFO

Dataset link: https://zenodo.org/record/7964288

Keywords:
Accident detection
Data fusion
Spatiotemporal analysis
Machine learning
XGBoost
SHAP

ABSTRACT

Quick and reliable automatic detection of traffic accidents is of paramount importance to save human lives in transportation systems. However, automatically detecting when accidents occur has proven challenging, and minimizing the time to detect accidents (TTDA) by using traditional features in machine learning (ML) classifiers has plateaued. We hypothesize that accidents affect traffic farther from the accident location than previously reported. Therefore, leveraging traffic signatures from neighboring sensors that are adjacent to accidents should help improve their detection. We confirm this hypothesis by using verified ground-truth accident data, traffic data from radar detection system sensors, and light and weather conditions and show that we can minimize the TTDA while maximizing classification performance by considering spatiotemporal features of traffic. Specifically, we compare the performance of different ML classifiers (i.e, logistic regression, random forest, and XGBoost) when controlling for different numbers of neighboring sensors and TTDA horizons. We use data from interstates 75 and 24 in the metropolitan area that surrounds Chattanooga, TN. Our results show that the XGBoost classifier produces the best results by detecting accidents as quickly as 1.0 min after their occurrence with an area under the receiver operating characteristic curve of up to 83% and an average precision of up to 49%. We describe limitations, open challenges, and how the proposed framework can be used for quicker operational accident detection.

1. Introduction

The recent emergence of advanced information, communication, and traffic monitoring technologies has increased the availability, spatiotemporal resolution, and quality of transportation and mobility data in many urban areas (Anda, Erath, & Fourie, 2017; Bibri & Krogstie, 2020). These technologies enable real-time and continuous collection and integration of traffic-related data (e.g., vehicle volume, speed, trajectories, and intersection performance) with large spatial coverage in the United States. These solutions include a wide variety of sensing technologies, such as radar detection systems, loop detectors, onboard GPS, mobile sensing apps, and IoT-connected cameras. These devices generate large amounts of urban mobility data, which creates opportunities for diverse, innovative, and big-data driven transportation management and smart city applications that can improve transportation efficiency, help investigate mobility dynamics, promote urban livability and sustainability, and enhance traffic safety in cities (Berres, LaClair et al., 2021; Xu et al., 2022, 2023).

Among these applications, the development of automatic incident detection (AID) methods enable the accurate detection of traffic accidents to support time-critical emergency response (e.g., dispatch medical and police resources to prevent fatality and severe infrastructure damage or reroute drivers to incident-free roadways to reduce congestion) (Han et al., 2020). As the number of traffic accidents has increased to represent a significant fraction of deaths among people in the United States and around the world (National Highway Traffic Safety Administration, 2021), there is a growing urgency to more

☆ This manuscript has been co-authored by UT-Battelle, LLC, under contract DE-AC05-00OR22725 with the US Department of Energy (DOE). The US government retains and the publisher, by accepting the article for publication, acknowledges that the US government retains a nonexclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this manuscript, or allow others to do so, for US government purposes. DOE will provide public access to these results of federally sponsored research in accordance with the DOE Public Access Plan (http://energy.gov/downloads/doe-public-access-plan).

The code (and data) in this article has been certified as Reproducible by Code Ocean: (https://codeocean.com/). More information on the Reproducibility Badge Initiative is available at https://www.elsevier.com/physical-sciences-and-engineering/computer-science/journals.

* Corresponding author.
E-mail addresses: moriano@ornl.gov (P. Moriano), andy.berres@nrel.gov (A. Berres), xuh4@ornl.gov (H. Xu), jibo.sanyal@nrel.gov (J. Sanyal).

https://doi.org/10.1016/j.eswa.2023.122813
Received 13 June 2023; Received in revised form 30 November 2023; Accepted 30 November 2023
Available online 18 December 2023

rapidly detect their occurrences and learn patterns that might be useful for predicting and preventing accidents. To that end, the AID methods aim for quick accident detection and high-quality predictions.

Many recent AID applications depend on traffic sensors to capture traffic conditions and accidents (Klein, 2001). Among them, a variety of methods that aim to automatically detect accidents focus on the use machine learning (ML) classifiers to mine traffic patterns from vast amounts of traffic data (Lu, Chen, Wang, & Van Zuylen, 2012; Motamed et al., 2016). Significant progress has been made to maximize classification performance, which is usually measured through metrics such as recall, false positive rate, and area under the receiver operating characteristic curve (AUC-ROC). However, there is still significant room for improvement for minimizing the time to detect accidents (TTDA) while maintaining high classification performance. This is because TTDA is not even considered in some studies (Li, Sheng, Du, Wang, & Ran, 2020; Lin, Li, Jing, Ran, & Sun, 2020; Liu, Cai, Zhong, Sun, & Chen, 2020; Xiao, 2019). When TTDA has been considered, the suggested TTDA is around five minutes (Parsa, Taghipour, Derrible, & Mohammadian, 2019), which can be too long for life-threatening situations caused by traffic accidents (Loten, 2019). In fact, regulators estimate that each minute saved in response time could save up to 10,000 lives annually in the United States (The Wall Street Journal, 2023).

In this paper, we investigate the feasibility of detecting traffic accidents within a much shorter TTDA (i.e., as quickly as 1.0 min) while maintaining high operational performance (i.e., AUC-ROC up to 83% and an area under the precision and recall curve [AUC-PR] or average precision up to 49%). In doing so, we use pervasive and generic roadside traffic sensor measurements (e.g., speed, volume, and occupancy) along with lighting and weather conditions. Specifically, we present the design and development of an innovative ML-powered framework that employs spatiotemporal data to automate the timely detection of traffic accidents in large geographic areas. We propose a workflow for efficient data mining and ML to optimize the analysis of complex traffic data over spatiotemporal dimensions without the need for costly high-performance computing infrastructure.

The key methodological contribution of our work is exploring the impact that accidents have on traffic measurements at sensors near the accident location. This means that we use the measurements of immediate upstream and downstream sensors, as typically done in previous research (Parsa, Movahedi, Taghipour, Derrible, & Mohammadian, 2020; Parsa et al., 2019; Shang, Feng, & Gao, 2020), but we also study how measurements from more distant sensors (i.e., from sensors more than one hop apart from accidents) help inform ML classifiers to reduce TTDA while maintaining high operational performance. Our focus is on automatically detecting accidents as soon as possible after they occur. That means that we do not focus on other types of traffic incidents or non-recurrent events, such as disabled vehicles, spilled loads, temporary maintenance and construction activities, signal and detector malfunctions, or other unusual events that disrupt the normal flow of traffic.

In this paper, we make the following contributions. First, we validate our conjecture about the practical utility of using adjacent traffic sensor measurements to reduce TTDA while maintaining high operational performance. We show that using data from more distant neighboring sensors (both upstream and downstream with respect to the accident locations) helps achieve this objective. We observe that the TTDA decreases and performance increases up to a threshold of diminishing returns. To do so, we quantify the joint effect that more distant sensors (i.e., more distant hops) along with different TTDA horizons have on the classification task. We compare the classification performance under similar conditions, using different ML classifiers, including logistic regression, random forest, and XGBoost. Then, we verify our hypothesis in more complex classification models such as random forest and XGBoost as opposed to logistic regression. XGBoost results exhibit the best trade-off between TTDA and other performance metrics.

Second, we report results on the importance of individual features in the accident detection task by using SHapley Additive exPlanations (SHAP) (Lundberg & Lee, 2017). SHAP is a model-agnostic, game-theoretic framework used to explain the output of ML models. We report the SHAP results for XGBoost, which is the best performing classifier.

Specifically, we show that traffic-related features (i.e., speed, volume, and occupancy) from both upstream and downstream distant sensors, as opposed to only upstream sensors, are the most important features that drive classification performance. This means that they support the objectives of minimizing TTDA while maximizing performance—at most 11% in AUC-ROC (from 72% to 83%) and 26% in AUC-PR (from 23% to 49%). Following traffic-related features, we show that weather- and lighting-related features have a less significant impact on the likelihood of accident occurrence.

Third, we report results on two different datasets that correspond to two major interstates (i.e., I-75 and I-24) in the Chattanooga, TN metropolitan area. We apply the proposed framework to these datasets and show that we can verify our hypothesis. Our framework allows for verifiable results when using the extended neighborhood, which includes measurements from more distant sensors, to minimize TTDA and maximize performance. We detail the entire data processing workflow, including handling the unbalanced nature of the datasets by using the Synthetic Minority Over-Sampling TEchnique (SMOTE) (Chawla, Bowyer, Hall, & Kegelmeyer, 2002), which has some history of use in the field (Parsa et al., 2020, 2019; Shang et al., 2020). In addition, we compare classification performance by using AUC-PR, which copes with unbalanced data.

We focus on better feature design guided by the intuition that accidents affect regular traffic up to a certain distance from their location. We apply the proposed framework to historical accident data to understand basic conditions for quicker detection of future accidents. We hope that our research will inspire more efforts to leverage this novel understanding of the effect that accidents have on traffic, thereby enabling quicker and more accurate accident detection. To enable others to reproduce our findings, we provide access to the datasets (Berres et al., 2023), which are described in our Data in Brief paper.

2. Related work

There are two broad categories of AID methods differentiated by the type of models they use to represent traffic variables: (1) time series analysis (TSA) and (2) ML. On the one hand, TSA-based AID focuses on profiling the historic behavior of traffic variables to forecast their future values (Ahmed & Cook, 1982; Wang, Li, Liao, & Hua, 2013). TSA methods raise an alarm to represent an anomaly when the expected/predicted values are significantly different than the actual values. In general, TSA methods focus on identifying anomalous traffic changes, which means that incidents are usually identified as anomalies in the traffic stream. On the other hand, ML-based AID frames the problem of detecting incidents as a binary classification problem in which each data sample is a feature vector derived from traffic variables (e.g., speed, volume, occupancy). This is done for each segment of the road at a particular time (i.e., each road segment is labeled as having an incident or not). In general, ML methods are flexible and can find complex patterns of traffic variables encoded in a representative dataset that contains incidents. Our proposed framework employs ML.

2.1. Prior work closely related to the present study

Parsa et al. (2019) compared support vector machine (SVM) and probabilistic neural network (PNN) classifiers to detect accidents by using weather conditions, accident data, and loop detector data. They found that the PNN tended to outperform the classification performance

of the SVM. Specifically, the PNN had the best performance when using training data from 4 min before to 5 min after accidents occur. They used data from the Eisenhower expressway in Chicago, IL to compare the classifiers.

Shang et al. (2020) proposed a hybrid method that uses a random forest-recursive feature elimination algorithm and a long-short term memory (LSTM) network optimized by Bayesian optimization for incident detection. They focused on using traffic variables and their combinations. They tested and compared their approach with other state-of-the-art ML classifiers, including SVM, and showed that their approach outperforms them in multiple evaluation criteria. They tested their method with data from the I-880 freeway in California.

Parsa et al. (2020) used an XGBoost classifier to detect accidents by using traffic, network, demographic, land-use, and weather features. Additionally, they used SHAP to analyze the importance of each feature. They found that that the most important feature was the speed difference at the upstream location before and after accidents. They tested their framework with data from Chicago, IL's metropolitan expressways.

Compared to the studies mentioned above, the present paper is unique because it combines spatiotemporal features of traffic with weather and lighting conditions to find the best sensor configuration that minimizes the TTDA while maximizing classification results. Building upon this idea, we compare the classification performance of three distinct ML classifiers (logistic regression, random forest, and XGBoost) and show how additional neighboring sensor measurements (both upstream and downstream of the accident) help reduce the TTDA. This is unlike previous work (Parsa et al., 2020, 2019; Shang et al., 2020) that focused only on the temporal dimension.

We also use finer TTDA resolution (i.e., 30 s) to train and test the classifiers. This allows for finer control and helps us better understand the temporal conditions that minimize the TTDA as much as 1.0 min while maximizing classification performance as much as 83% in AUC-ROC and up to 49% in AUC-PR. In addition, in contrast with the previous, closely related work, we detail and contrast results in two different traffic and accident datasets, including two major highways in the metropolitan area surrounding Chattanooga, TN (i.e., I-75 and I-24). We share these datasets (Berres et al., 2023) with the community in a separate Data in Brief paper to ensure the reproducibility of our results.

2.2. Other prior work related to the present study

Cook and Cleveland (1974) used double exponential smoothing (DES) to forecast traffic-related signals and report incidents when there are significant deviations from the actual values. They used data from the lodge freeway in Detroit, MI. Cheu and Ritchie (1995) compared different neural network (NN) architectures to classify lane-blocking freeway incidents by using simulated and field data from the SR-91 riverside freeway in Orange County, CA.

Dia and Rose (1997) used a multilayer feedforward (MLF) NN trained and tested offline in a dataset from Tullamarine freeway in Melbourne, Australia. They also investigated the model's fault tolerance under corrupt and missing data conditions.

Abdulhai and Ritchie (1999) introduced a Bayesian-based probabilistic NN framework and explored its transferability without the need for explicit offline retraining in the new location. They showed that the performance of their method competes with the MLF while being computationally faster in training. They tested their approach on a large set of simulated incidents and real incident datasets from the I-880 freeway in California and the I-35 W in Minnesota.

Srinivasan, Cheu, Poh, and Ng (2000) proposed a hybrid, fuzzy-logic genetic algorithm technique that required less time to detect incidents and provided higher detection rates than the MLF. They tested their approach along the SR-91 riverside freeway in Orange County, CA. Jin, Cheu, and Srinivasan (2002) proposed a constructive probabilistic neural network (CPNN) architecture for incident detection. Their model is based on a mixture Gaussian model and trained by a dynamic decay adjustment algorithm. Their model was trained and tested on a simulated incident detection database from Singapore. They tested the transferability of the CPNN on the I-880 freeway in California.

Yuan and Cheu (2003) introduced and applied support vector machine (SVM) classifiers for arterial roads and freeways. Overall, SVM performed at least as well as the MLF when using data from the I-880 freeway in California. Teng and Qi (2003) used cumulative sum for change-point detection in traffic-related data. They tested their approach on data from the I-880 freeway in California. Tang and Gao (2005) proposed an improved non-parametric regression method to forecast traffic flow and report incidents based on the standard normal deviation (SND) rule. Their approach was validated by using traffic simulations. Srinivasan, Sharma, and Toh (2008) proposed a reduced multivariate polynomial-based NN to classify traffic conditions. Their model was trained and tested with real traffic data from the I-880 freeway in California. Chen, Wang, and Van Zuylen (2009) proposed an SVM ensemble to combine individual SVM classifiers based on certainty to outperform a single SVM. Their ensemble methods were tested on data from the I-880 freeway in California. Wang et al. (2013) proposed a hybrid approach that uses TSA and ML. Specifically, they used DES to model trends in the normal traffic and then to forecast normal traffic. An SVM was then used to distinguish between normal and incident-related traffic. This hybrid approach was tested with data from the I-880 freeway in California.

Chakraborty, Hegde, and Sharma (2019) extended the SND rule to account for robust univariate speed thresholds using historic traffic data. They de-noised these thresholds by using the spatiotemporal correlations of adjacent sensors and tested their approach with data from interstates I-80, I-35, and I-235 in Des Moines, IA. Kalair and Connaughton (2021) used kernel density estimates of raw density-flow data to define a contour of typical behavior in which normal traffic lies. They showed that deviations from this contour can be used to infer the presence of significantly anomalous behavior in the segments of a road associated with traffic incidents. They validated their approach to detect labeled incidents on London's M25 motorway. Taghipour, Parsa, Chauhan, Derrible, and Mohammadian (2022) proposed a deep ensemble framework to combine the prediction of deep learning techniques, including long-short term memory, gated recurrent unit, and deep learning networks, to detect accidents. They used traffic, accident, and weather condition data and found that multilayer perceptron and random forest classifiers perform best to create an ensemble of the output of the deep learning techniques. They tested their method by using data from Chicago's metropolitan expressways.

3. Dataset

We focus on the highway system in the Chattanooga, TN metropolitan area, including I-75, I-24, and US-27. The junction between interstates I-75 and I-24 is ranked 10th in the top 100 truck bottlenecks according to American Transportation Research Institute (ATRI) (2023). Additionally, the junction between I-24 and US-27 is ranked 29th.

In this section, we discuss the different data sources that constitute our ML inputs and the methodology we used to distill an input dataset. For more detailed information about our dataset, please refer to our Data in Brief document submitted alongside this manuscript, as well as the published dataset (Berres et al., 2023).

This study uses accident data, radar-based traffic data, and weather and lighting condition data from an observation period of 6 months from November 2020 to April 2021. Below, we describe the data sources and how they were used to synthesize a dataset to train the ML classifiers for accident detection.

An overview of the proposed framework is presented in Fig. 1. Our method is composed of three main phases: (1) dataset generation, (2)

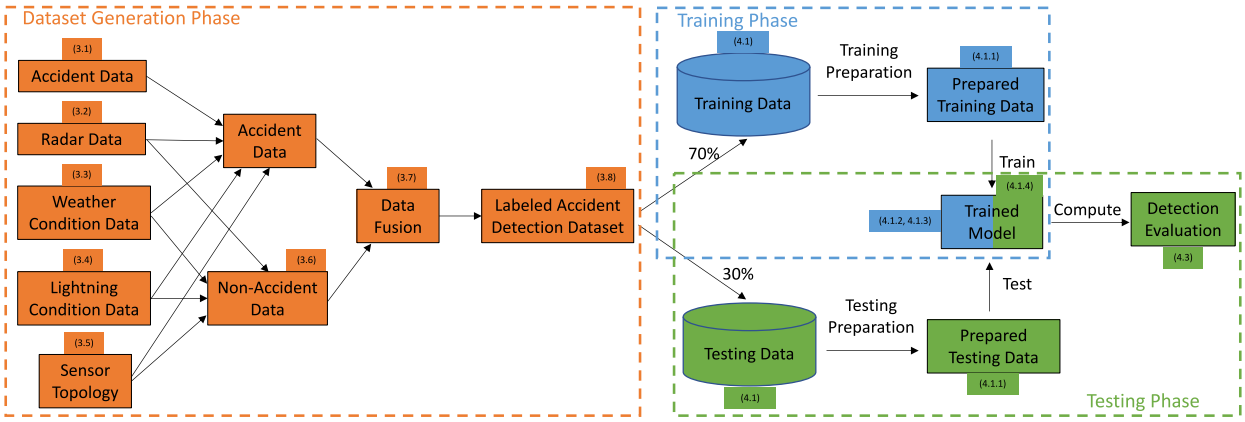

Fig. 1. Proposed AID pipeline.

training, and (3) testing. In the dataset generation phase, we build a labeled dataset by combining accident data (see Section 3.1), radar data (see Section 3.2), weather condition data (see Section 3.3), lighting condition data (see Section 3.4), and sensor topology (see Section 3.5). We combine these datasets to produce accident data samples based on their proximity to radar sensors and non-accident data samples (see Section 3.6). We finally fuse accident and non-accident data samples to come up with a dataset ready for ML classification (see Section 3.7). We detail the specifics of the dataset used in this study in Section 3.8. During the training phase, we process the training data (see Section 4.1) by conducting data preparation (see Section 4.1.1), model training (see Section 4.1.2), and selection (see Section 4.1.3) on a subset of classifiers (see Section 4.2). During the testing phase, we prepare the data and feed it into the trained model (see Section 4.1.4). Predictions from the classifiers are used to estimate the quality of the predictions (see Section 4.3). We associated each step in the proposed method (and their subsequent section numbers) with the corresponding phase in Fig. 1.

3.1. Accident data

We used accident data provided by the Tennessee Department of Transportation (TDOT) through the Enhanced Tennessee Roadway Information Management System (E-TRIMS) (Berres, Xu et al., 2021). This dataset contains information from 518,660 accidents that occurred throughout Tennessee from 2018 to 2021. We filtered this dataset down to a subset of accidents from the area of interest. First, we selected accidents along the main highways in the eight counties of the Chattanooga metropolitan area: Bledsoe, Bradley, Hamilton, Marion, McMinn, Meigs, Rhea, and Sesquatchie. This resulted in a total of 9644 accidents. Further specifying the time frame as November 2020 to April 2021 and focusing on I-75 (308 accidents) and I-24 (286 accidents) netted a total of 594 accidents. Fig. 2 shows the distribution of accidents (in each week day) during the observation period for I-75 (Fig. 2a) and I-24 (Fig. 2b).

E-TRIMS is an extract of the Tennessee Integrated Traffic Analysis Network (TITAN) dataset, which is a rich dataset published quarterly. Unfortunately, TITAN contains personally identifiable information. On the other hand, E-TRIMS is updated weekly and contains all the relevant data for accident detection without the privacy concerns inherent in the TITAN dataset. E-TRIMS is a rich tabular dataset with detailed information on each recorded accident. In the following, we list the information hat would be available in a real-time detection scenario, as the goal of this work is to detect accidents. Additional fields available in the data are discussed in the companion Data in Brief paper.

- Date and time the accident was reported
- Geographic location of each accident: county, route name, and geo-coordinates
- Type and severity of crash: property damage, suspected minor/major injury, fatality
- Lighting and weather conditions: These will be discussed in more detail in Sections 3.3 and 3.4

3.2. Radar data

The Tennessee highway system has TDOT-maintained and operated radar detectors placed at intervals of roughly 1 mile along the major highways in each of Tennessee's four metropolitan areas, as Fig. 3.

The Tennessee highway system has TDOT-maintained and operated radar detectors placed at intervals of roughly 1 mile along the major highways in each of Tennessee's four metropolitan areas, as Fig. 3. These radar detectors emit low-energy microwave radiation that is reflected by the vehicles (Xu et al., 2022) and can be captured by the sensor at lane-by-lane resolution. For each lane, the sensors capture the number of vehicles passing by, their speeds, and the lane occupancy, with each aggregated to 30-s intervals. The specific radar sensors used in this study are SmartSensor V by Wavetronix. These radar sensors provides true eight-lane detection of vehicle speed, volume, and occupancy. These sensors are an important component of the SmartWay highway traffic information system enabling situational awareness and travel time estimation (TDOT, 2022). Their typical accuracy per direction for speed, volume, and occupancy is in the range of ±5 mph, 96%–98%, and ±10% respectively.[1]

We hypothesize that when an accident occurs, it will affect measurements of neighboring radar detectors. Fig. 4 illustrates the impact that we observed in traffic and how we hypothesize it translates to sensor measurements. We consider data from sensors upstream and downstream of the accident location in addition to the data from the nearest sensor.

For I-75, we focused on 41 radar detectors on a northbound stretch and 46 radar detectors on the southbound stretch. These sensors divide up this I-75 stretch into 40 northbound segments (average length of 0.9 miles) and 45 southbound segments (average length of 0.8 miles). For I-24, we focused on 28 radar detectors on an eastbound stretch and 25 radar detectors on the westbound stretch. These sensors divide up this I-24 stretch into 27 eastbound segments (average length of 0.5 miles) and 24 westbound segments (average length of 0.5 miles). Fig. 3 shows the locations of all radar detectors in this region, including the sensors placed along I-75 and I-24.

[1] The full list of technical specifications of the radar sensors used in this study can be found at: http://www.signalcontrol.com/products/wavetronix/Wavetronix_SmartSensor_V.pdf.

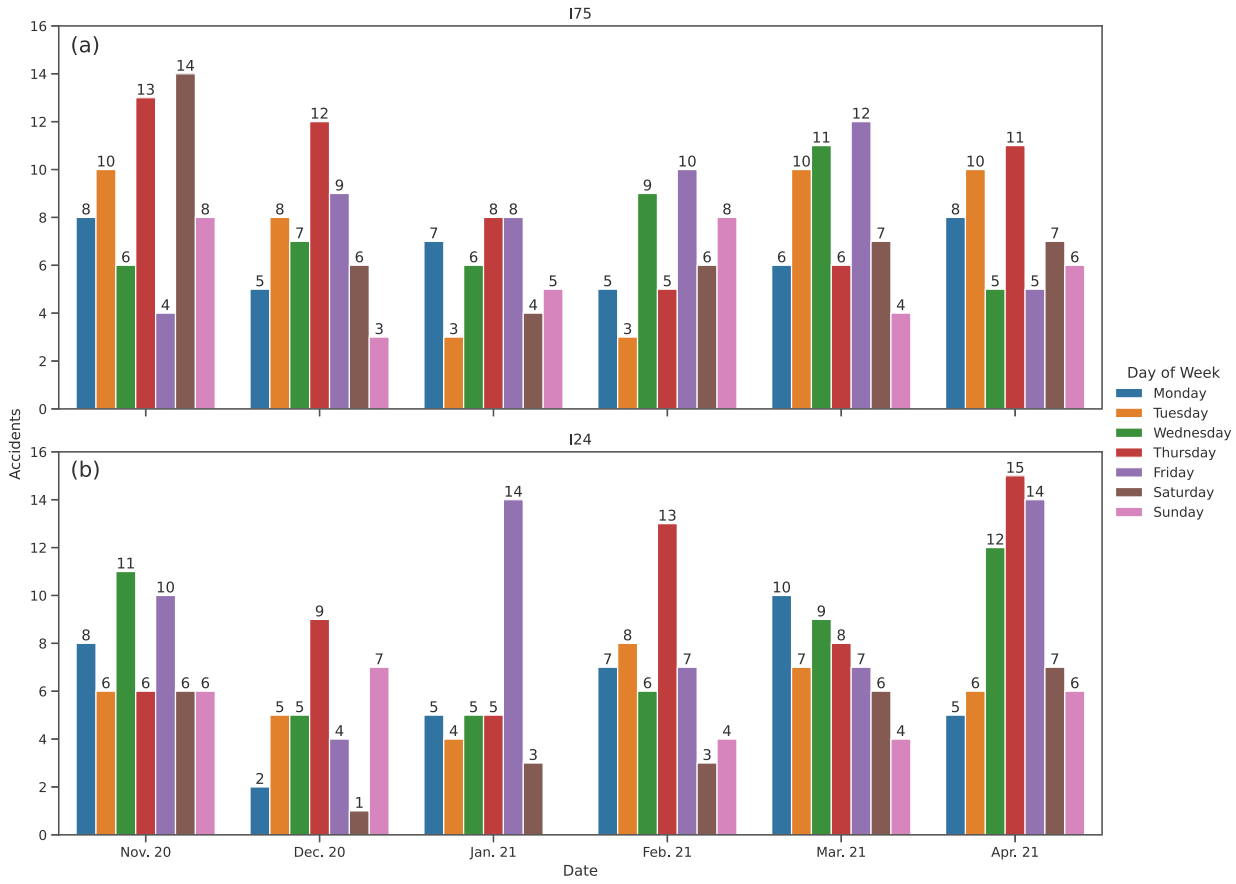

Fig. 2. Number of accidents (in both directions) during the observation period. (a) 302 accidents on I-75 and (b) 280 accidents on I-24.

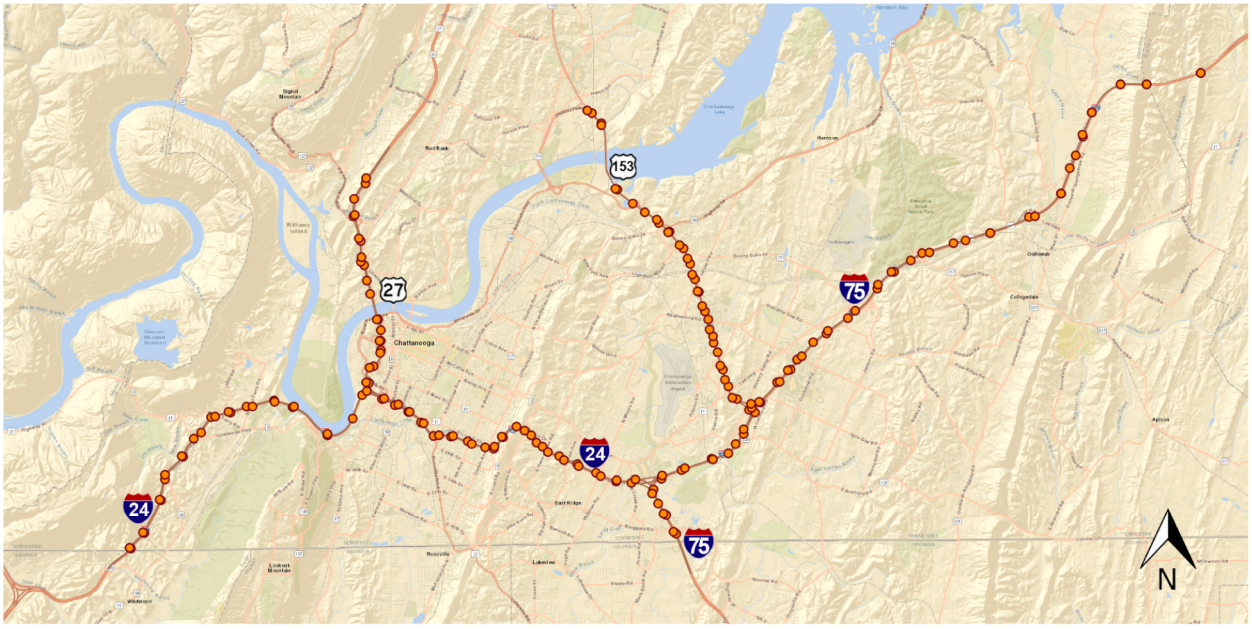

Fig. 3. Radar detectors in the Chattanooga metropolitan area stretch along I-75, I-24, SR-153, and US-27. They cover areas with heavy traffic on each roadway all the way to the Georgia state line.

3.3. Weather condition data

Weather has an impact on driving behavior and traffic safety, as reported by the US DOT Federal Highway Association (FHWA) (2023a). E-TRIMS provides weather condition data as one of its variables. This gives us information on conditions when accidents occurred, but it does not tell us what weather conditions were like when there was no accident. To obtain this information, we used meteorological data to augment the feature space and aid in accident classification. Specifically, we used data from the Prediction Of Worldwide Energy Resources

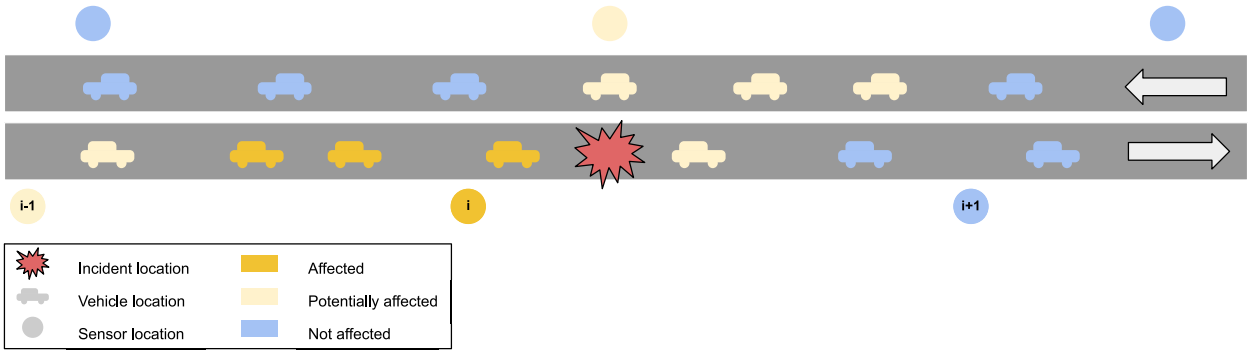

Fig. 4. Schematic of traffic impact from an accident, which is marked by a spiky red symbol. Sensor placement is illustrated with circles next to the road, and vehicle data is represented by small vehicle symbols. The extent of traffic impact is visualized by using dark yellow for strong impact, light yellow for some impact, and light blue for little to no impact. In the illustration, nearby vehicles upstream from the accident are affected most. Upstream vehicles that are farther away and have just passed the accident and sometimes nearby vehicles in the opposing traffic direction may experience some impact (e.g., slowing down or speeding up). Vehicles even farther away continue at their usual travel speed.

(POWER) project funded by the National Aeronautics and Space Administration (NASA) to supplement weather information for non-accident training data (NASA, 2023). For the sake of consistency, we decided to use this data as a source of weather conditions for both accident input data and non-accident input data.

The original E-TRIMS weather condition information has the following categories: clear, cloudy, rain, fog, sleet/hail, snow, blowing snow, severe crosswinds, blowing sand/soil/dirt, smog/smoke, other, and unknown.

As a first step, we simplified these categories to rain, snow (snow, sleet/hail, blowing snow), wind (severe crosswinds, blowing sand/soil/dirt), and unknown (clear, other, unknown). When multiple categories were possible (e.g., blowing snow could be wind or snow in this scenario), we chose the category with a bigger traffic impact based on FHWA's report.

We then used POWER data to reproduce these categories from the hourly measurements. We used the following variables from this dataset:

- Temperature (C): Average air (dry bulb) temperature at 2 m above the surface
- Precipitation (mm/h): Average of total precipitation at the surface, including water content in snow
- Wind Speed (m/s) at 2 m: Wind speed at 2 m above the surface

We used these basic measurements to synthesize weather conditions comparable to the E-TRIMS weather conditions.

First, we determined precipitation. According to the United States Geological Survey (USGS) (2023), a "heavy (thick) drizzle" is defined as 1 mm of precipitation per hour, and it can impair visibility. We therefore used 1 mm as our threshold for a precipitation classification and we used temperature to distinguish between snow and rain. Next, we used the definition from the National Weather Service (NWS) (2023) for wind advisories to determine when to use the wind classification. Finally, if the category was neither rain, snow, nor wind, we set it to – (unknown).

3.4. Lighting condition data

Lighting has a big impact on traffic accidents. Only 25% of travel occurs after dark; however, about 50% of all fatalities occur at night. Although drowsy driving and intoxication account for some of these accidents, decreased visibility can also contribute to this problem because drivers are more likely to have an accident if they cannot see a hazard (US DOT Federal Highway Association (FHWA), 2023b). Studies have shown that drivers often do not adjust their speed to lighting conditions (Jägerbrand & Sjöbergh, 2016). In addition to full darkness, dusk and dawn are also hazardous due to the glare from sunrise and sunset, and the sharp contrast between bright sky and dark roads and environment can make it difficult for drivers to see.

The original E-TRIMS lighting condition data contains the categories daylight, dusk, dawn, dark (lighted, not lighted, unknown lighting), other, and unknown. Applying the same reasoning we used for weather data, we augmented the feature space with solar data to aid in accident classification and to have consistent data for both accidents and non-accidents. We chose the sunrise-and-sunset (Sunrise Sunset, 2023) dataset as our external data source for light. We used the times for civil twilight start, sunrise, sunset, and civil twilight end to aggregate the lighting conditions to dawn, daylight, dusk, and dark.

We did not consider artificial lighting in this study because information on lighting along highways is not always available. Furthermore, artificial lighting along the highway remains the same throughout the study's time frame; therefore, the presence or absence of artificial lighting along the highways should not affect the results of our ML classifiers. As we will discuss in Section 3.8, the non-accident data we produce mirrors the times and locations of the accident data.

3.5. Sensor topology

Along each highway, radar detectors are placed at intervals between 0.5 to 1.0 mile for each travel direction. We split the highway into segments at the mid-point between each pair of adjacent sensors such that each sensor corresponds to the highway segment that is closer to this sensor than any other sensor. In areas with highway junctions, we cross-reference the sensor name (which contains the name of the highway it relates to) to ensure that close proximity does not result in incorrect assignments. Fig. 5 demonstrates the necessity of such a step.

In this step, we produce two datasets which are needed for further processing. First of all, we create a topological representation of the network of sensors, which stores each sensors upstream and downstream neighbors. Second, we produce a geometry file consisting of polygons which allow us to determine the nearest sensor for any given accident location. A more detailed description of these datasets can be found in the companion Data in Brief paper.

3.6. Synthesizing non-accident data

The ML algorithm needs examples of accidents and non-accidents. To synthesize non-accident data, we must first determine which sensors are affected by accidents throughout all times within the entire 6-month study. We saved this information as a matrix of *(Number of time steps)* × *(Number of sensors)* to enable fast lookup. We then checked the data to see which of the similar time frames (same location, same day of the week, same times) are free of accidents and created a list of non-accidents. Within the given time frame (27 weeks), we can have a maximum of 26 non-accidents for any given accident. This ensures that

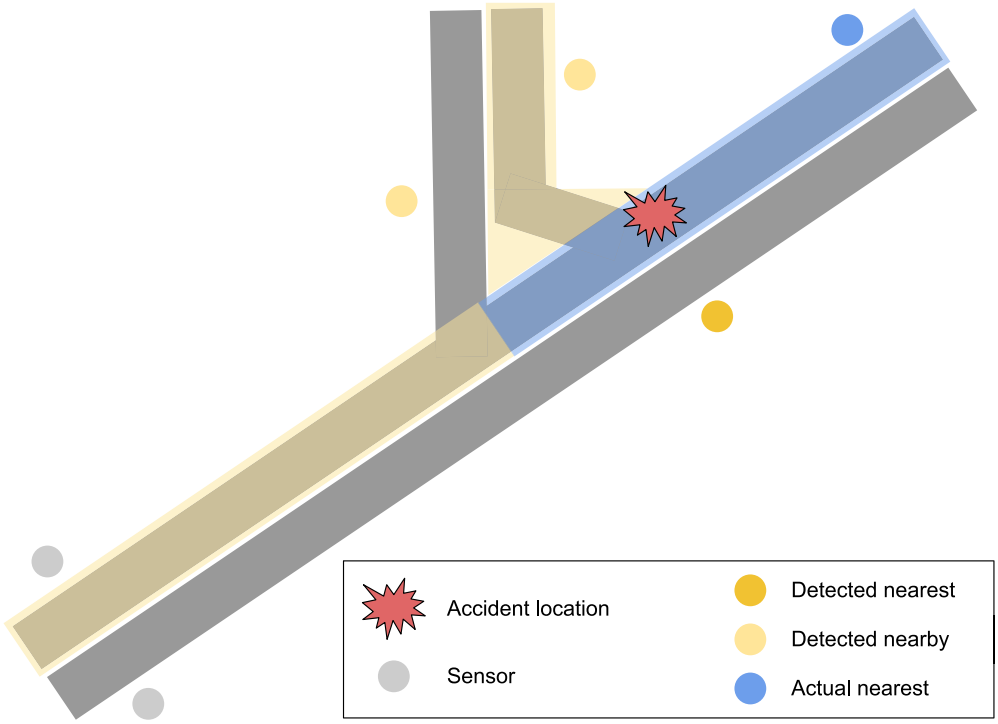

Fig. 5. Sketch of a highway junction, where one highway (vertical) branches off of another highway (diagonal). Road-side radar sensors are represented as circles, and their corresponding range is represented by a colorful overlay. An accident (marked in red) occurs near the junction point on the diagonal highway. Based on the range assigned to each sensor, it should be assigned to the blue sensor. A purely proximity-based approach would assign a sensor that overlooks the opposite traffic direction (dark yellow). Other nearby sensors which are closer than the blue sensor are shown in light yellow. The potentially affected road segments are highlighted in blue and yellow respectively.

we account for factors that are specific to a location (e.g., speed limits, artificial lighting), day of the week (e.g., different traffic patterns on weekdays vs. weekends), and time of day (e.g., rush hour). We mirror all properties for the accident data with the exception of the type, which we set to *None*.

Moving forward, we refer to data that can be either an accident or a non-accident as an *event*.

3.7. Data fusion

The goal of this work is to train an ML algorithm to detect accidents. The input for this ML algorithm is fused data, which combines event data with traffic data. For each event, we produced dedicated machine learning input by fusing the different data sources into a single tabular format with the following columns:

- Time steps (rows): We begin data collection 15 min prior to the accident time and end 15 min after the event time.
- Event data (columns):

1. A boolean signifies whether the event is an accident (i.e., 1) or a non-accident (i.e., 0).
2. The road that the original accident occurred on and the mile marker it occurred at (e.g., *00I75S* and *4.8* if the original accident occurred near mile marker 4.8 on I-75 southbound).
3. The type of event (e.g., *Prop Damage [over]*) for an accident with property damage over a predefined threshold. If the event is a non-accident, we set this column to *None*.
4. The event's date, time (e.g., *18:35*) and hour (e.g., *18*). These columns have the date/time/hour of the accident recorded in E-TRIMS, and they remain the same value for the entire file.
5. The sensor data's time. This column contains the timestamp of the sensor data contained in each row.
6. Triplets of *speed(i)*, *volume(i)*, and *occupancy(i)* for each sensor from 5 sensors upstream to 5 sensors downstream (e.g., *speed(i-5), volume(i-5), occupancy(i-5) ..., speed(i), volume(i), occupancy(i), ..., speed(i+5), volume(i+5), occupancy(i+5)*) from the data.
7. Weather and lighting conditions (e.g., *Rain* and *Dusk*).

3.8. Final dataset

To train the ML classifiers used in this research, we randomly selected non-accident samples during the same observation period. Specifically, we selected non-accident events by looking at the day and time of accident events and randomly sampling 24 non-accident events per accident with similar date, time, and day of the week given the strong temporal component of traffic data. After filtering missing data caused by gaps in radar data (e.g., missing windows of several hours), the full dataset consists of 302 accidents and 7455 non-accident events on I-75 and 280 accidents and 6916 non-accident events on I-24.

We used a subset of this data that focuses on accidents in which the nearest radar detector could be one of up to five available upstream or downstream sensors. This subset also has an accident type of either *suspected minor/major injury* or *fatality* because they have greater impact on traffic conditions. With those parameters, the dataset consists of 24 accidents and 3039 non-accident events on I-75 and 31 accidents and 4852 non-accident events on I-24.

The dataset we share with the community in our Data in Brief submission includes all accident and non-accident data, including property damage (Berres et al., 2023).

Table 1 details the set of explanatory variables used in this research, including their description. They include traffic data (i.e., speed, volume, and occupancy) and environmental data (i.e., weather and light). We focused on using traffic variables from neighboring radar detectors instead of leveraging predictions based on historical data. Specifically, we used up to five neighboring sensors in upstream and downstream

Table 1
Explanatory variables used in this research along with their description. n differs from 0 to 7 min in 30-s intervals.

Variable	Description
Traffic data	
Speed	Speed of up to five neighboring radar detectors in upstream and downstream directions.
	We used speed measurements from 4 min before up to n min after an accident/non-accident.
Volume	Volume of up to five neighboring radar detectors in upstream and downstream directions.
	We used volume measurements from 4 min before up to n min after an accident/non-accident.
Occupancy	Occupancy of up to five neighboring radar detectors in upstream and downstream directions.
	We used occupancy measurements from 4 min before up to n min after an accident/non-accident.
Environmental data	
Weather	One hot encoded representation of clear, cloudy, and rain/snow weather conditions.
Light	One hot encoded representation of daylight, dark, and dawn light conditions.

directions on each roadway. We also trained our classifiers to detect accidents by using different TTDA from 0 to 7 min after the occurrence of accidents.

4. Methods

This section describes the mathematical foundation of the algorithms used to perform this research.

4.1. Study setup

We trained each classifier on 70% of the data and tested on the remaining 30%. The testing data is only used once for computing the performance of the classification task. We implemented the proposed framework with the scikit-learn (Pedregosa et al., 2011), imbalanced-learn (Lemaître, Nogueira, & Aridas, 2017), and SHAP (Lundberg & Lee, 2017) APIs. As detailed below, after the train/test split, we prepare the data (Section 4.1.1), build the model (Section 4.1.2), optimize the model (Section 4.1.3), and then finally apply the optimized accident models to the test dataset (Section 4.1.4).

4.1.1. Data preparation

We first standardized the independent variables. Given that our dataset is highly imbalanced—24 accidents vs. 3039 non-accident events (≈0.8%) on I-75 and 31 accidents vs. 4852 non-accident events on I-24 (≈0.6%), we over-sampled the minority class because this tends to perform better than under-sampling in severely imbalanced datasets (García, Sánchez, & Mollineda, 2012). Specifically, we used SMOTE to handle the imbalance (Chawla et al., 2002). SMOTE processes each sample in the minority class to generate new synthetic samples by joining them to their k-nearest neighbors. We used regular SMOTE with $k = 5$ because of its simplicity and high performance (Parsa et al., 2019).

4.1.2. Build model

We trained classification models by using the labeled training dataset. We describe each of the classifiers we used in Section 4.2.

4.1.3. Optimize model

To achieve optimal performance, we tuned each classifier to find the optimal set of hyperparameters that maximizes the AUC-ROC. We used grid search to test a small combination of parameters with reasonable values and performed a 5-fold cross validation with the training data (see Section 4.2). The training data was randomly split into five subsamples, and four of these subsamples were used for training, and the remaining subsample was withheld for validation. This procedure was repeated five times until each subsample was used once for validation purposes. This allowed us to measure the classifier performance consistently across the entire training dataset.

4.1.4. Apply model

The optimized accident classification models are applied to the testing dataset. For each traffic sample in the testing dataset, the proposed classification model predicts whether the traffic sample is likely to represent an accident and then outputs a binary label.

4.2. Classification models

We used three different classifiers that have shown promising results in AID. Specifically, we include (1) logistic regression (a regression-based classifier) (Kitali, Alluri, Sando, & Wu, 2019); (2) random forest (an ML-based classifier) (Ozbayoglu, Kucukayan, & Dogdu, 2016); and extreme gradient boosting (XGBoost) (an ML-based classifier) (Parsa et al., 2020). Here, we assume that for each road segment in a given direction, accident detection can be viewed as a binary classification problem. Without loss of generality, suppose that the accident data has n samples $(\boldsymbol{x}_i, y_i), i \in 1, \dots, n$, where $\boldsymbol{x}_i = (x_{i1}, x_{i2}, \dots, x_{id})$ contains d explanatory variables or features (as described in Table 1), and y_i is a dependent variable that represents an accident indicator (i.e., where 1 means that an accident is caused by the explanatory variables, and 0 means that no accident occurred). We detail each of the classifiers we used below: logistic regression (Section 4.2.1), random forest (Section 4.2.2), and XGBoost (Section 4.2.3).

4.2.1. Logistic regression

Logistic regression is an extension of linear regression for classification problems with binary outcomes (Hosmer, Lemeshow, & Sturdivant, 2013). Logistic regression represents class-conditional probabilities by using a linear combination of the explanatory variables as below:

$$\log\left(\frac{Pr(y_i = 1)}{Pr(y_i = 0)}\right) = \beta_0 + \beta_1 x_{i1} + \cdots + \beta_d x_{id} = \beta_0 + \boldsymbol{x}_i^{\top}\boldsymbol{\beta},$$

where $\boldsymbol{\beta} = \beta_1 + \beta_2 + \cdots + \beta_d$ is a vector of coefficients to be estimated, and $Pr(y_i = 1)$ and $Pr(y_i = 0)$ are the probabilities of class labels 1 and 0, respectively. We performed a grid search over the inverse of the regularization strength parameter: $C \in [0.01, 0.1, 1.0, 10, 100]$, and we found that the optimal value is 100.

4.2.2. Random forest

Random forest is an ensemble method based on bagging (or bootstrap aggregation) that trains a B number of decision trees from subsets of the original dataset with the same size sampled with replacement (Breiman, 2001). Thus, each of these trees focuses on a random subset of features. To make a prediction on a new sample, $\boldsymbol{x}_i$, let $\hat{C}_b(\boldsymbol{x}_i)$ be the class prediction of the bth random-forest tree. The random forest then aggregates each of the results from the trees to make a final decision by using the majority vote method as

$$\hat{C}_{\mathrm{rf}}^{B}(\boldsymbol{x}_i) = \text{majority vote } \{\hat{C}_b(\boldsymbol{x}_i)\}_1^B.$$

We performed a grid search of trees in the forest parameter: `n_estimators` $\in [10, 100, 1000]$, and we found that the optimal value is 100.

4.2.3. XGBoost

XGBoost is an implementation of gradient-boosted decision trees designed for speed and performance (Chen & Guestrin, 2016). Boosting is an ensemble method in which K models are added iteratively to predict a dependent variable:

$$\hat{y}_i = \sum_{k=1}^{K} f_k(\boldsymbol{x}_i), f_k \in \mathcal{F},$$

where f_k is an independent tree structure with a continuous score in each leaf, and $\mathcal{F}$ is the space of trees. We performed a grid search of the learning rate parameter, `learning_rate` $\in [0.001, 0.01, 0.1]$, and of the number of trees in the forest parameter, `n_estimators` $\in [10, 100, 1000]$. We found that the optimal set of values is 0.01 for the learning rate and 1000 for the number of trees.

4.3. Detection evaluation

The comparison of different classifiers and conditions is based on counting the number of samples that are labeled as follows. True positive (TP) indicates that an accident instance is correctly detected. False positive (FP) indicates that a non-accident instance is incorrectly detected as an accident. False negative (FN) indicates that an accident instance is missed. True negative (TN) indicates that a non-accident instance is correctly classified as non-accident.

We used these classifiers to compute widely accepted metrics for accident detection. Detection rate (DR), or recall, indicates the actual proportion of accidents that have been detected. False alarm rate (FAR), or FP rate, indicates the proportion of non-accidents detected over the total number of non-accidents. The AUC-ROC indicates the overall performance of a classifier based on the variation of DR with respect to FAR at various thresholds. Due to the overly optimistic view provided by AUC-ROC estimates in highly imbalanced scenarios, such as accident detection, we also report the AUC-PR, or average precision, to indicate the overall performance of a classifier based on the variation of correctly identified accidents out of the total (precision) with respect to recall (or DR) at various thresholds. AUC-PR changes with the ratio of positive and negative instances capturing the susceptibility of classifiers to imbalanced datasets, thereby placing more importance on the detection of the minority class (accidents) (Davis & Goadrich, 2006; Saito & Rehmsmeier, 2015). Because of that, we used AUC-PR as a single metric to compare classifier performance. Notably, the baseline of the AUC-PR is given by the proportion of accidents: 0.8% for I-75 and 0.6% for I-24 (Section 4.1.1). We did not report accuracy to indicate the proportion of correctly predicted accident and non-accident instances, given the imbalanced nature of the dataset. We used the following definitions for each of the above metrics:

$$\text{DR} = \frac{\text{Number of true accident reports}}{\text{Total number of accidents}} = \frac{TP}{TP + FN}$$

$$\text{FAR} = \frac{\text{Number of false accident reports}}{\text{Total number of accidents}} = \frac{FP}{TN + FP}$$

$$\text{Precision} = \frac{\text{Number of true accident reports}}{\text{Total number of predicted accidents}} = \frac{TP}{TP + FP}$$

4.4. Feature importance analysis

Feature importance refers to computing a score for all predictors of a given classifier. Scores quantify the importance of each feature for the prediction task. Thus, the higher the score, the larger the effect that a particular feature has on a classifier that is used to predict a certain variable. In this study, we used SHAP to estimate feature importance (Lundberg & Lee, 2017). SHAP is a game theoretic approach used to explain the output of any ML classifier. SHAP focuses on connecting optimal credit allocations with local explanations (Ribeiro, Singh, & Guestrin, 2016) through Shapley values from game theory (Štrumbelj & Kononenko, 2014). In SHAP, feature values of a data sample act as players, and Shapley values indicate how to fairly distribute predictions among features. Thus, the contribution, $\phi_i \in \mathbb{R}$, of an individual feature, i, is based on their marginal contribution (Nowak & Radzik, 1994). It specifies the explanation through a linear function of binary functions, g, defined by:

$$g(z') = \phi_0 + \sum_{j=1}^{M} \phi_j z'_j,$$

where $z' \in \{0,1\}^M$ is the coalition vector (i.e., equals 1 when a feature is observed and 0 otherwise), and M is the number of input features.

In this study, we compute feature importance for the best performing classifier (XGBoost) and focus on AUC-PR. Note that this procedure does not reflect the intrinsic predictive value of the features themselves but rather how important the features are for a particular classifier. This means that the most important features may differ depending on which classifier is used. Other methods to compute feature importance include mean decrease impurity (MDI) (Louppe, 2014) and permutation feature importance (Breiman, 2001). We did not use these two methods because MDI tends to be strongly biased toward high cardinality features (Strobl, Boulesteix, Zeileis, & Hothorn, 2007), and because permutation feature importance may produce misleading values on strongly correlated features (Nicodemus, Malley, Strobl, & Ziegler, 2010).

5. Results

In this section, we present our results of the impact of neighboring measurements on accident detection. We conduct two different but complementary analyses.

First, we perform a spatiotemporal sensibility analysis on the impact that neighboring sensor measurements and TTDA have on the accident detection task (Section 5.1). We do this for each classifier considered in this research: logistic regression, random forest, and XGBoost. Specifically, we compare classification results by computing the effect that neighboring sensors have under two settings. In Setting 1, we use up to five neighboring sensors located upstream from the accident. In Setting 2, we use symmetric sensors located upstream *and* downstream from the accident location (up to five sensors in each direction). We hypothesize that accidents significantly affect upstream and downstream traffic up to a certain distance, so using these features helps to design better classifiers. To study the effect of the number of sensors and their settings, we trained classifiers and report results on the test data by controlling the influence of neighboring sensors placed at incremental distances from accidents. In each of these settings, we also computed results when using different TTDAs, from 0 min to 7 min at 30 s intervals.

We found an optimal combination of neighboring sensor arrangements and TTDA for achieving the best classification results (using the AUC-PR metric). Specifically, the classification performance increases as we consider more distant sensors, but it starts reaching a point of diminishing results. Adding more sensors (upstream and downstream for Setting 2) produces marginal improvements at 4–5 hops. In addition, along with a specific neighboring sensor configuration, TTDA matters. In particular, we found that using traffic data up to 1 min after accidents produces reasonably good results for the best performing classifier in the richer dataset (i.e., I-24). In our analysis, XGBoost classifiers produce the best results over logistic regression and random forest.

Second, we perform feature importance analysis to better understand the most influential features that drive the results, (i.e., how much each feature contributed to the prediction) (Section 5.2). We did this for the best performing classifier (XGBoost) under Setting 2 (i.e., up to five upstream and downstream sensors) at 1.5 min TTDA for I-75 and 1.0 min TTDA for I-24. The most influential features contributing to the predictions are traffic-related variables, (e.g., speed, volume, and occupancy) from locations farther from the accidents, including those

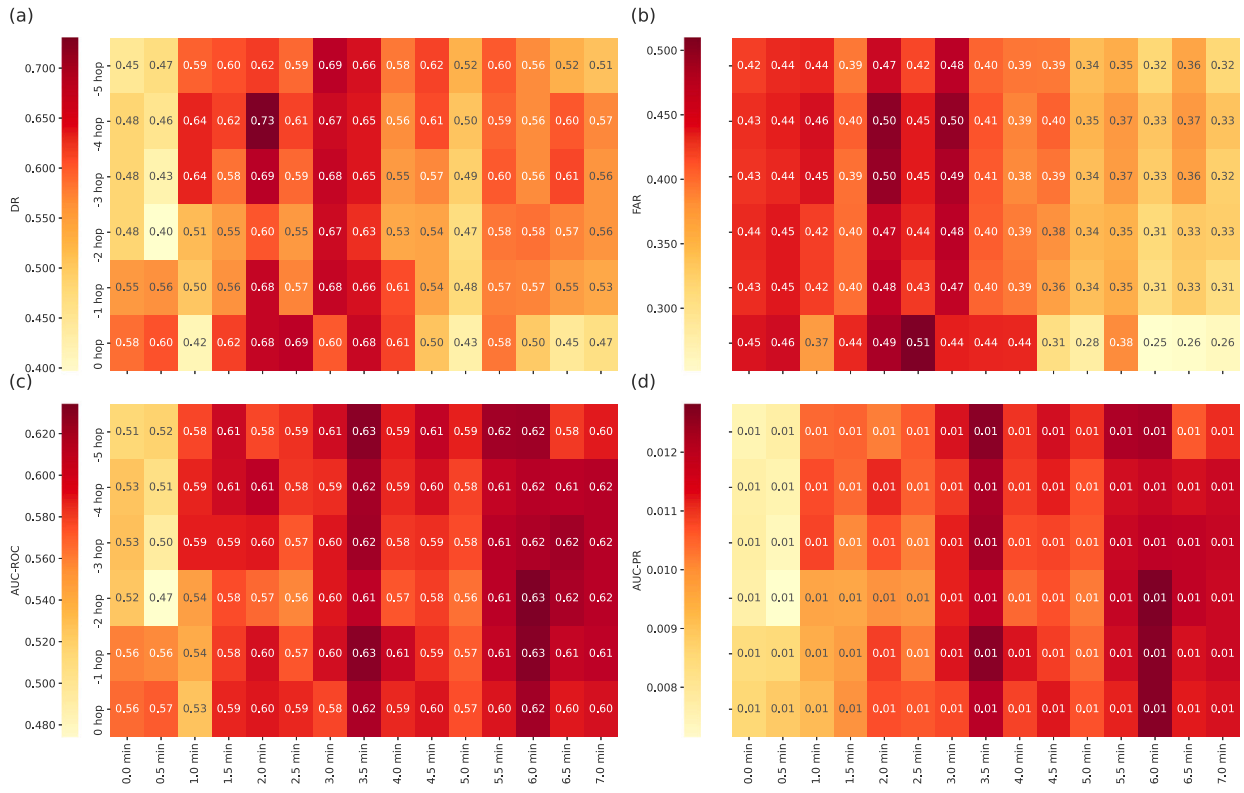

Fig. 6. The I-75 classification results with a logistic regression classifier under Setting 1: (a) is DR, (b) is FAR, (c) is AUC-ROC, and (d) is AUC-PR.

up to five sensors apart. This supports our hypothesis that adding further sensors improves classification results and reduces TTDA. Overall, features related to weather and lighting seem to be less important to the classification results. Although weather and lighting conditions can contribute to the occurrence of accidents, they are expected to factor in more significantly in general predictions of how likely accidents are to occur, and this can help better allocate local law enforcement and emergency response services (Roland, Way, Firat, Doan, & Sartipi, 2021). In contrast, our work focuses on detecting accidents that have already happened and are actively affecting traffic measurements.

5.1. Spatiotemporal sensitivity analysis

Figs. 6–17 visualize different classification metrics for DR (a), FAR (b), AUC-ROC (c), and AUC-PR (d), as defined in Section 4.3. Each cell of the heat maps depicts a single performance metric value for a specific combination of neighboring sensors (i.e., Setting 1 or Setting 2) and TTDA. The results are rounded at two decimal places. The horizontal axis represents the TTDA (from 0 to 7 min at 30 s intervals). The vertical axis represents the number of neighboring sensors included for training and testing the classifiers in each of the settings: using sensors upstream from an accident (Setting 1) and using sensors upstream *and* downstream from an accident (Setting 2). We organize the results by road (I-75 and I-24) and classification algorithm (logistic regression, random forest, and XGBoost).

5.1.1. I-75 analysis

Figs. 6 and 7 show the classification results for a logistic regression classifier under Settings 1 and 2, respectively. Note that two things happen. First, for Setting 1, there are no important differences in performance when accounting for more upstream sensors (vertical axis) and different TTDAs (horizontal axis) for DR, FAR, and AUC-ROC. Additionally, AUC-PR remains constant at 1%, which reflects no significant increase over the baseline of 0.8%. Second, for Setting 2, we observed no significant performance increase when considering both upstream and downstream sensors, and AUC-PR plateaus at around 1%. Interestingly, in Setting 2, FAR is lower by up to 16% (from 44% to 28%) than in Setting 1.

Figs. 8 and 9 show the performance results when using a random forest classifier under Settings 1 and 2, respectively. Note that under Setting 1, considering more upstream sensors improves classification results across all metrics. In particular, there is an increasing trend in DR, AUC-ROC, and AUC-PR and a decreasing trend in FAR. The highest AUC-PR is 13%, which is achieved at 7 min. Similarly, for Setting 2, considering more sensors (both upstream and downstream) helps improve detection metrics. Specifically, the AUC-PR can be as high as 33% and can be achieved at 7 min. Leveraging Setting 2 on a random forest classifier represents an increase of as much as 20% (from 13% to 33%) in AUC-PR over Setting 1. Remember that the baseline for AUC-PR in the I-75 dataset is 0.8%.

Figs. 10 and 11 show the performance of the XGBoost classifier under Settings 1 and 2, respectively. For Setting 1, the DR degrades as we start using more neighboring upstream sensors. This is the opposite of what we observe with logistic regression and random forest classifiers. The remaining metrics, including FAR, AUC-ROC, and AUC-PR, show a trend of performance improvement as we use data from more upstream sensors. Here, the 27% peak AUC-PR is achieved when using data up to 5.5 min. On the other hand, for Setting 2, we see an improvement in every performance metric. Here, the performance can be as good as 45% in AUC-PR (including 82% in AUC-ROC and almost negligible in FAR) and is achieved at 5.5 min. This peak in performance represents an increase of as much as 18% over the same metric in Setting 1.

5.1.2. I-24 analysis

The I-24 analysis is similar to the I-75 case study. Figs. 12 and 13 show performance results when using a logistic regression classifier under Settings 1 and 2, respectively. Under Setting 1, there is no particular advantage to using more upstream sensors in the classification task. In terms of AUC-PR, the metric plateaus at 1% regardless of how

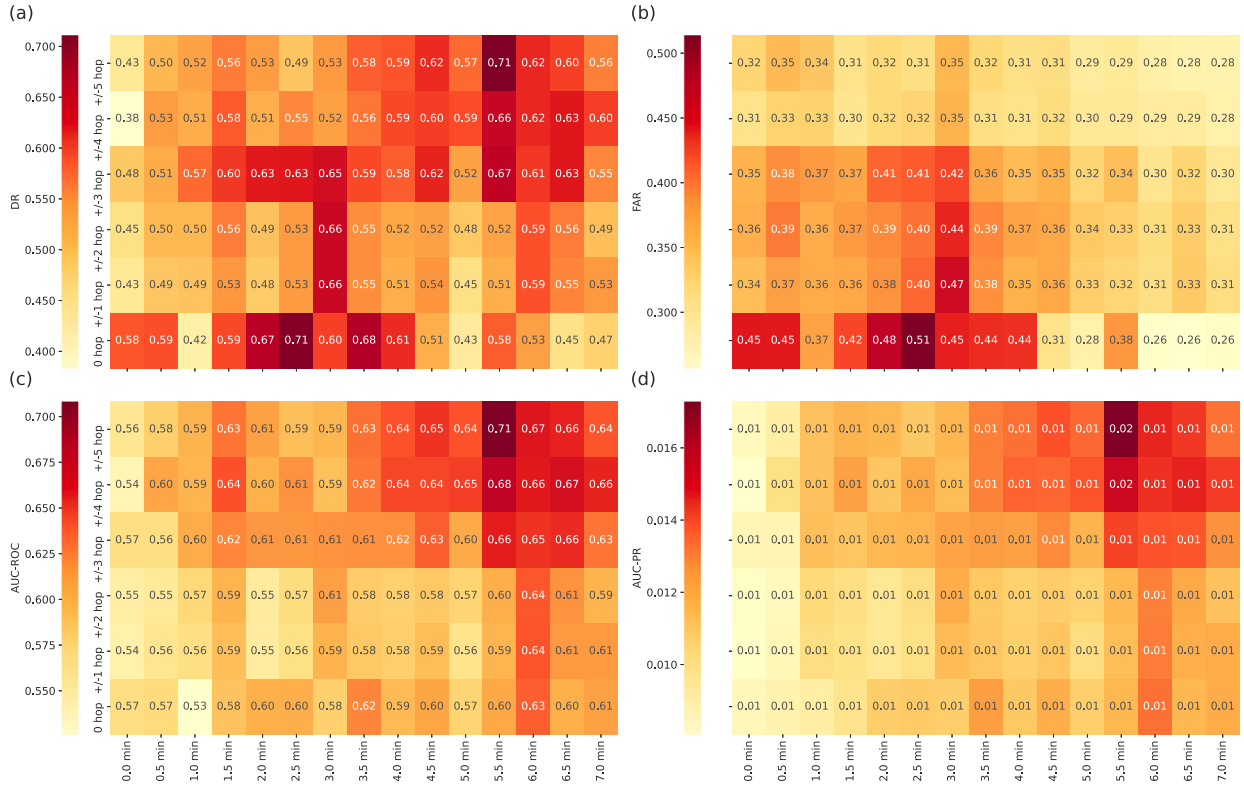

Fig. 7. The I-75 classification results with a logistic regression classifier under Setting 2.

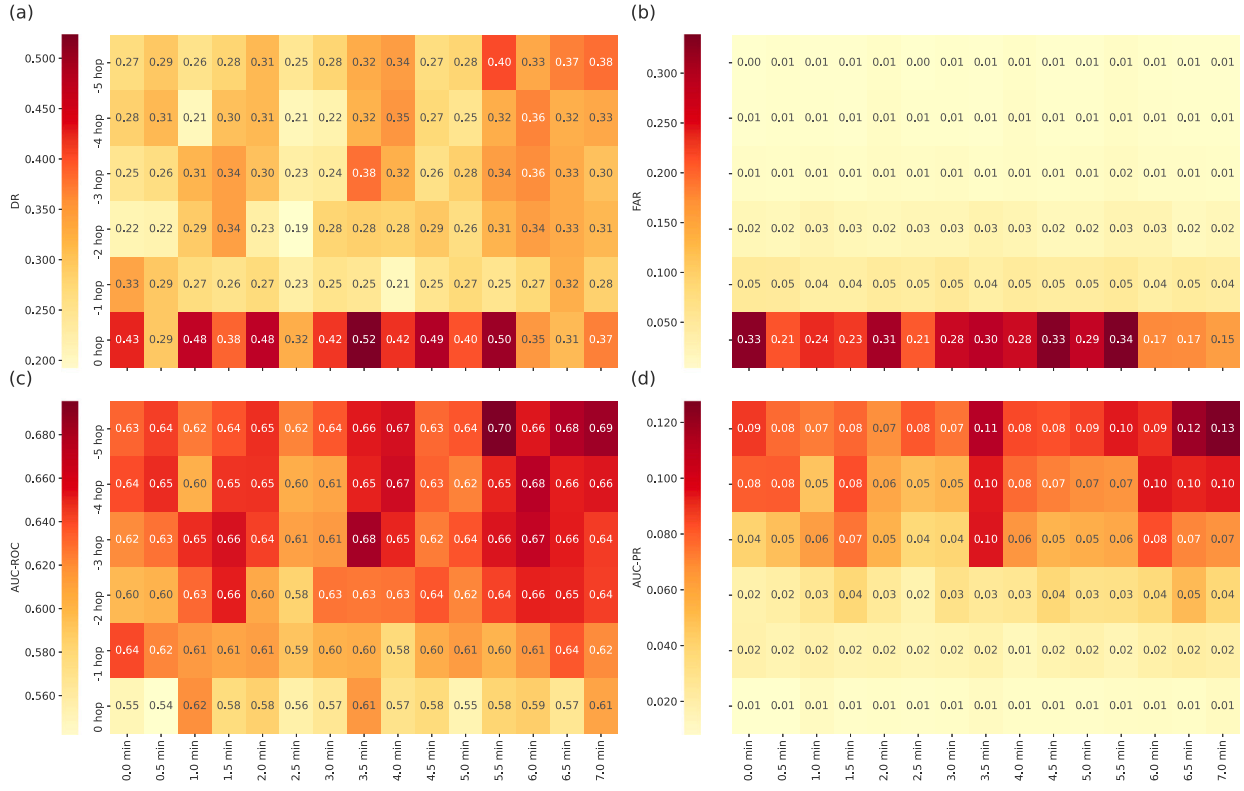

Fig. 8. The I-75 classification results when using a random forest classifier under Setting 1.

many sensors are used and the TTDA. This agrees with the results for I-75. We also find similar results under Setting 2: when using a logistic regression classifier, we see an improvement when including more sensors (upstream and downstream) or when considering extensive TTDA for training, but the improvements start stagnating as we increase the number of neighboring sensors.

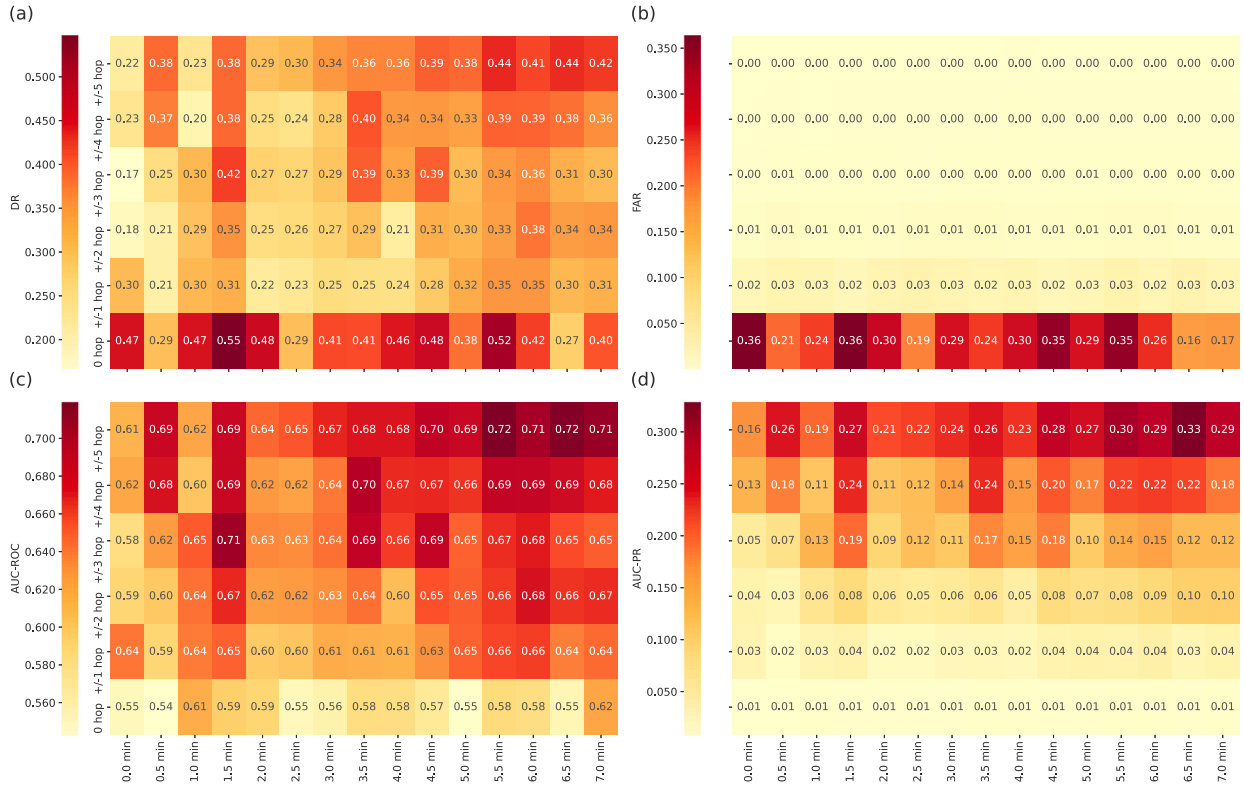

Fig. 9. The I-75 classification results when using a random forest classifier under Setting 2.

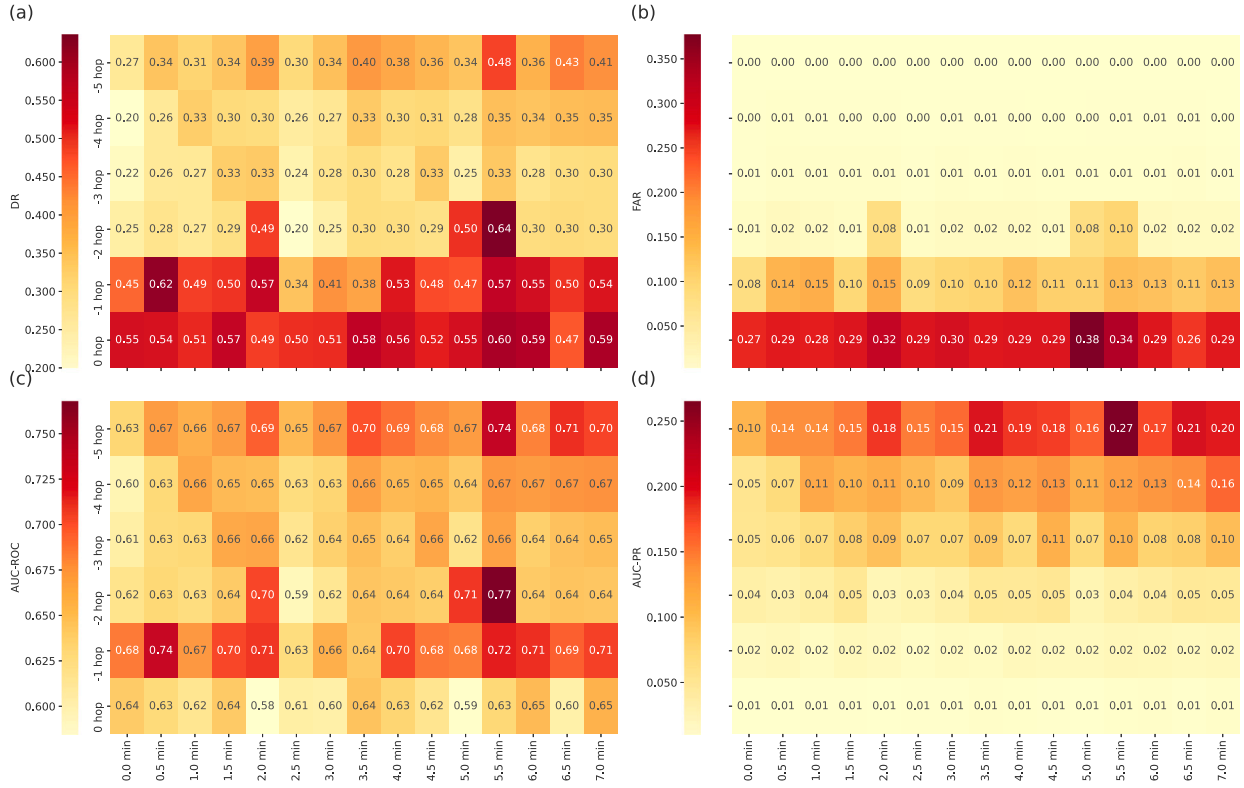

Fig. 10. The I-75 classification results when using an XGBoost classifier under Setting 1.

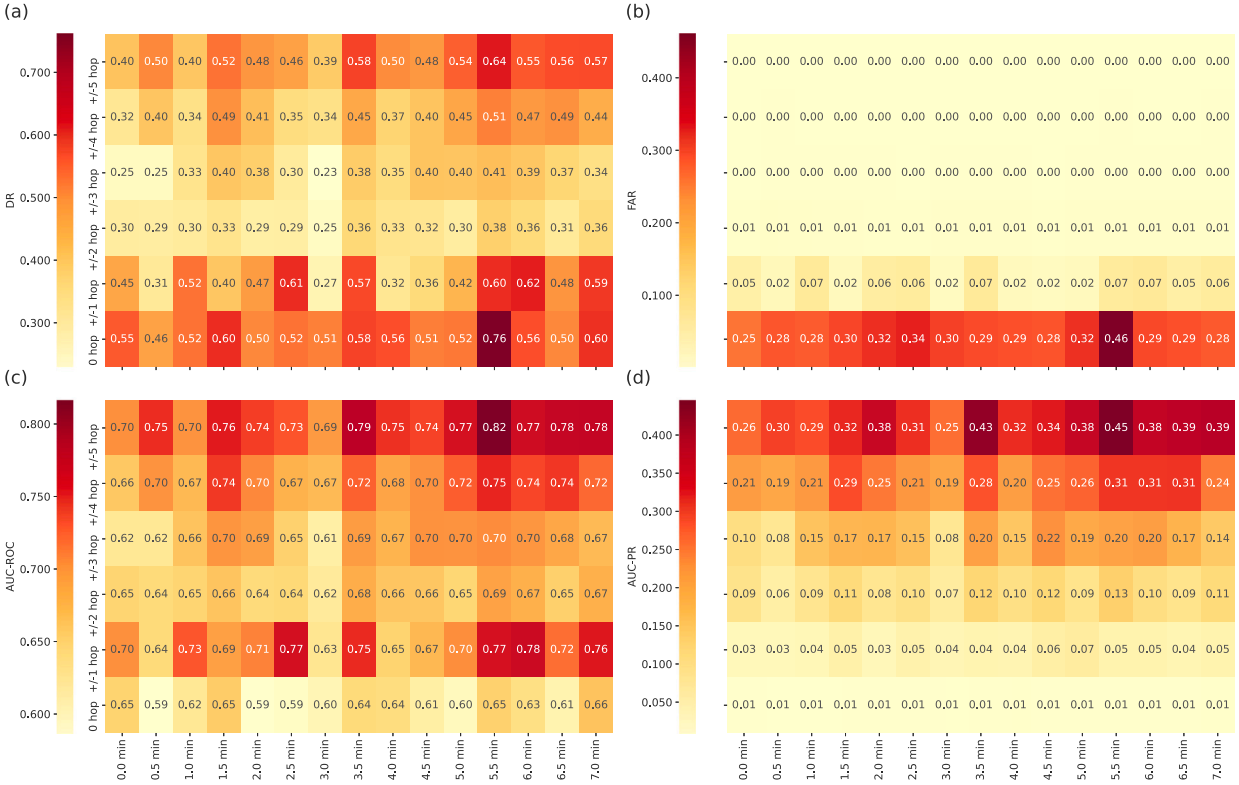

Fig. 11. The I-75 classification results when using an XGBoost classifier under Setting 2.

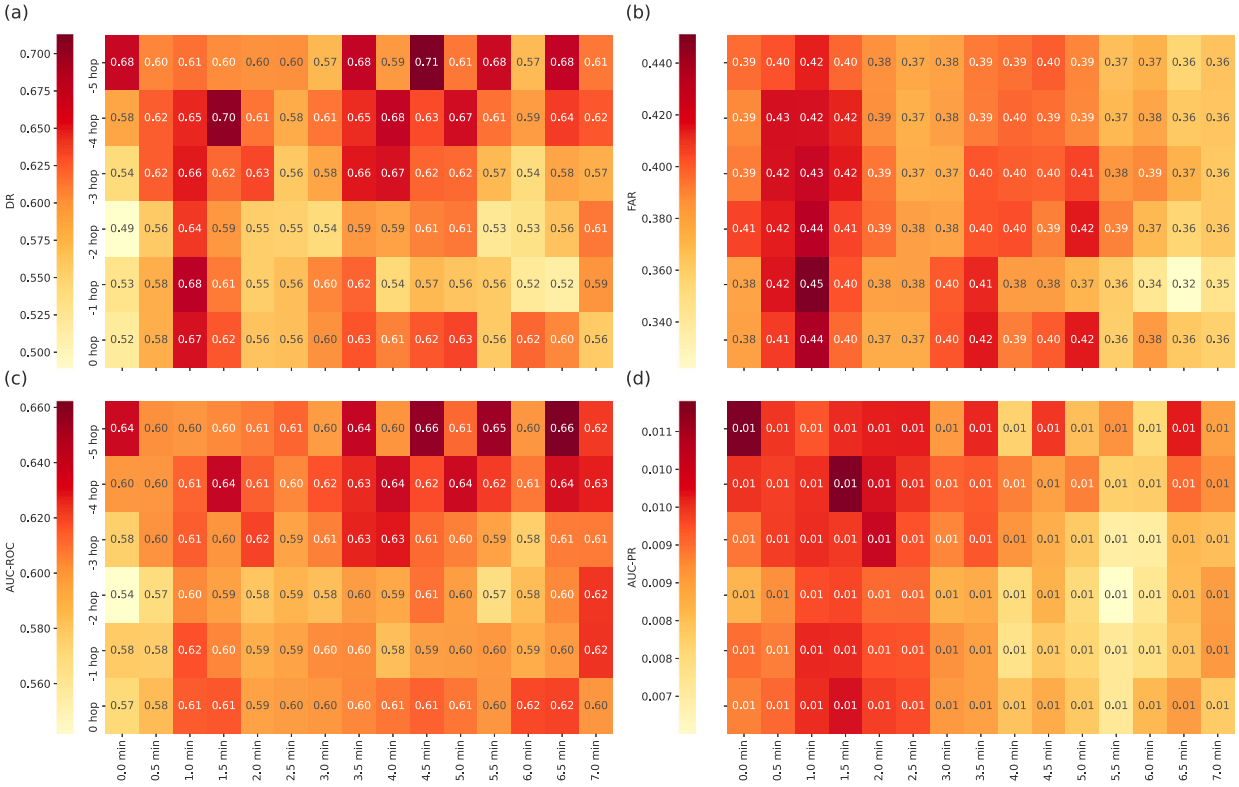

Fig. 12. The I-24 classification results when using a logistic regression classifier under Setting 1: (a) is DR, (b) is FAR, (c) is AUC-ROC, and (d) is AUC-PR.

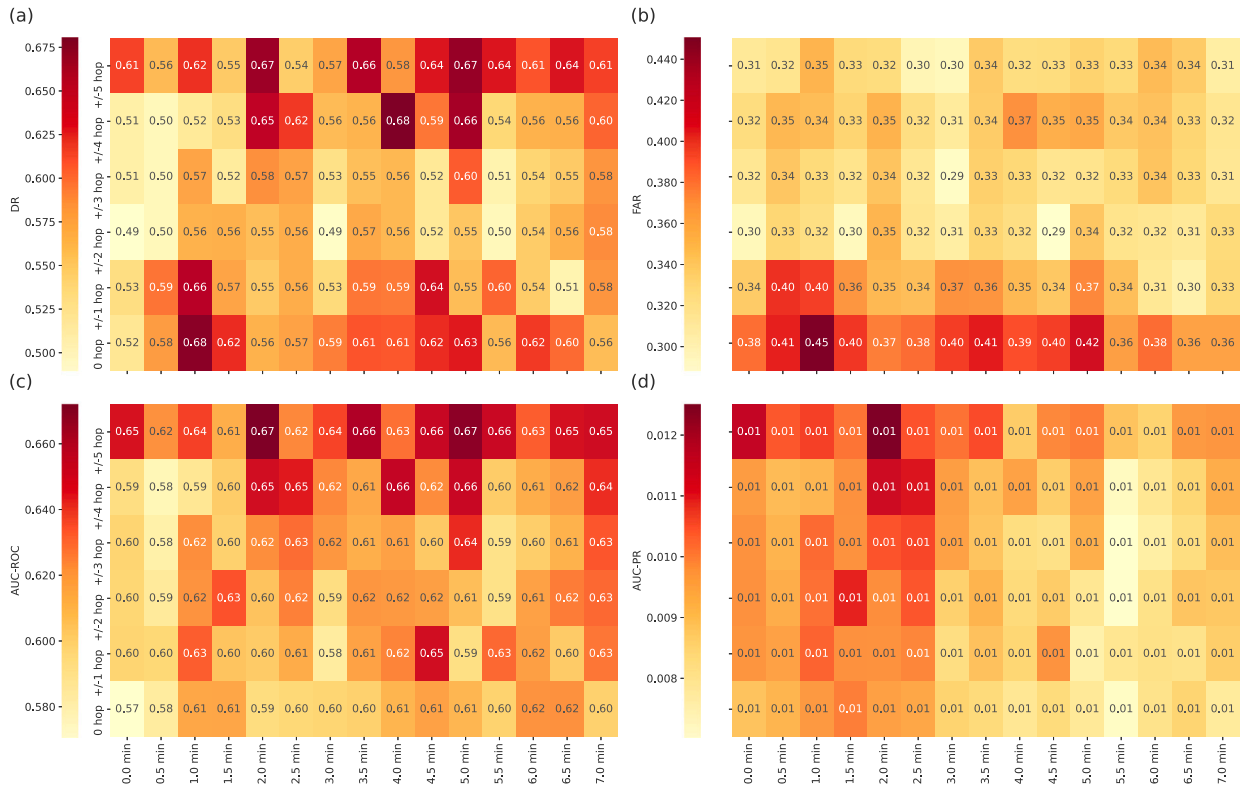

Fig. 13. The I-24 classification results when using a logistic regression classifier under Setting 2.

Figs. 14 and 15 show classification results for a random forest classifier under Settings 1 and 2, respectively. Under Setting 1, using more upstream sensors helps improve every classification metric. Additionally, the 19% performance peak in AUC-PR (with a corresponding 76% in AUC-ROC and 1% of FAR) is reached 5 min after an accident occurs. Under Setting 2, the random forest classifier consistently produces better results after incorporating more sensors. The earliest/best performance for AUC-PR is 44% (with a corresponding 76% in AUC-PR and negligible FAR), which is reached at 2.0 min after an accident occurs.

Figs. 16 and 17 show classification results for an XGBoost classifier under Settings 1 and 2, respectively. Under Setting 1, using more upstream sensors improves classification metrics. Specifically, the earliest performance peak for AUC-PR is 23% (with a corresponding AUC-ROC of 72% and negligible FAR) reached at 1.0 min. In agreement with experiments for I-75, under Setting 2, adding mores sensors (upstream and downstream) improves the overall detection of accidents vs. adding only upstream sensors. In Setting 2, the earliest performance peak for AUC-PR is 49% (with a corresponding AUC-ROC of 83% and negligible FAR) reached at 1.0 min after an accident. This represents an improvement of at least 26% in AUC-PR (from 23% to 49%) vs. Setting 1.

Table 2 summarizes the results for both settings, both datasets, and all classifiers. Note that even when reporting summary statistics such as the mean and median of performance metrics obtained with different sensor configurations and TTDA, we notice overall superior classification performance under Setting 2 and XGBoost classifier.

5.2. Feature importance analysis

Figs. 18 and 19 show a summary of feature importance based on classifier predictions in I-75 and I-24, respectively. Recall that we used SHAP to estimate feature importance (Section 4.4). The horizontal axis depicts individual SHAP values. The greater the value, the higher the impact on the prediction. The vertical axis represents individual features. Note that features are in decreasing order of importance (from top to bottom). For each feature, the color of each point is determined by its SHAP value: points with higher values are redder, and points with lower values are bluer. The figures highlight the top-15 most important features for the best performing classifier in each case study (i.e., XGBoost under Setting 2) and 5.5 and 3.5 min TTDA.

We observe that traffic-related features, including speed, volume, and occupancy, from downstream and upstream locations have the greatest impact on classifier outputs. Note the importance of considering both upstream and downstream features (Setting 2) vs. only upstream features (Setting 1). Accidents affect traffic features both upstream and downstream, so considering both helps improve classification performance and reduce TTDA. Additionally, measurements from sensors that are farther away from the accident locations (i.e., more than one hop apart) tend to have a stronger influence on classification results. In fact, perceived changes in traffic conditions from more distant sensors help discriminate between accidents and non-accident events vs. considering only the sensors closest to the accident. This reinforces the idea that a shock wave moves rapidly around neighboring locations after accidents, as discussed previously by Parsa et al. (2020) and Wang, Xie, Liu, Ragland, et al. (2016).

Note also that higher SHAP values associated with higher feature values (redder points) correspond to higher accident probabilities. Conversely, higher SHAP values associated with lower feature values (bluer points) correspond to lower accident probabilities. In general, when present in the top 15 of features, upstream measurements one-hop apart (i.e., −1 hop) tend to associate lower speed and lower volume values with higher chances of an accident. In contrast, downstream measurements one-hop apart (i.e., +1 hop) tend to associate higher speed and lower volume values with higher chances of an accident. This is consistent with previous empirical measurements on the topic, including the work by Shang et al. (2020).

Finally, weather- and lighting-related features tend to be less relevant features, and they do not appear in the top 15. That said, we found that snow conditions have a slight positive impact on accident

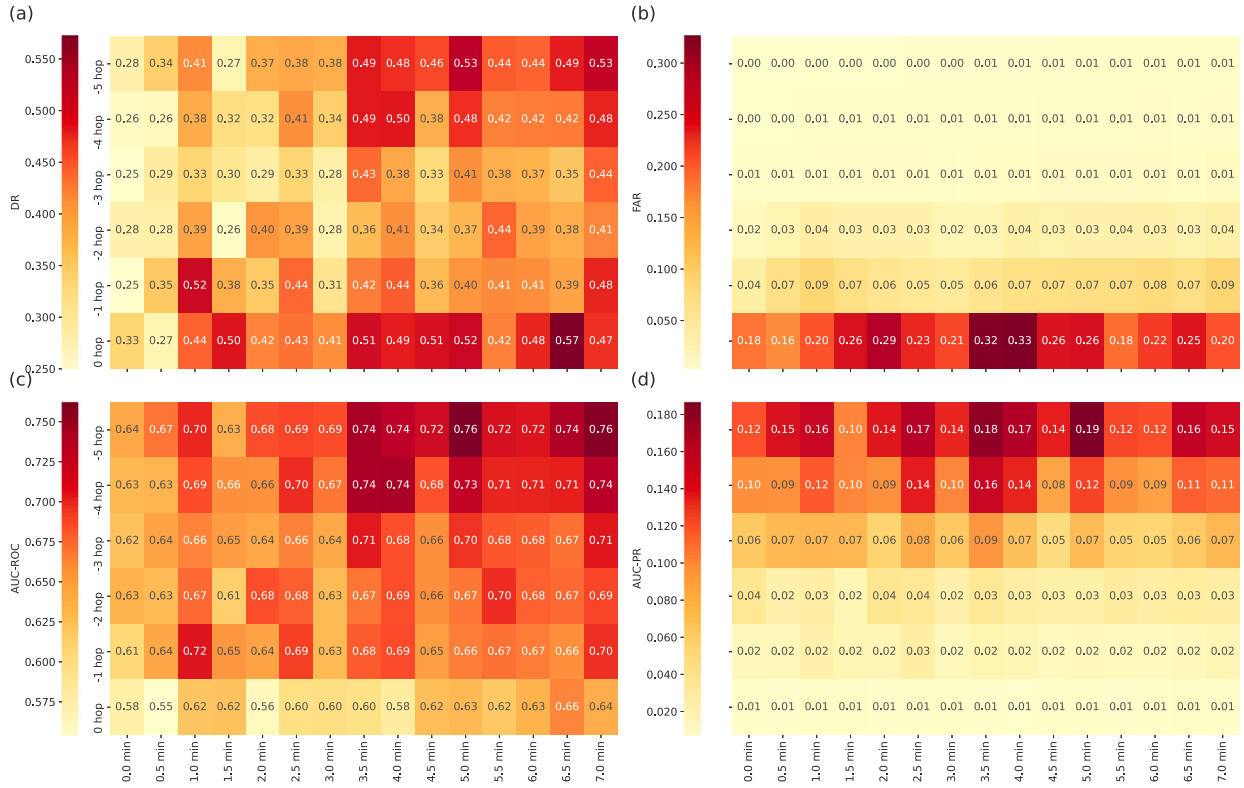

Fig. 14. The I-24 classification results when using a random forest classifier under Setting 1.

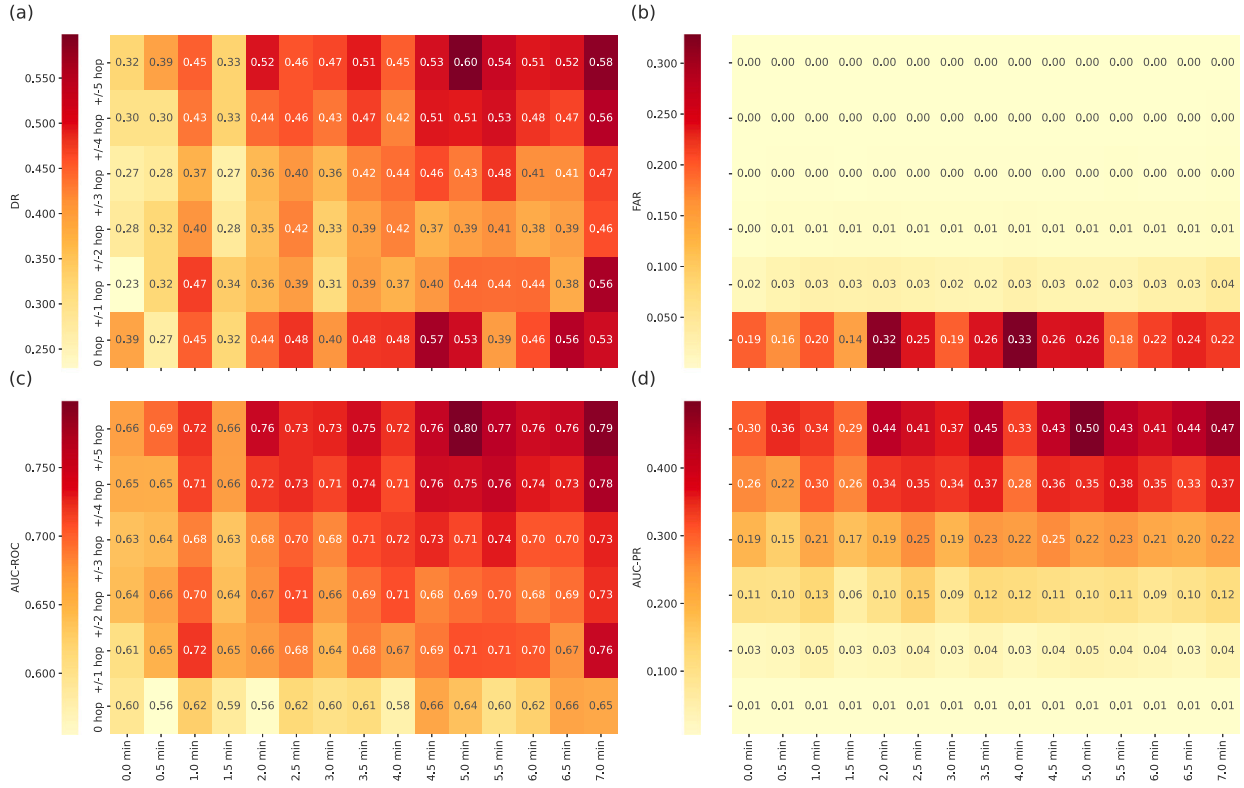

Fig. 15. The I-24 classification results when using a random forest classifier under Setting 2.

Table 2
Summary of performance evaluation. Numbers are presented as mean (standard deviation)/median. Classification methods are logistic regression (LR), random forest (RF), and XGBoost (XG).

Configuration			DR	FAR	AUC-ROC	AUC-PR
Setting 1	I-75	LR	0.57 (0.07)/0.57	0.40 (0.06) 0.40	0.59 (0.03)/0.59	0.01 (0.00)/0.01
		RF	0.31 (0.07)/0.30	0.06 (0.09)/0.02	0.62 (0.03)/0.63	0.05 (0.03)/0.04
		XG	0.39 (0.12)/0.34	0.08 (0.11)/0.01	0.66 (0.04)/0.65	0.07 (0.06)/0.05
	I-24	LR	0.60 (0.05)/0.60	0.39 (0.02)/0.39	0.61 (0.02)/0.60	0.01 (0.00)/0.01
		RF	0.39 (0.08)/0.40	0.06 (0.09)/0.02	0.67 (0.04)/0.67	0.06 (0.05)/0.04
		XG	0.41 (0.09)/0.40	0.04 (0.05)/0.01	0.69 (0.04)/0.68	0.09 (0.07)/0.07
Setting 2	I-75	LR	0.55 (0.07)/0.55	0.35 (0.05)/0.34	0.60 (0.04)/0.60	0.01 (0.00)/0.01
		RF	0.33 (0.08)/0.33	0.05 (0.10)/0.01	0.64 (0.04)/0.64	0.11 (0.09)/0.09
		XG	0.43 (0.11)/0.41	0.06 (0.11)/0.01	0.69 (0.05)/0.68	0.15 (0.12)/0.12
	I-24	LR	0.57 (0.05)/0.57	0.34 (0.03)/0.34	0.62 (0.02)/0.62	0.01 (0.00)/0.01
		RF	0.42 (0.08)/0.42	0.05 (0.09)/0.00	0.69 (0.05)/0.69	0.18 (0.15)/0.15
		XG	0.50 (0.12)/0.49	0.03 (0.06)/0.00	0.74 (0.07)/0.73	0.23 (0.18)/0.22

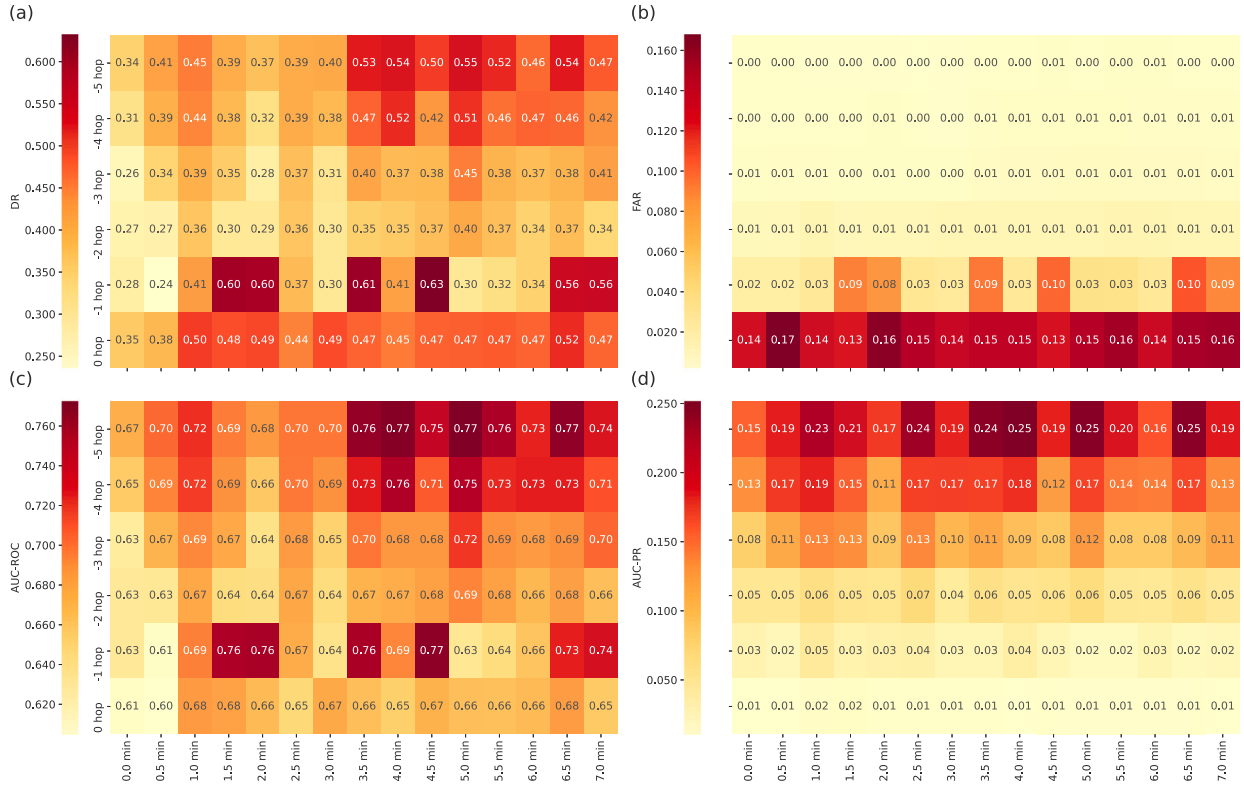

Fig. 16. The I-24 classification results when using an XGBoost classifier under Setting 1.

occurrence, followed by adverse lighting conditions (not shown in the figures).

6. Discussion

Reducing accident response time is crucially important for saving lives. Recent progress has shown the potential of using ML classifiers and traffic-related data to achieve this goal. However, despite the use of more sophisticated classifiers, more progress must be made to reduce the TTDA and improve classification performance, both of which have plateaued. To that end, this work proposes a novel automatic accident detection framework that exploits the topological deployment of sensors in the road and their associated data. Specifically, we show how using data from neighboring sensors around accidents, along with weather and lighting data, can reduce TTDA while improving classification performance. The proposed framework shows how we can optimize the automatic accident detection by combining spatiotemporal traffic data to minimize the TTDA and maximize classification performance.

To validate the effectiveness of the proposed framework, we analyzed accident detection on two highway stretches over I-75 and I-24 in the Chattanooga metropolitan area (Section 3.1). We tested different ML classifiers (logistic regression, random forest, and XGBoost) over a dataset of 24 accidents and 3039 non-accident events for I-75 and 31 accidents and 4852 non-accidents events for I-24. After carefully balancing the dataset and using traffic data from both upstream and downstream locations, the best performing classifier (XGBoost) achieves 45% AUC-PR, 82% AUC-ROC, 64% DR, and negligible FAR at 5.5 min TTDA for I-75 and 56% AUC-PR, 86% ROC-AUC, 73% DR, and negligible FAR at 3.5 min TTDA for I-24.

We evaluated how different configurations of spatiotemporal traffic data affect the classification task for detecting accidents at road segments. Including both upstream and downstream traffic data up to a certain distance increases classification performance while reducing the TTDA. We performed a similar analysis by including only traffic data from upstream traffic and corroborated that adding traffic data from downstream sensors leads to more accurate accident detection and reduced TTDA. We tested this key observation by using different

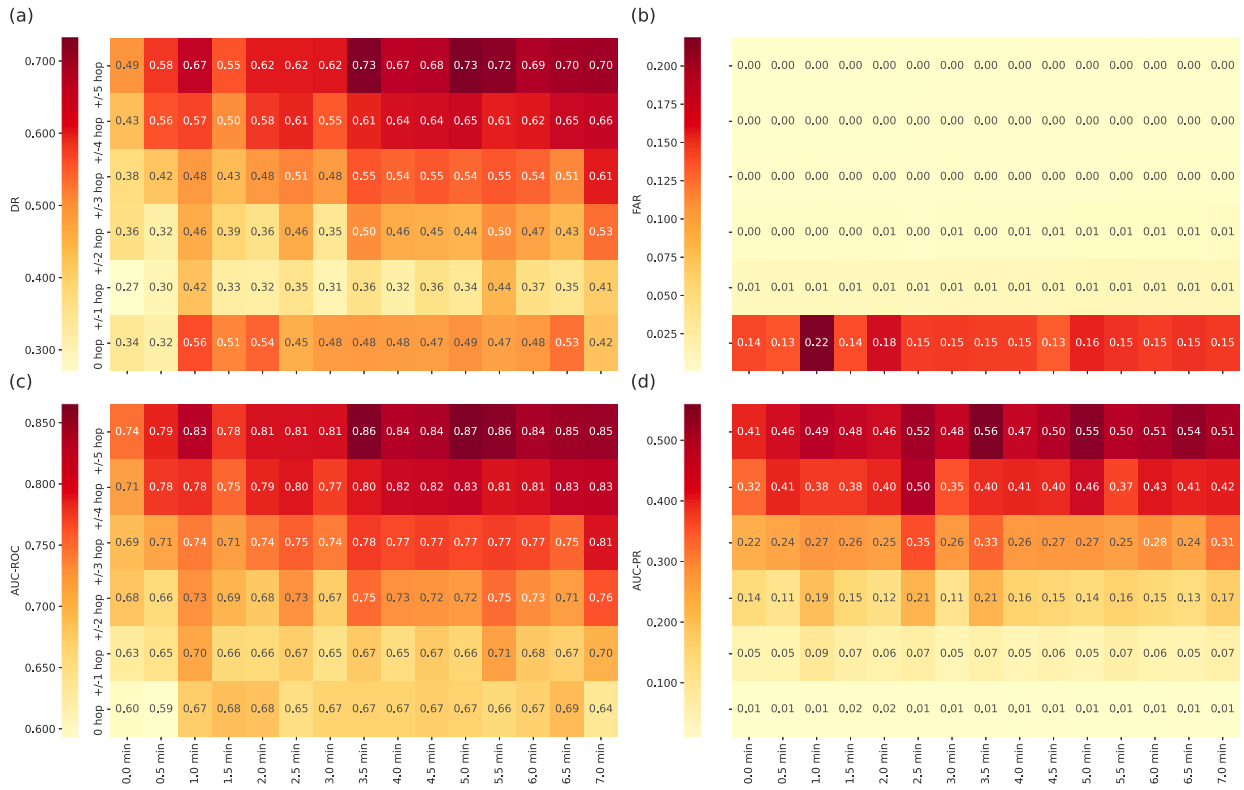

Fig. 17. The I-24 classification results when using an XGBoost classifier under Setting 2.

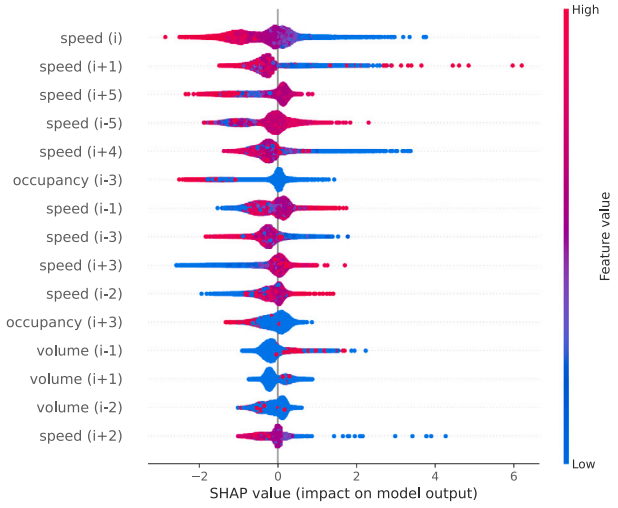

Fig. 18. Feature importance analysis for the best performing classifier in the I-75 analysis.

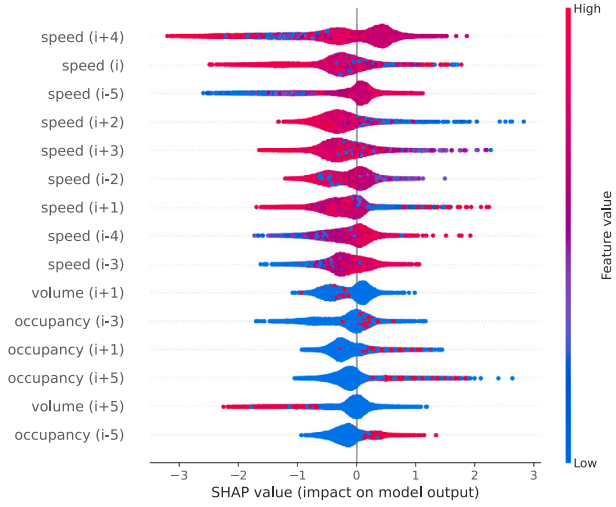

Fig. 19. Feature importance analysis for the best performing classifier in the I-24 analysis.

ML classifiers and computed the most important features that drive classification.

The proposed framework provides insights on how to leverage traffic data to reduce accident response time and increase classification performance based on spatiotemporal analysis of traffic. Having noted the potential of our approach, we are also aware of the followings limitations of our proposed work.

Lack of Real-Time Field Test: Our proposed framework was trained and tested on empirical, static datasets. We reported results on these datasets and showed that, based on the configuration of sensors used in the classification task, it is possible to guarantee a peak in performance for a specific TTDA. We did not perform a real-time field evaluation of the proposed framework because real-time data feeds (while under development by the transportation agency) were not yet production-ready.

Continuous Classifier Adaptation: We presented results on the performance of classifiers trained and tested on static datasets. However, we acknowledge that accident patterns continuously evolve, and classifiers must adapt based on updated traffic patterns of accident and non-accident conditions. Thus, retraining or incrementally updating the classifiers requires thoughtful exploration to provide adaptation.

Distinction between Other Road Events: The incident datasets used contain only accident events and associated metadata. Although the traffic sensor data includes all scenarios for the studied time frame, the

incident datasets did not include ground-truth data for other events that may influence traffic patterns (e.g., construction, special events in the city). This means that the accident detection presented in this paper cannot account for such incidents.

Topology of Sensor Placement: In the case studies, we focused on a single linear neighborhood between sensors near junctions. However, incidents that occur close to junctions may affect upstream or downstream traffic on multiple highways. For instance, if there was an accident on I-75 southbound immediately north of its junction with I-24, then this would affect downstream traffic on I-24 westbound and I-75 southbound. Similarly, an accident in the northbound direction could affect upstream traffic from I-24 eastbound and I-75 northbound.

Use of Default Parameter Values in ML Classifiers: We performed a grid search over a subset of hyperparameters for the proposed classifiers that had been explored in previous research (Van Rijn & Hutter, 2018). Therefore, we did not explore additional hyperparameters that could further improve performance.

Results from this work show how to reduce the TTDA by using ML classifiers and leveraging the traffic impact of accidents in upstream and downstream traffic conditions. We anticipate that field experiments can be performed to validate the results of this work based on the rich data sources readily available, as we have shown in our two case studies.

7. Conclusion

We show the utility of using spatiotemporal traffic data sources for the quick and accurate detection of accidents by leveraging different ML classifiers (i.e., logistic regression, random forest, and XGBoost). As a complement to current automatic accident detection approaches, we have demonstrated a proof-of-concept to reduce the accident response time while increasing classification performance to as early as 1.0 min after accident occurrence with an AUC-ROC of up to 83% and an AUC-PR of up to 49%. The proposed framework relies on empirical traffic data along with weather and lighting data sources. We have demonstrated the benefits of using our approach on a set of 24 accidents on I-75 and 31 accidents on I-24 in the Chattanooga metropolitan area. The framework proposed in this paper relies on accidents having a shock wave effect that expands to neighboring locations upstream and downstream from accidents. Thus, considering traffic data from more distant accident locations helps to more quickly identify accidents. Relying on this observation, we detailed an automatic accident detection framework based on ML classification for quick and accurate detection of accidents. Specifically, we showed that the proposed framework can detect accidents rapidly while still performing reasonably well. After 2.0 min TTDA, we achieved 38% AUC-PR, 74% AUC-ROC, 48% DR, and negligible FAR for I-75; after 1.0 min TTDA, we achieved 49% AUC-PR, 83% ROC-AUC, 67% DR, and negligible FAR for I-24.

Future work in this area includes the following ideas:

- Real-time incident detection: examine the effectiveness of the proposed framework with real-time traffic data from radar detectors. We expect that in the near future, this data can be readily available and accessible through APIs across different transportation authorities in the United States.
- Integration in traffic analysis platforms: integrate the proposed methods in data-driven platforms for transportation analysis, monitoring, and data visualization, such as the Regional Integrated Transportation Information System, which is coming online across many states.
- Probe data: using probe data as an alternative to stationary sensors to obtain link-level speeds. As the penetration rates for most available probe data are fairly low (5-10% of all vehicles/devices in many areas), the resulting traffic counts may not be reliable and there could be delays in reporting speed changes due to the sparser sampling of vehicles. However, probe data have the advantage of not requiring traffic infrastructure to be in place.

In summary, an implementation of a prototype for automatic accident detection, based on the principles described here, should be feasible with the availability of real-time data from diverse traffic-related, data-driven platforms. By detailing our experimental procedure in this work and sharing our dataset with the research community (Berres et al., 2023), we hope that further studies can reflect on comparing a broader set of classification models beyond logistic regression, random forest, and XGBoost. We look forward to further research in this area.

CRediT authorship contribution statement

Pablo Moriano: Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Software, Validation, Visualization, Writing – original draft, Writing – review & editing. **Andy Berres:** Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Project administration, Validation, Visualization, Writing – original draft, Writing – review & editing. **Haowen Xu:** Data curation, Visualization, Writing – review & editing. **Jibonananda Sanyal:** Funding acquisition, Project administration, Resources, Supervision.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Pablo Moriano, Andy Berres, Haowen Xu, Jibonananda Sanyal has patent pending to UT-Battelle.

Data availability

The dataset used in this research is available at: https://zenodo.org/record/7964288.

Acknowledgments

The authors would like to thank TDOT for providing radar and E-TRIMS data and the Chattanooga Department of Transportation and Chattanooga Transportation Planning Organization for their guidance and for providing the road network. POWER data was obtained from the NASA Langley Research Center's POWER Project, which is funded through the NASA Earth Science/Applied Science Program. This work was supported in part by the US Department of Energy (DOE) through UT-Battelle LLC under contract DE-AC05-00OR22725 and in part by the US DOE's Office of Energy Efficiency and Renewable Energy and the Vehicle Technologies Office, United States. Finally, the authors would like to thank Sarah Tennille for her support in data acquisition and exploration and Srinath Ravulaparthy and Rajesh Paleti for their discussions and support of early manual explorations of this data.

References

Abdulhai, B., & Ritchie, S. G. (1999). Enhancing the universality and transferability of freeway incident detection using a Bayesian-based neural network. *Transportation Research Part C (Emerging Technologies)*, *7*(5), 261–280.

Ahmed, S., & Cook, A. R. (1982). Application of time-series analysis techniques to freeway incident detection. *Transportation Research Record*.

American Transportation Research Institute (ATRI) (2023). Top 100 truck bottlenecks – 2022. https://truckingresearch.org/2022/02/08/top-100-truck-bottlenecks-2022. Online; accessed: 2023-02-20.

Anda, C., Erath, A., & Fourie, P. J. (2017). Transport modelling in the age of big data. *International Journal of Urban Sciences*, *21*(sup1), 19–42.

Berres, A. S., LaClair, T. J., Wang, C., Xu, H., Ravulaparthy, S., Todd, A., et al. (2021). Multiscale and multivariate transportation system visualization for shopping district traffic and regional traffic. *Transportation Research Record*, *2675*(6), 23–37.

Berres, A., Moriano, P., Xu, H., Tennille, S., Smith, L., Storey, J., et al. (2023). A tagged traffic accident dataset for machine learning. http://dx.doi.org/10.5281/zenodo.7964288.

Berres, A. S., Xu, H., Tennille, S. A., Severino, J., Ravulaparthy, S., & Sanyal, J. (2021). Explorative visualization for traffic safety using Adaptive Study Areas. *Transportation Research Record*, *2675*(6), 51–69. http://dx.doi.org/10.1177/0361198120981065, arXiv:https://doi.org/10.1177/0361198120981065.

Bibri, S. E., & Krogstie, J. (2020). The emerging data–driven smart city and its innovative applied solutions for sustainability: The cases of London and Barcelona. *Energy Informatics*, *3*(1), 1–42.

Breiman, L. (2001). Random forests. *Machine Learning*, *45*(1), 5–32.

Chakraborty, P., Hegde, C., & Sharma, A. (2019). Data-driven parallelizable traffic incident detection using spatio-temporally denoised robust thresholds. *Transportation Research Part C: Emerging Technologies*, *105*, 81–99.

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, *16*, 321–357.

Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. In *KDD* (pp. 785–794).

Chen, S., Wang, W., & Van Zuylen, H. (2009). Construct support vector machine ensemble to detect traffic incident. *Expert Systems with Applications*, *36*(8), 10976–10986.

Cheu, R. L., & Ritchie, S. G. (1995). Automated detection of lane-blocking freeway incidents using artificial neural networks. *Transportation Research Part C (Emerging Technologies)*, *3*(6), 371–388.

Cook, A. R., & Cleveland, D. E. (1974). Detection of freeway capacity-reducing incidents by traffic-stream measurements. *Transportation Research Record*, *495*(1–11), 5.

Davis, J., & Goadrich, M. (2006). The relationship between precision-recall and ROC curves. In *Proceedings of the 23rd international conference on machine learning* (pp. 233–240).

Dia, H., & Rose, G. (1997). Development and evaluation of neural network freeway incident detection models using field data. *Transportation Research Part C (Emerging Technologies)*, *5*(5), 313–331.

García, V., Sánchez, J. S., & Mollineda, R. A. (2012). On the effectiveness of preprocessing methods when dealing with different levels of class imbalance. *Knowledge-Based Systems*, *25*(1), 13–21.

Han, X., Grubenmann, T., Cheng, R., Wong, S. C., Li, X., & Sun, W. (2020). Traffic incident detection: A trajectory-based approach. In *2020 IEEE 36th international conference on data engineering (ICDE)* (pp. 1866–1869). IEEE.

Hosmer, D. W., Jr., Lemeshow, S., & Sturdivant, R. X. (2013). *Applied logistic regression*. John Wiley & Sons.

Jägerbrand, A. K., & Sjöbergh, J. (2016). Effects of weather conditions, light conditions, and road lighting on vehicle speed. *SpringerPlus*, *5*(1), 1–17.

Jin, X., Cheu, R. L., & Srinivasan, D. (2002). Development and adaptation of constructive probabilistic neural network in freeway incident detection. *Transportation Research Part C (Emerging Technologies)*, *10*(2), 121–147.

Kalair, K., & Connaughton, C. (2021). Anomaly detection and classification in traffic flow data from fluctuations in the flow–density relationship. *Transportation Research Part C (Emerging Technologies)*, *127*, Article 103178.

Kitali, A. E., Alluri, P., Sando, T., & Wu, W. (2019). Identification of secondary crash risk factors using penalized logistic regression model. *Transportation Research Record*, *2673*(11), 901–914.

Klein, L. A. (2001). Sensor technologies and data requirements for ITS.

Lemaître, G., Nogueira, F., & Aridas, C. K. (2017). Imbalanced-learn: A python toolbox to tackle the curse of imbalanced datasets in machine learning. *Journal of Machine Learning Research*, *18*(17), 1–5.

Li, L., Sheng, X., Du, B., Wang, Y., & Ran, B. (2020). A deep fusion model based on restricted Boltzmann machines for traffic accident duration prediction. *Engineering Applications of Artificial Intelligence*, *93*, Article 103686.

Lin, Y., Li, L., Jing, H., Ran, B., & Sun, D. (2020). Automated traffic incident detection with a smaller dataset based on generative adversarial networks. *Accident Analysis and Prevention*, *144*, Article 105628.

Liu, X., Cai, H., Zhong, R., Sun, W., & Chen, J. (2020). Learning traffic as images for incident detection using convolutional neural networks. *IEEE Access*, *8*, 7916–7924.

Loten, A. (2019). 911 Response times are getting faster thanks to data integration. *Wall Street Journal*, *13*, Published June.

Louppe, G. (2014). Understanding random forests: From theory to practice. arXiv preprint arXiv:1407.7502.

Lu, J., Chen, S., Wang, W., & Van Zuylen, H. (2012). A hybrid model of partial least squares and neural network for traffic incident detection. *Expert Systems with Applications*, *39*(5), 4775–4784.

Lundberg, S. M., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. *Advances in Neural Information Processing Systems*, *30*.

Motamed, M., et al. (2016). *Developing a real-time freeway incident detection model using machine learning techniques* (Ph.D. thesis), University of Texas at Austin.

NASA (2023). The POWER project. https://power.larc.nasa.gov. Online; ccessed: 2023-02-20.

National Highway Traffic Safety Administration (2021). *Traffic safety facts 2019: A compilation of motor vehicle crash data: (Report no. DOT HS 813 141)*, Washington, DC: National Highway Traffic, Safety Administration.

National Weather Service (NWS) (2023). Wind information page. https://www.weather.gov/dmx/dsswind. Online; accessed: 2023-02-20.

Nicodemus, K. K., Malley, J. D., Strobl, C., & Ziegler, A. (2010). The behaviour of random forest permutation-based variable importance measures under predictor correlation. *BMC Bioinformatics*, *11*(1), 1–13.

Nowak, A. S., & Radzik, T. (1994). The Shapley value for n-person games in generalized characteristic function form. *Games and Economic Behavior*, *6*(1), 150–161.

Ozbayoglu, M., Kucukayan, G., & Dogdu, E. (2016). A real-time autonomous highway accident detection model based on big data processing and computational intelligence. In *2016 IEEE international conference on big data (big data)* (pp. 1807–1813). IEEE.

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

Parsa, A. B., Taghipour, H., Derrible, S., & Mohammadian, A. K. (2019). Real-time accident detection: coping with imbalanced data. *Accident Analysis and Prevention*, *129*, 202–210.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., et al. (2011). Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, *12*, 2825–2830.

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). "Why should I trust you?" Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 1135–1144).

Roland, J., Way, P. D., Firat, C., Doan, T.-N., & Sartipi, M. (2021). Modeling and predicting vehicle accident occurrence in Chattanooga, Tennessee. *Accident Analysis and Prevention*, *149*, Article 105860. http://dx.doi.org/10.1016/j.aap.2020.105860, URL: https://www.sciencedirect.com/science/article/pii/S0001457520316808.

Saito, T., & Rehmsmeier, M. (2015). The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets. *PLoS One*, *10*(3), Article e0118432.

Shang, Q., Feng, L., & Gao, S. (2020). A hybrid method for traffic incident detection using random forest-recursive feature elimination and long short-term memory network with Bayesian optimization algorithm. *IEEE Access*.

Srinivasan, D., Cheu, R. L., Poh, Y. P., & Ng, A. K. C. (2000). Development of an intelligent technique for traffic network incident detection. *Engineering Applications of Artificial Intelligence*, *13*(3), 311–322.

Srinivasan, D., Sharma, V., & Toh, K.-A. (2008). Reduced multivariate polynomial-based neural network for automated traffic incident detection. *Neural Networks*, *21*(2–3), 484–492.

Strobl, C., Boulesteix, A.-L., Zeileis, A., & Hothorn, T. (2007). Bias in random forest variable importance measures: Illustrations, sources and a solution. *BMC Bioinformatics*, *8*(1), 1–21.

Štrumbelj, E., & Kononenko, I. (2014). Explaining prediction models and individual predictions with feature contributions. *Knowledge and Information Systems*, *41*(3), 647–665.

Sunrise Sunset (2023). Sunrise and sunset times in chattanooga, TN. https://sunrise-sunset.org/us/chattanooga-tn. Online; accessed: 2023-02-20.

Taghipour, H., Parsa, A. B., Chauhan, R. S., Derrible, S., & Mohammadian, A. K. (2022). A novel deep ensemble based approach to detect crashes using sequential traffic data. *IATSS Research*, *46*(1), 122–129.

Tang, S., & Gao, H. (2005). Traffic-incident detection-algorithm based on nonparametric regression. *IEEE Transactions on Intelligent Transportation Systems*, *6*(1), 38–42.

TDOT (2022). Travel time messaging on dynamic message signs. https://ops.fhwa.dot.gov/publications/travel_time_study/nashville/nashville_ttm.htm. Online; accessed: 2023-08-29.

Teng, H., & Qi, Y. (2003). Detection-delay-based freeway incident detection algorithms. *Transportation Research Part C (Emerging Technologies)*, *11*(3–4), 265–287.

The Wall Street Journal (2023). 911 Response times are getting faster thanks to data integration. https://www.wsj.com/articles/911-response-times-are-getting-faster-thanks-to-data-integration-11560468747n. Online; accessed: 2023-03-02.

United States Geological Survey (USGS) (2023). How do weather events impact roads? https://water.usgs.gov/edu/activity-howmuchrain-metric.html#:~:text=Heavy%20rain%3A%20Greater%20than%204,than%2010%20mm%20per%20hour. Online; accessed: 2023-02-20.

US DOT Federal Highway Association (FHWA) (2023a). How do weather events impact roads? https://ops.fhwa.dot.gov/weather/q1_roadimpact.htm. Online; accessed: 2023-02-20.

US DOT Federal Highway Association (FHWA) (2023b). Nighttime visibility. https://safety.fhwa.dot.gov/roadway_dept/night_visib/general-information.cfm. Online; accessed: 2023-02-20.

Van Rijn, J. N., & Hutter, F. (2018). Hyperparameter importance across datasets. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining* (pp. 2367–2376).

Wang, J., Li, X., Liao, S. S., & Hua, Z. (2013). A hybrid approach for automatic incident detection. *IEEE Transactions on Intelligent Transportation Systems*, *14*(3), 1176–1185.

Wang, J., Xie, W., Liu, B., Ragland, D. R., et al. (2016). Identification of freeway secondary accidents with traffic shock wave detected by loop detectors. *Safety Science*, *87*, 195–201.

Xiao, J. (2019). SVM and KNN ensemble learning for traffic incident detection. *Physica A. Statistical Mechanics and its Applications*, *517*, 29–35.

Xu, H., Berres, A., Tennille, S. A., Ravulaparthy, S. K., Wang, C., & Sanyal, J. (2022). Continuous emulation and multiscale visualization of traffic flow using stationary roadside sensor data. *IEEE Transactions on Intelligent Transportation Systems*, *23*(8), 10530–10541. http://dx.doi.org/10.1109/TITS.2021.3094808.

Xu, H., Berres, A., Yoginath, S. B., Sorensen, H., Nugent, P. J., Severino, J., et al. (2023). Smart mobility in the cloud: Enabling real-time situational awareness and cyber-physical control through a digital twin for traffic. *IEEE Transactions on Intelligent Transportation Systems*, *24*(3), 3145–3156. http://dx.doi.org/10.1109/TITS.2022.3226746.

Yuan, F., & Cheu, R. L. (2003). Incident detection using support vector machines. *Transportation Research Part C (Emerging Technologies)*, *11*(3–4), 309–328.