Designing a Human-centered AI Tool for Proactive Incident Detection using Crowdsourced Data Sources to Support Emergency Response

YASAS SENARATH, George Mason University, USA
AYAN MUKHOPADHYAY, Vanderbilt University, USA
HEMANT PUROHIT, George Mason University, USA
ABHISHEK DUBEY, Vanderbilt University, USA

Time of incident reporting is a critical aspect of emergency response. However, the conventional approaches to receiving incident reports have time delays. Non-traditional sources such as crowdsourced data present an opportunity to detect incidents proactively. However, detecting incidents from such data streams is challenging due to inherent noise and data uncertainty. Naively maximizing detection accuracy can compromise spatial-temporal localization of inferred incidents, hindering response efforts. This paper presents a novel human-centered AI tool to address the above challenges. We demonstrate how crowdsourced data can aid incident detection while acknowledging associated challenges. We use an existing *CROME* framework to facilitate training and selection of *best* incident detection models, based on parameters suited for deployment. The human-centered AI tool provides a visual interface for exploring various measures to analyze the models for the practitioner's needs, which could help the practitioners select the best model for their situation. Moreover, in this study, we illustrate the tool usage by comparing different models for incident detection. The experiments demonstrate that the CNN-based incident detection method can detect incidents significantly better than various alternative modeling approaches. In summary, this research demonstrates a promising application of human-centered AI tools for incident detection to support emergency response agencies.

CCS Concepts: • **Human-centered computing → Visualization systems and tools**; • **Applied computing → Computing in government**; • **Computing methodologies → Machine learning**.

Additional Key Words and Phrases: Emergency Response, Incident Detection, Human-centered AI Tool, Crowdsourcing

ACM Reference Format:
Yasas Senarath, Ayan Mukhopadhyay, Hemant Purohit, and Abhishek Dubey. 2023. Designing a Human-centered AI Tool for Proactive Incident Detection using Crowdsourced Data Sources to Support Emergency Response. *Digit. Gov. Res. Pract.* 1, 1, Article 1 (January 2023), 19 pages. https://doi.org/10.1145/3633784

1 INTRODUCTION

All across the globe, people call responders from government agencies such as emergency services, fire departments, and police departments for assistance in situations of distress. These calls involve *incidents* like accidents, natural disasters, and mental and physical health crises. FEMA[1] defines an incident as "An occurrence, natural or human-caused, that requires a response to protect life or property". Responders aim to reach the scene *as quickly*

[1] https://training.fema.gov/programs/emischool/el361toolkit/glossary.htm

Authors' addresses: Yasas Senarath, ywijesu@gmu.edu, George Mason University, USA; Ayan Mukhopadhyay, ayan.mukhopadhyay@vanderbilt.edu, Vanderbilt University, USA; Hemant Purohit, hpurohit@gmu.edu, George Mason University, USA; Abhishek Dubey, abhishek.dubey@vanderbilt.edu, Vanderbilt University, USA.

2639-0175/2023/1-ART1
https://doi.org/10.1145/3633784

as possible to minimize the risk to human life [10, 11]. In order to minimize response times, it is imperative for responders to also perform proactive planning, i.e., resources must be strategically stationed (and dispatched) in anticipation of future incidents [19]. A major bottleneck, however, is receiving the notification that an incident has occurred and dispatch is needed [18]. Typically, responding agencies depend on traditional and official channels for someone to report an incident (e.g., through the 911 emergency telephone number in the United States) [18], *after* which a responder is dispatched. This bottleneck raises an important question: *can alternate sources of data (e.g., crowdsourced data from social media and IoT traffic sensors) be used to gain situational awareness proactively such that incidents (and possibly requirements on the scene) can be gauged before they are officially reported*? In this paper, we present a human-centered AI tool design called *CROMEx* to *detect* incidents such as traffic accidents from non-traditional, crowdsourced information by using Artificial Intelligence (AI) techniques. The resulting tool design can be used by government agencies to expedite emergency response as per their needs.

The desired capability for incident detection is a recent addition to the information systems that support emergency response pipeline of government agencies. Traditionally, the pipeline consisted of a complex set of estimation and decision-making problems, for instance, the time and place of incident occurrence must be estimated, responders must be strategically allocated, and then the appropriate resource must be dispatched when required [18]. The entire pipeline operates on the central premise that dispatching a resource after it has been reported suffices. This assumption is partly based on the lack of an approach to do otherwise, i.e., for decades, reporting through channels such as 911 in the U.S. was the only established mechanism for dispatching resources (with some exceptions, for example, inter-agency communication through radio). However, the advent of big data has enabled the use of multiple channels of information to detect incidents *before* they are reported. For example, consider an urban fire. As people observe smoke and fire, they share it on social media (e.g., Twitter). Drivers share information about accidents through crowdsourced applications such as Waze [2]. Responding agencies can use such crowdsourced information to accurately detect the type, place, and time of occurrence of incidents and dispatch resources quickly.

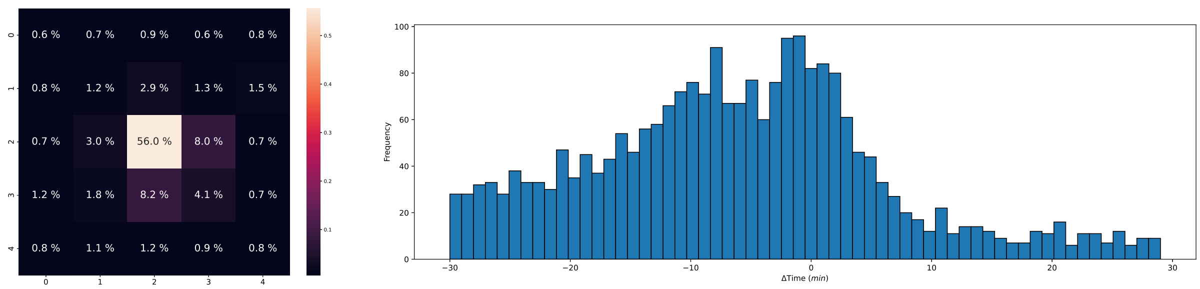
While using crowdsourced information to detect the occurrence of incidents has the natural advantage of potentially expediting response, it also presents challenges, both from technical and policy perspectives. For example, emergency response, in general, deals with situations that are critical. As a result, resources can only be dispatched to incidents whose occurrence (and the requirement for resources such as an ambulance) has been detected with some *degree of certainty*. An erroneous dispatch (i.e., dispatch for an incident that does not exist) essentially increases response time and resource availability for future incidents. Moreover, crowdsourced data is inherently noisy and uncertain, with the types of uncertainty spanning across multiple dimensions. For example, consider user posts reporting an incident on an application such as Waze, where users can notify about observed traffic accidents. There are several challenges with such reports. First, no notification or input is guaranteed to be correct; since users notify through the smartphone app while driving, it is conceivable that the notification could be triggered erroneously. Second, the time and location at which a user posts reports (which are automatically extracted when a button on the smartphone app is pressed) are typically noisy. This noise is added due to the difficulty in practice for a driver to press a button on his/her phone as soon as s/he sees an accident. Depending upon the traffic conditions and the speed at which s/he is driving both spatial and temporal noise are added to the user input [25]. Therefore, tools to aid emergency responders using such data must be able to extract the correct spatial and temporal parameters with certainty before dispatching resources.

Let us consider a scenario from the dataset we use to explain the context and the motivation for our tool design. Figure 1 shows three separate road incidents in Davidson County, Tennessee, United States between September 2019 and October 2019. Incident 1 (I_1) and incident 2 (I_2) show two junctions where road accidents occurred (the actual locations are shown by red triangles). In each case, we observe that several passer-bys used the Waze

[2]https://www.waze.com/

(a) I_1 – Incident 1 (b) I_2 – Incident 2 (c) I_3 – Incident 3

Fig. 1. Three sample incidents indicating official road accident report (red triangle) and Waze reports (circles) within a period of −30 *mins* and +30 *mins* of that incident. The color of the circles indicating Waze reports show the difference in time with the official report.



(a) (b)

Fig. 2. (a) Heatmap of the percentage of Waze reports surrounding an incident report. Waze reports and the incident reports through official channels are discretized into square-shaped grids of size 1 *km* and sampled at 5 *minute* frequency. The percentage shows Waze reports that were posted within 30 *minutes* prior to the incident report time. (b) Histogram of the time difference between each official incident report and Waze reports observed inside the same grid. $\Delta Time$ is the difference in time between reports.

application on their phones to report the accidents before the official incident report (shown by circles, with the time of reporting denoted by the color gradient). Moreover, we observe that the crowdsourced inputs on Waze were reported no more than 100 *meters* from the incident. However, we also observe the presence of both spatial and temporal noise in the reports (among themselves as well as with the actual incidents). Our analysis also revealed incidents with considerably higher volumes of noise. As a result, agencies that seek to dictate and optimize dispatch by using crowdsourced data must deal with the uncertainties introduced by the manner in which such information is retrieved. We specifically highlight the following types of uncertainty.

Spatial Uncertainty: Figure 2a illustrates the percentage of Waze reports from Davidson County in 2019 that were posted within 30 *minutes* prior to the time on which an actual incident was reported. We discretize both reports (i.e., crowdsourced and official) in space and time for simplicity. We divide the space into grids of length 1 *km* on each side in the above analysis. This analysis considered a total of 51, 415 Waze reports and 8, 912 reported incidents (details in Section 5.1.) The location of the Waze report is based on the relative position to an actual

incident, which is shown in the cell at the center of the grid $(2, 2)$ in Figure 2a. We observe that most Waze reports (close to 56%) are actually from the same cell as the actual incident. However, this observation also means that about 44% of the reports come from cells that introduce spatial uncertainty in the data.
Temporal Uncertainty: Figure 2b shows the histogram of the time difference between each official incident report and Waze reports observed in the vicinity of the incident (denoted by a square cell). We observe that while the majority of the Waze reports appear after the official incident report (indicated by negative $\Delta Time$), a significant number of reports are made before the official reporting, which denotes that such information can be used to detect incidents proactively to expedite response. We also observe the spatial noise in the data, which makes it imperative that practitioners incorporate methods to make inferences while considering this uncertainty.

In this paper, we specifically look at the feasibility of developing a tool to detect emergency incidents such as traffic accidents by using the non-traditional source of crowdsourced data from Waze. The proposed tool demonstrates the effectiveness of AI techniques in helping government agencies meaningfully explore such data sources and presents a promising direction to design and leverage human-centered AI tools for various operational goals and policy-making in the future. A human-centered AI design [27] aims to facilitate human control on an automated system to augment rather than replace human capabilities.

Considering the use-case of emergency management services, we show that using crowdsourced data processed with AI modeling techniques can expedite the dispatch of emergency responders. However, we point out that naively maximizing the accuracy of the incident detection model can hamper practitioner-centric factors as per their performance needs (we explain such parameters below). Hence, we suggest adopting a human-centered approach for model selection. Given that looking at a large number of models manually takes a lot of effort and to alleviate this burden on practitioners and policymakers, we propose limiting the number of "most helpful" models for a situation to a manageable level. This entails optimizing a multi-objective function that takes into account both practitioner-centric metrics and model performance-centric metrics like accuracy. The proposed human-centered AI tool provides greater flexibility for practitioners and policymakers to select a model to support emergency response by narrowing down the models with the best prediction capability at the desired resolutions for time and space. Thus, while using such tools for assistance, it is the responsibility of the practitioners or policymakers to select the final model to deploy that could be used by their agency's incident detection process ultimately. Moreover, to facilitate this human-centered model selection process, the proposed tool offers a user-friendly interface that presents visualizations of the model performance metrics and predictions based on simulated scenarios.

Paper Organization: The rest of the paper is organized as follows. Section 2 provides a review of related work. Section 3 details the problem formulation of incident detection and Section 4 describes the approach taken in designing the proposed human-centered AI tool for incident detection. Section 5 presents the experimental evaluation study that was carried out to demonstrate the effectiveness of the proposed human-centered approach in facilitating model selection for different incident detection modeling techniques. Section 6 shows the results of our experiments and a discussion of those results. Finally, Section 7 presents the concluding remarks.

2 BACKGROUND AND RELATED WORK

Emergency response has traditionally focused on incident prediction, resource allocation, and dispatch [18]. While the focus on incident detection is rather recent, literature on incident detection has been covered in studies relating to different domains of government services, such as emergency management and disaster response [21], and traffic management [2]. In this research, we will focus on studies related to traffic incidents such as road accidents. As discussed in Section 1, traditional incident detection is performed through official forms of information gathering, such as 911 calls in the United States. Smarter forms have been introduced, such as incorporating automatic name and location information (ANI/ALI) [4]. However, a call must be made explicitly by a person involved in the incident or a third party to initiate the process. A large and growing body of literature has investigated road incident detection with non-traditional sources as solutions to the limitations of traditional methods, such as the time taken for the agencies to detect an incident from its occurrence. Several forms of non-traditional data for incident detection include social-media data and crowdsourced data from sources like Twitter, Waze, and Foursquare. Prior research has shown the feasibility of using such non-traditional data for emergency response [25] and traffic management [2].

Recent studies have focused on using data from Waze to perform analysis on possible detection of incidents. Zhang et al. [31] identified that it is possible to detect secondary crashes (crashes due to non-recurrent congestion originating from primary crashes) with the help of Waze reports. Moreover, Amin-Naseri et al. [2] demonstrated that Waze reports could detect incidents earlier compared to traditional methods with broad coverage and reasonably accurate localization. Additionally, Li et al. [15] have shown the importance of combining traditional data sources (police reports) with non-traditional data (Waze reports) to identify high-risk road segments. In their study, the authors observed that 60.24% of road segments received higher Waze reports than police reports. This additional data helped them identify 14 miles of high-risk road segments, whereas they found only 8 miles of high-risk road segments with just police reports. Lenkei [14] compared the accident Waze reports and the traffic database in Sweden (Trafikverket), and they indicated that Waze could detect 27.5% of the incidents sooner than the reported times. Senarath et al. [25] presented a Bayesian information fusion approach to identify incidents from multiple crowdsourced Waze reports. Moreover, they demonstrated early incident detection using Waze reports compared to official incident reports.

Artificial agents are increasingly being employed in various government services such as predicting climate change, incidents, earthquakes, and flu outbreaks [7, 8, 28]. Various techniques that rely on machine learning and deep learning have become popular choices for analyzing and generating valuable insights for government agencies [7, 8, 16]. Gupta et al. [7] utilized sentiment analysis to aid the government and healthcare sectors in their COVID-19 planning efforts by constructing models that could predict the number of cases. Liu and Tang [16] proposed an AI-based real-time forecasting method that has the potential to provide early warnings of the government's economic situation. In recent times, there have been efforts to integrate crowdsourced data to assist government agencies. For example, Reynante et al. [22] introduced a crowdsourcing-based theoretical framework for addressing civic issues. Moreover, efforts have been made to include citizens in democratic decision-making processes such as in the work by Arana-Catania et al. [3]. In contrast, in this work, we propose a human-centered model selection process to choose an incident detection model with good performance and the ability to meet practitioner-centric requirements, such as accurate incident localization.

There have been relatively fewer studies on proposing a human-centered AI tool to support government services for detecting incidents from non-traditional data sources. Citizen-Helper[3] [20], AIDR[4] [9], and Dataminr[5] are some of those tools that are designed to be used in emergency management domain to detect and analyze emergency events using non-traditional data sources like crowdsourced data. While these tools have the capability to support incident detection from real-time data from non-traditional sources, it is still challenging for practitioners to understand and evaluate the value of different AI models with such tools for the goals of incident detection. Recently, Senarath et al. [24] introduced a practitioner-centric method for selecting optimal AI models for incident detection and identifying the best models based on different variables relevant to a practitioner. However, their proposed approach does not facilitate an interactive mechanism to keep the practitioner in the loop during optimal model selection. In this paper, we extend their approach–CROME framework, by facilitating an

[3]https://citizenhelper.orc.gmu.edu/
[4]http://aidr.qcri.org/
[5]https://www.dataminr.com/

interactive, human-centered AI tool for supporting practitioners in selecting their choice of incident detection model dynamically.

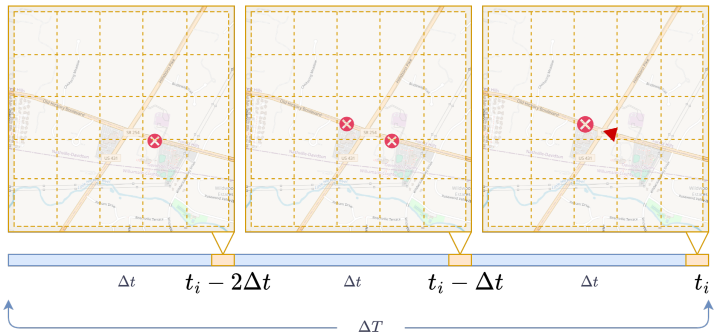


Fig. 3. An example scenario indicating the variables involved in the problem. The circles with cross marks indicate Waze reports and the triangle indicates the incident associated with those Waze reports.

3 PROBLEM FORMULATION

We begin by providing an example scenario to illustrate the problem. The Figure 3 shows an incident similar to the example incidents provided in the introduction but differs in that the three temporal snapshots show only one incident (shown by a triangle) and its associated crowdsourced Waze reports (shown by red circles) across different time-steps. While incidents (and their corresponding reports) occur in real-time (i.e., in continuous time), we discretize the problem for ease of exposition. We refer to each discrete time interval as time steps (of duration Δt). Moreover, at an arbitrary time step (t_i), we assume access to a set of reports (W_i) from a non-traditional data source like Waze. Typically, these reports contain information about the approximate location of the incident using two variables: latitude (l_w) and longitude (g_w). To simplify the problem further, we do not directly use this coordinate system; instead use a grid-based (G) system with squares of side length Δs as indicated in Figure 3 to represent the location with (x_w, y_w) denoting the coordinates along the two axes. While simplifying the coordinate system can reduce incident localization accuracy, one could always discretize space fine enough to match the specific requirements for a use-case. The variable ΔT shows the total time before the time of analysis which determines the number of time steps considered for modeling. The triangle in Figure 3 shows the official incident report. Similar to the reports from the non-traditional data source (Waze), we identify the grid and the time step for the official incident report to be used in incident detection model training. Based on different values of Δs and Δt, we can prepare different incident detection models that can detect the likelihood of incidents at different spatial and temporal resolutions. Let such an incident detection model be represented by $M(\Delta s, \Delta t)$. Our problem then reduces to optimally selecting the hyper-parameters Δs and Δt while maximizing the accuracy (or a similar quantifiable metric) of the model M.

The first part of our tool development is to create a model M that can determine the likelihood of incidents in each grid cell by examining noisy reports that are within a certain spatiotemporal vicinity. This is related to the idea of events in the Shannon's Information Theory. However, Shannon entropy is utilized for measuring the uncertainty of the incident occurrence and does not reveal much about the accuracy of the detection process. As a result, we focus on training a model to detect the likelihood score on each cell conditional on relevant features based on ground truth labels retrieved from historical data. This training process may however implicitly improve on uncertainty by penalizing the incorrect estimations through the loss function, forcing the algorithm to learn a function that produces high scores for cells where the incident is most likely to have occurred.

Problems: Specifically, we focus on the following problems: **1)** Detect road incidents proactively through non-traditional crowdsourced data sources such as Waze before an official report arrives, and provide a tool for their real-time visualization.; **2)** Determine and present the best localization parameters through an interactive visualization that can be used for tool deployment as per the choice of a practitioner user based on his/her role. Our objective is to offer greater flexibility to the end-user, rather than restricting them to a single model with optimal (model) performance, allowing them to have more control over the model selection process in alignment with human-centred AI principles.

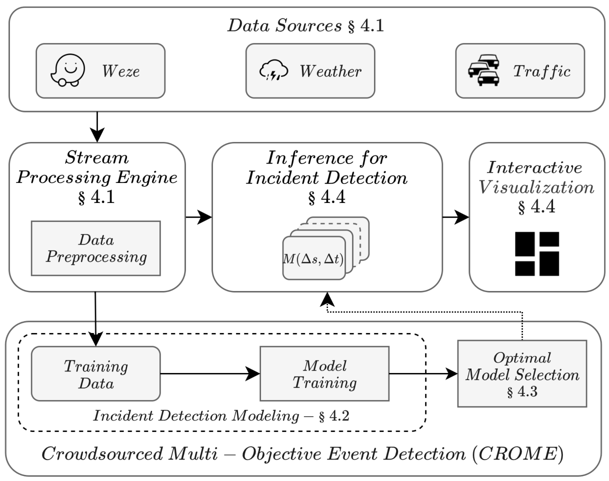


Fig. 4. The high-level architecture of CROME*x* human-centered AI tool for learning models, selection of best model, and detecting incidents. Sections numbers (indicated as §) correspond to the subsection in Section 4
discussing that component.

Data: Input data D_t for training, Validation data D_v for scoring
Result: M_O
$M_A \leftarrow \varnothing$;
for $\Delta\ t \in T$ **do**
  for $\Delta\ s \in S$ **do**
    $D_t' \leftarrow$ Discretize(D_t, $\Delta\ s$, $\Delta\ t$) ;
    Model $\leftarrow$ CreateModel(D_t') ;
    $D_v' \leftarrow$ Discretize(D_v, $\Delta\ s$, $\Delta\ t$) ;
    Score $\leftarrow$ Evaluate(Model, D_v') ;
    $M_A \leftarrow M_A \cup \{(\Delta\ s, \Delta\ t, \text{Score}, \text{Model})\}$;
  end
end
$M_O \ \leftarrow\ ParetoOptimize(M_A)$;

Algorithm 1: Algorithm for training models in CROME and getting the best models. The symbols utilized are defined in Table 1.

Table 1. Table with symbols utilized in the Algorithm 1.

Symbol	Definition
T	A set of pre-defined temporal resolutions to train and evaluate.
S	A set of pre-defined spatial resolutions to train and evaluate.
M_A	Set of all the models. Each identified with tuple ($\Delta\ s$, $\Delta\ t$, model score, trained model parameters)
M_O	Set of optimized models.
$\varnothing$	Empty set.
$Discretize(\cdot)$	Spatiotemporal discretization function that takes data points (a set of observations with longitude, latitude and time attributes) and output matrixes that represents the input.
$CreateModel(\cdot)$	Model training function that returns the model (with trained parameters).
$Evaluate(\cdot)$	Evaluation function that takes the model and validation data and outputs score value (e.g., F1-score).
$ParetoOptimize(\cdot)$	Multi-objective optimization algorithm that takes in a set of all parameter tuples to optimize and return a optimal subset of parameter tuples.

4 APPROACH

Figure 4 shows the architecture of *CROMEx*, the proposed human-centered AI tool. We extend the approach identified in CROME [24] and provide more control by providing a variety of visualizations to interpret the behavior of the "most helpful" models for the practitioners to compare and select the best models for deployment. We show the high-level algorithmic approach outlined by CROME in Algorithm 1. The tool comprises of several components, which we describe in the following subsections. By following best practices for human-centered AI system design such as modularity, extensibility, interactivity, and adaptability, these components can be extended to meet end-user requirements [26].

4.1 Data Sources and Stream Processing Engine

While a multitude of data sources can be used for incident detection, we use Waze, a crowdsourced application that enables users to share information about traffic and accidents (among others). We also use weather and traffic information for road accident detection. The data provided by these means are streamed in real-time and obtained using application programming interfaces (APIs), databases, and scraping. Furthermore, extending this module to support other data sources is possible. Since different data sources may provide data in different formats, this module enables data transformation to support the rest of the incident detection process. Specifically, it can extract the relevant information and produce a *JSON* formatted object with metadata, including mandatory fields *longitude*, *latitude*, and *time*. We may include other parameters such as the *reliability* of a Waze report when the modeling process requires it. Then the observed reports from data sources are pre-processed to form a matrix of features for each grid cell of grid system G for each time step as defined in Section 3. Specifically, this feature matrix is computed based on three sources of features for each grid: reliability and presence of Waze reports; the amount of precipitation from the nearest weather station; and mean traffic congestion. Moreover, to capture the volume as well as the reliability of the crowdsourced Waze reports, we generate the following three features from the Waze reports: 1) Volume: the total number of crowdsourced reports, 2) Sum of Reliability: the sum of the reliability scores of the crowdsourced reports, and 3) Mean of Reliability: the average reliability score of the crowdsourced reports.

4.2 Incident Detection Modeling

The next step of our process is to build the model(s) to be used in the incident detection process. Here we treat incident detection as a multi-label classification task that tries to identify the grids with incidents at the end of each time step. We approximate a statistical model $M(\Delta s,\ \Delta t)$ for varying values of Δs and Δt using deep learning and traditional machine learning techniques. The model's output is a mapping of the grid to the presence of an incident at a given time step.

We leverage the structure of the incident detection problem while choosing a model that can draw inferences from spatiotemporal crowdsourced data. Intuitively, we try to capture the proximity of observation and reporting. After users observe an incident, they move in space, and a certain amount of time elapses before they report it. However, it is natural to assume that the displacement (both spatial and temporal) is not very high; it is unlikely that users move many miles and report an incident hours after they observe it. Therefore, reports generated in a cell (say $g_i \in G$) could have been observed in other cells that lie close to g_i. To leverage the spatial structure of the data, we use a Convolutional Neural Network (CNN) [6] to build the incident detection model, and the details about it and other alternative models (Bayesian Information Fusion, and K-nearest neighbor) will be provided in Section 5.2.

4.3 Optimal Model Selection

As we highlighted before, we are interested in not only maximizing the detection accuracy of the proposed pipeline but also improving spatial and temporal localization. The very structure of our problem formulation dictates that no single solution necessarily optimizes all the objectives simultaneously. Indeed, we show through experimental evaluation that as the spatial resolution increases, the accuracy of the model degrades. One straightforward solution to this problem is employing a weighted sums approach, where we provide weights for each objective and take the sum and optimize the resulting value. However, this method necessitates prior manual investigation of model performance. To alleviate such involvement, we seek to find a set of Pareto-optimal solutions. A solution is called Pareto-optimal or non-dominated if none of the objective functions can be improved further without sacrificing the value of at least one of the other objective functions [5]. Therefore, for a given model, the Pareto optimal solution is one where an improvement in one performance metric cannot be achieved without a trade-off resulting in a reduction in at least one other performance metric. To find the Pareto optimal solution, we use a multi-objective evolutionary algorithm [5] based on the concept of ϵ-dominance [13]. The idea of ϵ-dominance maintains a well-distributed set of non-dominated solutions by not allowing two solutions with a difference less than an exogenously specified threshold (ϵ_i in the i^{th} objective) to be non-dominated to each other. For a detailed description of how such an approach can be used for multi-objective optimization, we refer readers to the work by Deb et al. [5]. If any important metric is identified later, it can be seamlessly incorporated into the optimal model selection process by adding it as an input feature to the multi-objective optimization algorithm.

4.4 Incident Detection and Interactive Visualization

The *incident detection module* uses the optimal models obtained through the CROME approach to detect incidents of interest in real-time from streaming data. The practitioner can select the best model suited for the situation. The *interactive visualization component* presents the identified incidents along with the input data. Furthermore, this component shows the performance of (optimal) models in an interactive environment for the practitioners to analyze and select the best-performing model for their use case. The interactive visualization component

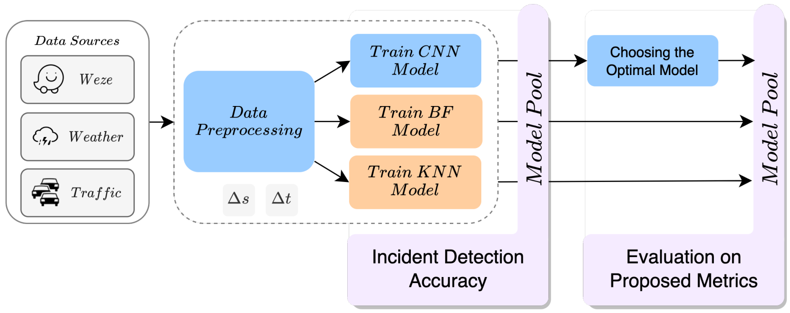


Fig. 5. Overview of the implementation process to experiment with and choosing alternative models in the *CROMEx* tool.

contains three main functionalities: **1)** To help the practitioner explore the training data and identify the optimal hyper-parameters for space and time resolutions; **2)** Explore the model optimization process and identify the best-suited model for predicting incidents; **3)** Show the detected incidents in real-time using the streaming data.

5 EXPERIMENTAL EVALUATION

This section will cover the data sources, experimental setup, modeling approaches, and measurements utilized for the experimental evaluation. We use the experimental setup discussed in this section to train and test the models with the help of the identified data sources. The measurements presented in this section are solely employed to evaluate and compare the differences between models to study their behavior. The experimental setup section elaborates on the model training specifications of the CROME approach that primarily relies upon CNN model. Notably, this method first generates models for several spatial and temporal resolutions and subsequently refines to a manageable number of "most helpful" models to be further investigated by a practitioner for deployment. In the results section, we compare CNN models optimized using CROME to two other incident detection models, as discussed below. In Figure 5, we showcase a flow diagram depicting the model training and evaluation process. The components associated with CROME are visually emphasized in blue. Notably, the processes within the dashed box are executed multiple times for each Δs and Δt to obtain multiple models for comparisons.

5.1 Data Platforms

We briefly describe the data that we use for analysis.

- **Crowdsourced Data**: Waze is a GPS navigation application and crowdsourcing platform [29]. We look at user reports concerning roadway accidents from Sep/01/2019 to Dec/31/2019. The selection of this period is based on data availability.
- ***Ground-Truth* Incident Data**: We collect accident data from the public safety office of a large metropolitan area in the USA, with a size of about 500 sq. miles. We consider such incidents as the ground-truth. It is certainly possible that some accidents were not reported to the public safety office. However, it is not possible to consider such reports for analysis since their occurrence remains unknown. To remove noise, we map each incident to its closest roadway segment (typically at a distance of less than 25m). We collect information about roadway geometry through INRIX [1], a private entity that provides location-based data.
- **Traffic Data**: Traffic information is known to be one of the most important determinants of accident forecasting [18]. We collect roadway traffic data in a time resolution of 5-minute intervals for the area under consideration, resulting in approximately 270 million measurements. We utilize the congestion information from this data as input for our modeling process.

- **Weather Data**: We collect weather information from Weatherbit [30]. We collect data about the precipitation from all stations that lie in a 100 sq. mile radius from the center of the spatial area under consideration.

5.2 Setup

We collect crowdsourced data from the Waze platform for the period between Sep 01, 2019 to Dec 31, 2019. We divide the data between training sets of three months and a test set of one month in a manner that each month is used as the test set in an independent evaluation. The reason for dividing the data in this specific manner, rather than opting for a random set of training incidents, is to ensure there are no temporal overlaps between the training and testing datasets. Hyper-parameters such as the number of epochs and threshold for classification pertaining to the CNN are tuned through k-fold cross-validation. The incident detection models were built on a system with Red Hat Enterprise Linux as the operating system with Tesla K80 GPU. The total memory allocated for model training was 32 *GB* per model. The models were trained in parallel to reduce the overall training time. Importantly, inference time for the CNN model-based CROME approach is in the order of microseconds which will be helpful in proactive incident detection.

5.3 Modeling Approaches

To compare the performance of CROME (with CNN as the modeling approach), we implement two alternative approaches: Bayesian Information Fusion (BF) and K-Nearest Neighbor (KNN). We pick the alternative approaches based on a literature survey of incident detection and forecasting. Our first alternative approach, based on Bayesian Information Fusion [25] seeks to detect traffic accidents based on crowdsourced data. To the best of our knowledge, it is the current state-of-the-art in this domain. We choose the second alternative approach based on prior work done in the domain of incident forecasting. While incident detection is a relatively newer problem (in comparison to forecasting), we hypothesize that some approaches to forecasting can be modified to aid detection. As a result, we adopt an approach to incident forecasting based on the well-known k-nearest neighbor (KNN) model [17]. We describe the modeling approaches below:

- **Convolutional Neural Network [CNN]** - We implement two convolutional layer neural network with a max-pooling layer in between. The last convolutional layer is followed by a ReLU activated layer and a sigmoid layer. The CNN architecture enables us to consider how alerts generated in a cell are correlated with incidents that occur in its spatial proximity. We use 256 filters with a filter size of 2×2 for both convolutional layers. The size of the max-pooling layer is set to 2×2. The last two dense layers contain $2 \times N_o$ and N_o units accordingly, where N_o is the number of outputs that show the presence of incidents in each grid.
- **Bayesian Information Fusion [BF]** - Senarath et al. [25] propose a Bayesian-theoretic approach to detect spatial-temporal incidents based on crowdsourced data. They employ a combination of spatial-temporal segmentation and clustering (DBSCAN [23]). Incident detection and localization are inferred after learning posterior distributions over relevant variables conditional on historical incident data, crowdsourced data, and other determinants like traffic and weather. We implement their approach and use logistic regression to identify the optimal decision boundary for detecting incidents since it had the best performance in their work. We only modify the aggregation function over the crowdsourced reports according to the problem specification. Note that we *do not* choose a specific aggregation function based on our approach; we vary the hyper-parameters uniformly for all models.
- **K-Nearest Neighbor [KNN]** The k-nearest neighbor (KNN) has been used extensively to forecast the occurrence and duration of traffic incidents [12, 17]. The KNN method can be naturally extended for incident detection. We use the same features as mentioned in Section 4 and use the $l2$-norm to determine

proximity in the feature space. During inference, for a given crowdsourced report (say w_i), we first calculate the set of its k nearest neighbors (say N_i) in the feature space. Then, we use a majority voting scheme using the outputs of the members of N_i to assign a label to w_i.

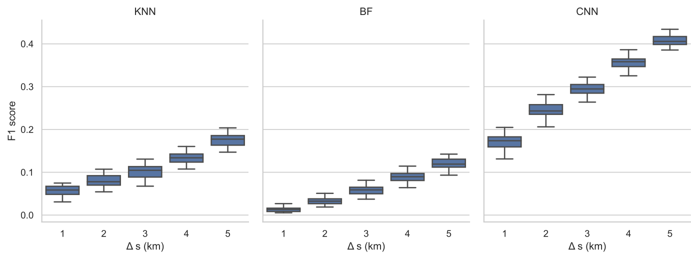
5.4 Evaluation Measures

We utilize the following metrics to evaluate the performance of the incident detection models. The metrics provided below are defined based on their significance to the first responders and practitioners. Using an assorted set of metrics can help policymakers in the decision-making process, given that the optimization problem we formulate has multiple objective functions.

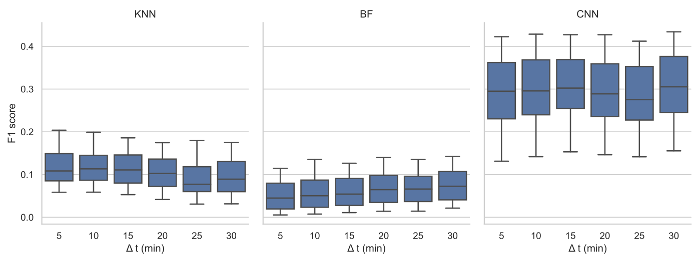
- **F-1 Score** - It is imperative that a model for incident detection is able to balance between precision (the fraction of correct detections among all detections) and recall (the fraction of correct detections among the total number of actual incidents). The F-1 score provides a harmonic mean of precision and recall measures. Emergency responders operate under limited resources, so the cost of false positives for an exogenous detector can be rather high. To deploy such a system in practice, it is important that responders are available when actual calls for aid are made, such as via 911 calls in the USA.
- **Average Early Prediction Ratio** - The fundamental goal of using crowdsourced data to detect incidents is to improve response times. In order to achieve that goal, it is crucial to correctly detect incidents *before* they are reported (before receiving an actual 911 call). Considering the inherent randomness in traffic accidents and the unreliability of individual crowdsourced reports, early prediction is non-trivial. While multiple reports can be accumulated over time, note that any mechanism to aid response can only be adopted if it consistently detects incidents before they are reported. Therefore, we measure the early prediction ratio, which is defined as the ratio of the total number of correctly early-predicted incidents to the total number of incidents reported.
- **Average Early Prediction Distance** - Our twin goals for the proposed framework are accuracy and localization. It is important that the detected incidents are close (in space and time) to the actual incidents so that first responders can dispatch resources efficiently. In order to measure localization, we calculate the geodesic distance between the center of the cell (in G) where the incident was detected and the location of the actual incident (obtained through the ground-truth report).
- **Average Early Prediction Time** - To measure temporal localization, we calculate the average difference between the reported time of the incident (obtained through ground-truth reports) and the time the incident is detected. This metric aims to measure how early (on average) a model detects an incident correctly.

6 RESULTS

6.0.1 Incident Detection Accuracy. We begin by comparing the detection accuracy of the CNN model (the modeling choice used in CROME approach) with respect to the alternative approaches (BF and KNN). We vary the spatial and temporal resolutions (Δs and Δt) for this comparison and present the results without using the Pareto optimization framework (we present complete results later). Our purpose in doing so is two-fold. We seek to examine the robustness of the models with respect to varying discretization parameters and also validate our hypothesis that detection accuracy can suffer as spatial and temporal resolutions increase. We present the results in Figure 6a and Figure 6b which show the influence of time and space resolution on the F-1 score. We have the following major findings: **1)** CNN significantly outperforms the alternative models in terms of F-1 score in all cases. **2)** As the spatial resolution increases and localization becomes more challenging, the F-1 score of all the models decreases. **3)** Surprisingly, temporal discretization does not affect the performance of any of the models (barring minor variations). We hypothesize that this is due to the effects of aggregating features over multiple time steps.



(a) Performance of models for different Δs.



(b) Performance of models for different Δt.

Fig. 6. Performance of models against different spatial and temporal resolutions using the test data sets. We observe that the CNN model significantly outperforms the other modeling approaches.

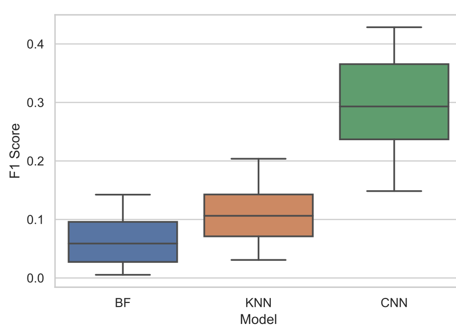


Fig. 7. Distribution of F1 score attained by different models on the test sets. The CNN model (part of CROME) outperforms the alternative approaches.

Next, to explore how CNN performs better in terms of F-1 score, we show aggregated F-1 scores (across all spatial and temporal resolutions). We present the aggregated results in Figure 7 and the distribution of precision and recall for all models in Figure 8. We observe that CNN outperforms the alternative models by a significant margin. We also observe that the performance is largely governed by the balance of precision and recall (see Figure 8); while the alternative methods result in higher precision *or* recall, they fail to balance the metrics to

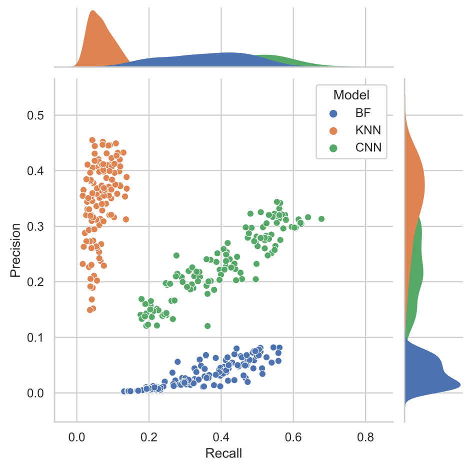


Fig. 8. Comparison plot of precision-recall for model training. We observe that the CNN model can balance precision and recall to achieve higher F-1 scores.

obtain higher F-1 scores. We show below that this balance is crucial for real-world scenarios. For example, the BF approach (Bayesian Information Fusion) has higher recall than the other models. However, while such an approach results in more number of successfully detected incidents, it also causes a significantly large number of false alerts, which is detrimental to emergency response in practice.

6.0.2 Choosing the Optimal Model. Having shown that the CNN model as part of CROME outperforms the alternative approaches (in terms of F-1 score), we now focus on evaluating CROME in its entirety. Recall that our goal is to simultaneously aid detection and localization by solving optimization problem. We first show how calculating the Pareto frontier helps us solve optimization problem (see Figure 9). The plot shows the three dimensions over which CROME optimizes, namely the spatial resolution (Δs), the temporal resolution (Δt), and the F-1 score. Each point in the three-dimensional plot represents a specific *learned* model. We show the non-dominated set in red. Notice that the non-dominated set consists of several learned models. We leave the final choice of selecting one (or more) models for deployment from the non-dominated set to the practitioners based on the relative importance of the specific objective functions (f, Δs, and Δt) for their needs.

6.0.3 Evaluation on Proposed Metrics. We now have access to a single solution per metric of interest; we obtain such a solution by sorting the non-dominated set based on a relevant criterion and choosing the model that maximizes (or minimizes) the corresponding objective function. This single learned model is the output of CROME that we compare with the alternative models in terms of early detection and spatial and temporal localization. We present the results in Table 2. In order to make a fair comparison with the alternative approaches, we pick specific instances of the alternative models that optimize the metric of interest. For example, while evaluating spatial localization, we choose the BF model that minimizes average prediction distance on the validation set.

Our key findings are as follows: **1)** While BF outperforms CROME in terms of early detection, 85% of the alerts that it generates are incorrect (false positives). Using such an approach is infeasible in practice as responders cannot be dispatched based on incorrect alerts. CROME, on the other hand, balances precision and recall to detect more than 40% of the incidents early. **2)** CROME performs nearly on-par with the alternative approaches in terms of localization. Indeed, on average, it detects incidents within 0.62 kilometers of the best-performing model on spatial localization (BF) and within 1.79 minutes of the best-performing model on temporal localization

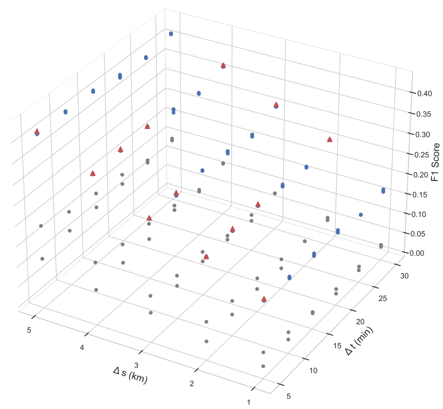


Fig. 9. The non-dominated set of solutions obtained through CROME (in red). All the learned CNN models are shown in blue. For the sake of comparison, we also show the performance of alternative models (in gray).

Table 2. Early detection performance of models. The F1 score is evaluated on the test set. The CROME approach's model used here is based on CNN.

Model	Δs (km)	Δt (min)	F1 Score	Early Pred %	Avg Distance (km)	Avg. Early Time (min)	Precision	Recall
				Best Early Pred %				
BF	1	5	0.60	**77.56**	3.27	15.02	0.00	0.14
KNN	5	5	19.71	17.72	2.99	14.14	0.36	0.14
CROME	5	5	**41.00**	40.28	2.96	13.94	0.32	0.56
				Best Avg. Distance				
BF	5	5	10.56	35.00	3.16	15.02	0.06	0.33
KNN	1	20	5.37	1.21	**2.05**	11.63	0.32	0.03
CROME	1	5	**16.41**	18.35	2.67	14.32	0.16	0.18
				Best Avg. Early Time				
BF	3	20	6.49	47.69	3.25	15.45	0.04	0.40
KNN	3	15	11.25	4.32	2.75	**16.03**	0.38	0.07
CROME	3	30	**30.40**	24.67	3.03	14.92	0.23	0.45

(KNN). Also, no alternative modeling approach outperforms CROME in both types of localization. **3)** We find that while tailoring models to specifically maximize precision or recall can maximize certain metrics associated with incident detection, the lack of consideration of practitioner-specific parameters can lead to detrimental consequences. For example, consider the KNN model that results in the best spatial localization (row 5 in Table 2); despite slightly improved localization, it fails to generate alerts for 86.5% of the incidents.

We point out the need to consider a combination of F-1 score and the metrics pertaining to detection and localization. Consider a model that generates alerts at all time steps on all cells. Such a model would detect all possible incidents early with remarkable localization (since localization is measured with respect to the ground-truth incidents). As a result, the balance of precision and recall is crucial in this setting. Based on Table 2, we conclude that the CNN model-based CROME approach, which seeks to include practitioner-centric parameters in selecting incident detection model, results in a significantly higher F-1 score, a significantly lower number of false alerts, and competitive spatial and temporal localization.

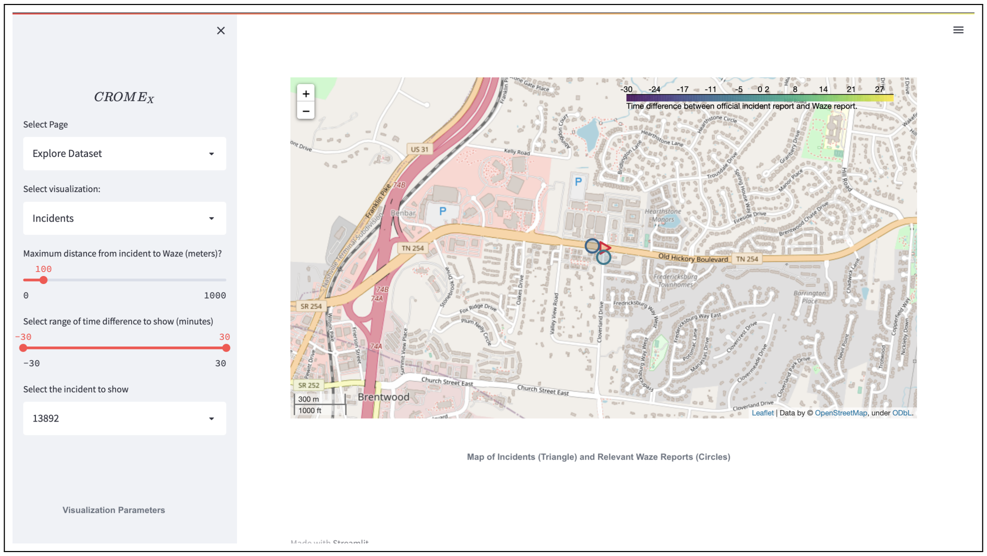


Fig. 10. Interface of the *Data Explorer* page of the dashboard of the proposed tool. This interface will help the practitioner to understand the input data source (Waze) and compare it with the actual incidents interactively.

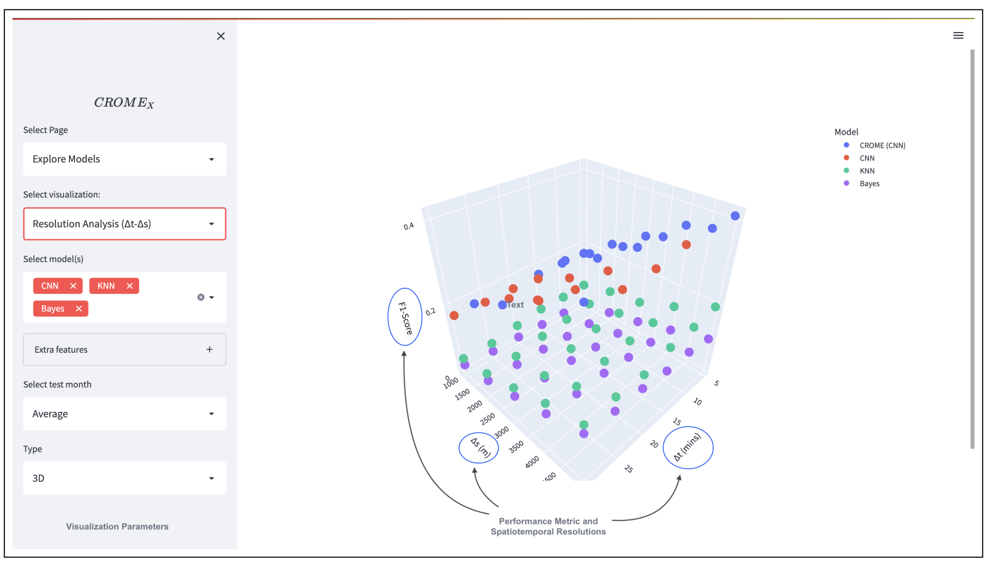


Fig. 11. Interface of the *Model Explorer* page of the dashboard of the proposed tool. This interface can be used by the practitioner to identify the best models out of all pre-trained models by comparing the performance metrics and .

6.0.4 Human-centered AI Dashboard. Our end goal is to ensure the approach highlighted in CROME*x* (Figure 4) can be used by practitioners for incident response. However, simply providing metrics that are popular among data scientists and machine learning professionals can inhibit the deployment of such approaches in practice. We present an open-source and practitioner-friendly tool that can be used for helping deployment. The tool can

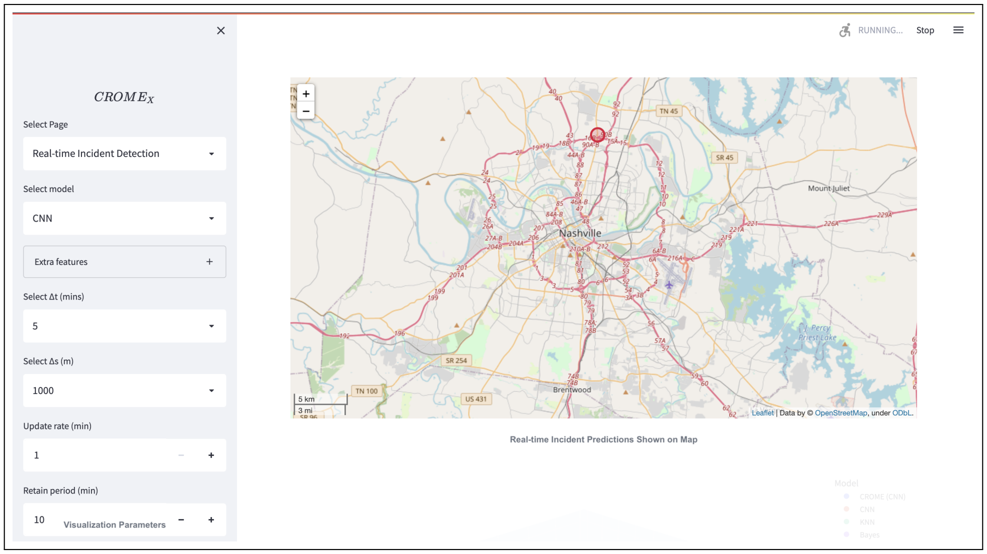


Fig. 12. Interface of the *Real-time Incident Detection* page of the dashboard of the proposed tool. This interface simulates a real-time incident detection process using a selected pre-trained model with the help of past data for practitioners to understand the model's behavior in practice.

be used to analyze and understand the behavior of different models to select one to be used in the deployed application /production. We present the design of different use-cases of the proposed tool below. As identified in Section 4, we have three different use-cases of the dashboard.

Case 1: Figure 10 shows the interface to help practitioners explore the training data and identify the optimal hyper-parameters. The user can use this tool to change variables like the distance between the incident and Waze reports and obtain different detection distributions. Moreover, we present the capability for the user to not only be able to visualize the individual incidents and relevant Waze reports but also able to analyze the dataset. We hypothesize that such a feature could help practitioners gauge whether deployment is feasible in different geographical areas or not (the frequency, volume, noise, and uncertainty associated with crowdsourced reports vary across geographical areas).

Case 2: The dashboard interface for model analysis is provided in Figure 11. This interface indicates the performance of pre-trained models and provides interactive controls to filter by the hyper-parameters used in the training. The axis of the chart in Figure 11 illustrates the performance metric and spatiotemporal resolution levels. Moreover, it showcases the results of the optimal set of models with the chosen hyper-parameters in a different color (blue). The practitioners can use this information in the process of deciding which model to employ in practice.

Case 3: The interface indicated by Figure 12 shows the real-time incidents. In our tool, we are using data from the test month to simulate the real-time streaming data. Based on the decision of the use-case 2 above, a user can change the pre-trained model and see if the predictions of that selected model satisfy the user's needs.

6.1 Directions for Deployment

There are various challenges in deployment, including technological, and personnel-related. The proposed approach requires hardware infrastructure that runs in real-time to meet the technological requirements. The effectiveness of the tool should be evaluated by deploying in a simulation training process for emergency response with government agency practitioners. Accordingly, any additional metric that needs to be taken into consideration must be identified and implemented in the system. Training government agency practitioners on how to effectively use this tool and navigate through the various interfaces and controls will be necessary for successful deployment.

6.2 Reproducibility

We share our implementation for the proposed tool as well as the models through an online repository CROME. While the data we used for experimentation is proprietary (specifically the official accident data), we release a small random sample dataset that can be used to run our code. The repository also contains detailed instructions about how to run the code, instructions regarding how to infer the results, pre-trained models, and training runtime for each model.

7 CONCLUSION

This paper proposes a human-centered AI tool (CROME*x*) for incident detection that facilitates an interactive visualization dashboard for leveraging the capability of a multi-objective optimization approach for early detection of spatial-temporal incidents using crowdsourced data. We illustrate how crowdsourced data, historical ground-truth incident data, and an arbitrary set of features can be combined to detect potential incidents before they are reported officially. We also show how practitioner-centric parameters can be incorporated into our approach and controlled through an interactive tool interface. The proposed tool combines convolutional neural networks (CNN) for incident detection and evolutionary algorithm for solving the multi-objective optimization problem. We show that this approach outperforms state-of-the-art methods on several technical and practitioner-centric metrics through extensive evaluation using real-world data. Finally, we showcase how we can provide practitioners the ability to explore the data through the proposed tool by different visualizations of the datasets, AI model performance, and interactive simulation of incident detection using past data on incidents. Although the study's findings present useful insight about how well the proposed tool performs, there are still many areas that could benefit from additional research. One area of prospective investigation is the evaluation of alternative approaches for the different components of the tool, such as different algorithms for multi-objective optimization and other models for detecting and locating incidents. In particular, it would be interesting to compare the performance of these alternative approaches with the proposed tool and evaluate their ability to enhance the overall performance.

In summary, this paper demonstrates the feasibility of designing a human-centered AI tool to assist government agencies in meaningfully leveraging non-traditional sources of crowdsourced data for early incident detection to assist in improving decision-making for response.

ACKNOWLEDGMENTS

This project was supported by resources provided by the Office of Research Computing at George Mason University and funded in part by the National Science Foundation grants (1814958, 1815459, 1625039, 2018631) and a grant from Tennessee Department of Transportation.

REFERENCES

[1] 2022. INRIX. https://inrix.com/
[2] Mostafa Amin-Naseri, Pranamesh Chakraborty, Anuj Sharma, Stephen B Gilbert, and Mingyi Hong. 2018. Evaluating the reliability, coverage, and added value of crowdsourced traffic incident reports from Waze. *Transportation Research Record* 2672, 43 (2018), 34–43.
[3] Miguel Arana-Catania, Felix-Anselm Van Lier, Rob Procter, Nataliya Tkachenko, Yulan He, Arkaitz Zubiaga, and Maria Liakata. 2021. Citizen participation and machine learning for a better democracy. *Digital Government: Research and Practice* 2, 3 (2021), 1–22.
[4] Ciy of Rochester. 2019. How 911 works - what happens when you dial 911. https://www.cityofrochester.gov/article.aspx?id=8589935579
[5] Kalyanmoy Deb, Manikanth Mohan, and Shikhar Mishra. 2005. Evaluating the ε-domination based multi-objective evolutionary algorithm for a quick computation of Pareto-optimal solutions. *Evolutionary Computation* 13, 4 (2005), 501–525.

- [6] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. 2016. *Deep Learning*. MIT press Cambridge.
- [7] Rajan Gupta, Gaurav Pandey, Poonam Chaudhary, and Saibal K Pal. 2020. Machine learning models for government to predict COVID-19 outbreak. *Digital Government: Research and Practice* 1, 4 (2020), 1–6.
- [8] Trung Dong Huynh, Niko Tsakalakis, Ayah Helal, Sophie Stalla-Bourdillon, and Luc Moreau. 2021. Addressing Regulatory Requirements on Explanations for Automated Decisions with Provenance—A Case Study. *Digital Government: Research and Practice* 2, 2 (2021), 1–14.
- [9] Muhammad Imran, Carlos Castillo, Ji Lucas, Patrick Meier, and Sarah Vieweg. 2014. AIDR: Artificial intelligence for disaster response. In *Proceedings of the 23rd international conference on world wide web*. 159–162.
- [10] Henrik Jaldell. 2017. How important is the time factor? Saving lives using fire and rescue services. *Fire Technology* 53, 2 (2017), 695–708.
- [11] Henrik Jaldell, Prachaksvich Lebnak, and Anurak Amornpetchsathaporn. 2014. Time is money, but how much? The monetary value of response time for Thai ambulance emergency services. *Value in health* 17, 5 (2014), 555–560.
- [12] Li Kuang, Han Yan, Yujia Zhu, Shenmei Tu, and Xiaoliang Fan. 2019. Predicting duration of traffic accidents based on cost-sensitive Bayesian network and weighted K-nearest neighbor. *Journal of Intelligent Transportation Systems* 23, 2 (2019), 161–174.
- [13] Marco Laumanns, Lothar Thiele, Kalyanmoy Deb, and Eckart Zitzler. 2002. Combining convergence and diversity in evolutionary multiobjective optimization. *Evolutionary computation* 10, 3 (2002), 263–282.
- [14] Zsolt Lenkei. 2018. *Crowdsourced traffic information in traffic management: Evaluation of traffic information from Waze*. Master's thesis. KTH, Transport Planning, Economics and Engineering.
- [15] Xiao Li, Bahar Dadashova, Siyu Yu, and Zhe Zhang. 2020. Rethinking Highway Safety Analysis by Leveraging Crowdsourced Waze Data. *Sustainability* 12, 23 (2020), 10127.
- [16] Yisheng Liu and Anying Tang. 2022. Prediction Method of Government Economic Situation based on Big Data Analysis. *Digital Government: Research and Practice* 3, 4 (2022), 1–16.
- [17] Yisheng Lv, Shuming Tang, and Hongxia Zhao. 2009. Real-time highway traffic accident prediction based on the k-nearest neighbor method. In *Conference on Measuring Technology and Mechatronics Automation*, Vol. 3. 547–550.
- [18] Ayan Mukhopadhyay, Geoffrey Pettet, Sayyed Mohsen Vazirizade, Di Lu, Alejandro Jaimes, Said El Said, Hiba Baroud, Yevgeniy Vorobeychik, Mykel Kochenderfer, and Abhishek Dubey. 2022. A review of incident prediction, resource allocation, and dispatch models for emergency management. *Accident Analysis & Prevention* 165 (2022), 106501.
- [19] Ayan Mukhopadhyay, Zilin Wang, and Yevgeniy Vorobeychik. 2018. A decision theoretic framework for emergency responder dispatch. In *International Conference on Autonomous Agents and MultiAgent Systems*. 588–596.
- [20] Rahul Pandey and Hemant Purohit. 2018. Citizenhelper-adaptive: Expert-augmented streaming analytics system for emergency services and humanitarian organizations. In *2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*. IEEE, 630–633.
- [21] Hemant Purohit, Saideep Nannapaneni, Abhishek Dubey, Prakruthi Karuna, and Gautam Biswas. 2018. Structured Summarization of Social Web for Smart Emergency Services by Uncertain Concept Graph. In *2018 IEEE International Science of Smart City Operations and Platforms Engineering in Partnership with Global City Teams Challenge (SCOPE-GCTC)*. IEEE, 30–35.
- [22] Brandon Reynante, Steven P Dow, and Narges Mahyar. 2021. A framework for open civic design: Integrating public participation, crowdsourcing, and design thinking. *Digital Government: Research and Practice* 2, 4 (2021), 1–22.
- [23] Erich Schubert, Jörg Sander, Martin Ester, Hans Peter Kriegel, and Xiaowei Xu. 2017. DBSCAN revisited, revisited: why and how you should (still) use DBSCAN. *ACM Transactions on Database Systems (TODS)* 42, 3 (2017), 1–21.
- [24] Yasas Senarath, Ayan Mukhopadhyay, Sayyed Mohsen Vazirizade, Hemant Purohit, Saideep Nannapaneni, and Abhishek Dubey. 2021. Practitioner-centric approach for early incident detection using crowdsourced data for emergency services. In *2021 IEEE International Conference on Data Mining (ICDM)*. IEEE, 1318–1323.
- [25] Yasas Senarath, Saideep Nannapaneni, Hemant Purohit, and Abhishek Dubey. 2020. Emergency Incident Detection from Crowdsourced Waze Data using Bayesian Information Fusion. In *IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology*. 187–194.
- [26] Yasas Senarath, Rahul Pandey, Steve Peterson, and Hemant Purohit. 2022. Citizen-Helper System for Human-Centered AI Use in Disaster Management. In *International Handbook of Disaster Research*. Springer, 1–21.
- [27] Ben Shneiderman. 2020. Bridging the gap between ethics and practice: guidelines for reliable, safe, and trustworthy human-centered AI systems. *ACM Transactions on Interactive Intelligent Systems (TiiS)* 10, 4 (2020), 1–31.
- [28] Nicolas Spatola and Karl F MacDorman. 2021. Why real citizens would turn to artificial leaders. *Digital Government: Research and Practice* 2, 3 (2021), 1–24.
- [29] Waze. 2021. Driving Directions, Traffic Reports & Carpool Rideshares by Waze. https://www.waze.com/
- [30] Weatherbit. 2021. Weatherbit – Historical Weather API. https://www.weatherbit.io/
- [31] Zhihua Zhang, Yuandong Liu, Lee D Han, and Phillip Bradley Freeze. 2021. Secondary Crash Identification using Crowdsourced Waze User Reports. *Transportation Research Record* (2021), 853–862.