Learning Traffic Incident Detectors

Tomas Singliar
Department of Computer Science University of Pittsburgh
Pittsburgh, PA 15260
tomas@cs.pitt.edu

Milos Hauskrecht
Department of Computer Science University of Pittsburgh
Pittsburgh, PA 15260
milos@cs.pitt.edu

Abstract

Automated traffic monitoring and traffic incident detection is becoming a reality thanks to expanding networks of traffic sensors placed on US highways. However, the current state-of-the-art incident detection systems employ algorithms that require significant manual tuning. In this work, we study machine learning traffic detection solutions that are intended to reduce the need for the time consuming setup. We show that combining a set of simple data traffic sensor data streams via classification methods is a promising way to obtain a detector with an acceptably low false-positive rate and high and fast recall. We build and test a number of SVM-detector solutions and their refinements and show that they can outperform the widely used baseline, the California 2 algorithm. The algorithms are tested on incident data obtained for a section of monitored highway in a metropolitan area.

1 Introduction

The cost of highway accidents is significantly reduced by their prompt detection. However, our data show an often significant delay between the occurrence of an accident and the moment that traffic managers learn about it, due to information propagation time between emergency responders and traffic management centers. This delays the response aimed at mitigating the effects of the accident at traffic management centers, such as proposing alternate routes to the drivers via the electronic information boards on highways. Automated incident detection systems leveraging online traffic flow data promise to alleviate the problem.

The most widely deployed traffic incident detection algorithm is the so-called California #2 algorithm, which is a combination of simple threshold detectors. The tuning of these thresholds requires extensive involvement of traffic experts, as the settings once made typically do not transfer to a new site and need to be set manually for each traffic sensor location. The objective of this work is to devise machine learning techniques that can extract the necessary knowledge directly from the data and thus eliminate the time-consuming human calibration process.

We start by analyzing a set of univariate feature models and their ability to detect incidents. Then we obtain a multivariate detection model by combining the sensor readings with a linear support vector machine [6] and show the improvement over univariate sensors. Since the granularity of information collected at sensors is limited, the probability of spurious detections (false positives) remains high. The false positives are curbed with persistency filtering that allows alarms only if a detection occurs in two consecutive time steps. To learn the models and to compare the solutions we use 1 year of traffic data for a section of an urban highway and the related incident reports.
The traffic data available to us is imperfect in recording the initial time of incidents. A traffic accident record is based on external reporting and thus it may be misaligned in time with the evidence of the accident. We propose and test a simple bootstrap technique to address the problem. The solution relies on a feedback of a high-sensitivity classification model to pre-label the time of accident occurrence in the training data. This approach allows us to build a more reliable detection model.

2 Data

The data are collected by a network of sensors that use a number of physical principles to detect passing vehicles. Three traffic quantities are normally observed and aggregated over a time period: the average speed, the volume (number of passing vehicles) and occupancy (the percentage of road length taken up by cars – “traffic density”). The typical aggregation period ranges from 30 seconds to 5 minutes; currently we have 5 minute aggregates available. We refer to the collection of aggregated measurements from one time interval as a datapoint.

The evaluation takes place on the most accident-prone segment of highway in Pittsburgh. The segment is defined by the positions of two sensors (1 mile apart), to which we will henceforth refer as the upstream and downstream sensors. There are 37 incidents in this segment verified by hand to leave a signature in the sensor measurements.

Incidents that the Traffic Management Center (TMC) was aware of are noted in the data: their approximate location, time of accident and time of clearing by emergency responders (Figure 1). In a supervised framework, we need to label the data as to the occurrence of the accident. Given the unreliability in incident time recording, any datapoint up to 15 minutes (lead tolerance) prior to the accident is considered "accident". As the effects of an accident may persist for some time, we also consider "accident" all datapoints up to 5 minutes (trail tolerance) after the accident is cleared.

Finally, we note that the results in this paper were obtained on data that was in no way "pre-processed". The most obvious such step, supressing diurnal trends by subtracting the daily mean profile [3], surprisingly did not result in significant changes in the reported performance.

3 Traffic features and detectors

In this section we describe the learning and evaluation setup. A detector is any algorithm that takes as input the sensor readings, current and past, and produces a continuous stream of binary outputs signifying the presence of an incident. Conceptually, we associate a detector with a roadway segment between the two given sensors.
3.1 Measuring performance

A false alarm occurs when the system raises an alarm, but no accident is present. The FAR is the number of false alarms divided by the number of detector invocations. The detection rate (DR), is the number of accidents actually detected, divided by the number of accidents that occurred; thus higher DR is more desirable. A performance envelope [8] is a curve relating FAR and DR. To give a more complete description of a detector’s performance, we also report traditional ROC curves [7]. Activity monitor operating characteristic (AMOC) curves are used for evaluation of rare event detection performance, such as detection of disease outbreaks [4]. AMOC curves relate false alarm rate (FAR) to time-to-detection (TTD), which here is the difference between the time of the first datapoint that was labeled as “accident” and the time as recorded in the data. Note that this number can be negative, because of the delayed incident recording. As we cannot guarantee to detect all accidents, we introduce a two-hour time-to-detection limit for accidents that remain undetected.

Accidents are observed indirectly, via their effect on the traffic flow. Their interference with traffic is strong when the highway is near its operating capacity and when the accident is major and blocks at least a single lane. However, it is fundamentally difficult to detect non-blocking accidents and those that occur under light load, as the deviation from normal traffic patterns may be negligible.

The aggregation period of our data – 5 minutes – also limits the achievable performance, because in minor accidents, the roadway if often blocked only for a few minutes before it is cleared. Fortunately, missing such minor incidents carries smaller cost.

The target performance at which a system is considered useful depends chiefly on its users. A study [9] surveying traffic managers found that they would seriously consider using an algorithm that achieves a DR over 88% and FAR under 2%. 1

3.2 Train/test splitting and features

Our dataset is one long sequence. It matters how we split it up into shorter sequences that will be used as learning instances. The straightforward random split cannot be used as it relies on the iid assumption. It is better to divide the train/test split by incidents, making sure an entire incident sequence makes it into one and only one of the sets. To create the training set, we first select \( I_{\text{train}} \) “incident” sequences of preset length \( L \) so that the reported time of the incident falls in the middle of the incident sequence. \( C \) “control” sequences without an incident are selected so that no incident is recorded within additional \( L/2 \) datapoints before and after the control sequence. This safeguards against the imprecise accident recording. By choosing \( I_{\text{train}} \) and \( C \), the class prior in the training set can be biased towards incident occurrences. The testing set consists of the \( I_{\text{test}} = I_{\text{all}} - I_{\text{train}} \) incident sequences that were not selected for the training set. Additional sequences without accidents are added so that the testing set has class prior equal to that in the entire dataset.

To obtain the experimental statistics, we use 10 different train/test splits using the above method, with \( C = 50, I_{\text{train}} = 20 \) and \( I_{\text{test}} = 17 \). All statistics reported are averages and standard deviations across this splits, even where error bars were dropped for sake of readability. Error bars in the graphs represent one standard deviation.

The evaluation focuses on a segment bounded by two physical sensors: \( s_{\text{up}} \) and \( s_{\text{down}} \). Thus our basic dataset contains of six measurements per datapoint: volume, speed and occupancy at the upstream and downstream sensors, respectively. More features are derived from the six basic features. Of particular importance are the differences and proportions of the respective values in time (between datapoints) and in space (between the two sensors).

1The users in the above study were very liberal in their tolerance of false alarms. A medium-sized city will have hundreds of detector sites. A 2% FAR would require the managers to tend to several alarms per minute, most of them false. The users were, in our opinion, victims to the base rate fallacy. However, not all false alarms are necessarily bad. While some alarms are not caused by accidents, they often indicate unusual traffic conditions that, by definition, should be of concern for the traffic managers.
The best performance was 0.288 DR at 0.001 FAR at the calibration

The rationale behind this step is while a capacity-reducing accident will always produce large absolute differences in occupancy, these may also be produced under almost stalled traffic conditions.

The best performance was 0.288 DR at 0.001 FAR at the calibration

The best performance was 0.288 DR at 0.001 FAR at the calibration

threshold in range (1, 10).

Figure 3: (a,b) Performance envelope and ROC curve for a simple detector: $S_{pd}(s_{up}, t_0)$, operating on the upstream sensor. Threshold is varied from the minimal to the maximal value of the reading found in data. (c,d) Performance envelope and ROC curve for $O_{cc}(s_{up}, t_0)$. The threshold is varied from 0 to 5 standard deviations above the mean occupancy. AMOC curves for the simplest detectors are omitted here to save space.

4 The California #2 baseline algorithm

The algorithm known as “California #2” is a popular baseline model against which new detection algorithms are most often compared. Improvements over California #2 have been proposed, but it remains widely deployed [10]. California #2 proceeds as follows:

- Let $O_{cc}(s_{up})$ denote occupancy at the upstream sensor $s_{up}$ and $O_{cc}(s_{down})$ the same at the downstream sensor. If $O_{cc}(s_{up}) - O_{cc}(s_{down}) > T_1$, proceed to the next step.
- If $(O_{cc}(s_{up}) - O_{cc}(s_{down}))/O_{cc}(s_{up}) > T_2$, proceed to the next step. The rationale behind this step is while a capacity-reducing accident will always produce large absolute differences in occupancy, these may also be produced under almost stalled traffic conditions.
- If $(O_{cc}(s_{up}) - O_{cc}(s_{down}))/O_{cc}(s_{down}) > T_3$, wait until the next reading. If $T_3$ is still exceeded, flag an alarm. The wait is introduced to cut down on false alarms.

Thresholds $T_1, T_2, T_3$ need to be calibrated manually for each road segment and we did so for ours. The best performance was 0.288 DR at 0.001 FAR at the calibration $T_1 = 13.0, T_2 = 0.77, T_3 = 5.0$ and verified by an exhaustive procedure trying all possible settings of the three parameters on a discrete grid covering a wide range of parameter values. The performance characteristics of the California #2 detector are in Figure 2. The steep slope of the initial section of the ROC curve is desirable as the most difficult challenge here is the low FAR. However, the model only detects a third of the incidents at best. This is hardly acceptable for practical purposes unless more fine-grained data are made available that could boost its performance.

5 Analysis of univariate detectors

Virtually every “pattern recognition” detection algorithm is built on basic threshold detectors. Let us examine what detection power they have in isolation. In Figure 3, we see the performance of the most basic detectors. Detector $O_{cc}(s_{up}, t_0)$ detects an accident whenever the occupancy sensor reading at the upstream sensor $s_{up}$ exceeds a threshold at the time $t_0$ of the detector invocation. The detector $S_{pd}(s_{up}, t_0)$ outputs 1 if the speed falls below a detection threshold.
In the first SVM experiment, the learner gets as features all the readings at sensors \(s_a\) for a combination of multiple features. We use a linear SVM model to design the detection rule.

It is natural to expect that a better detection performance will be achieved if the detector is based on multiple features. Unlike a benign congestion, an accident should cause radically different flow characteristics at the upstream and downstream sensors. This motivates the inclusion of features that correlate the measurements spatially (Figure 5).

Now we consider, in isolation, the features that capture temporal variation in flow. Intuitively, sharp changes in flow characteristic may be indicative of an accident, while congestion from capacity saturation should have a more gradual onset. The temporal derivative features (Figure 4) are designed to enable this distinction.

Unlike a benign congestion, an accident should cause radically different flow characteristics at the upstream and downstream sensors. This motivates the inclusion of features that correlate the measurements spatially (Figure 5).

### 6 Support vector machine detectors

It is natural to expect that a better detection performance will be achieved if the detector is based on a combination of multiple features. We use a linear SVM model to design the detection rule. \(^2\)

In the first SVM experiment, the learner gets as features all the readings at sensors \(s_{up}\) and \(s_{down}\) at the current time. In subsequent experiments, this basic set is swapped or extended with other classes of features. The results can be seen in Figure 6.

It appears that for our traffic data, the direct sensor readings (speed, volume, occupancy) provide the most detection leverage. Addition of the temporal and spatial difference (and proportion) features affects the performance minimally. This can be explained by the fact that our data are 5 minute averages and the sharp temporal effects important for detection are averaged out.

#### 6.1 Persistency filtering

The California 2 algorithm, which uses differences and proportions between occupancies at the upstream and downstream sensors, without incorporating the actual values, does poorly in the higher FAR portion of the curve. However, in the important region of very low FAR rates, California 2 shows its strength. This occurs because California 2 only signals an incident after it has been verified by the last step, a persistence check. The idea of persistence check can be applied to any detector. The expected effect is that the persistency will eliminate many fluke occurrences that resemble an accident and thus lower the FAR of the new detector. Figure 7 shows the performance statistics

\(^2\)Since our SVM code is designed for the equal misclassification cost, we simulate the unequal cost (loss) by supersampling the corresponding datapoints.
for the SVM-based detector without and with persistency filtering. The results clearly illustrate the benefit of persistency filtering on FAR.

7 Accident relabeling

Unfortunately, accidents in our data are not always time-logged accurately and misalignments between the actual time of an accident and the appearance of accident symptoms often occur. The typical misalignment pattern is that the start time is delayed from the actual accident time. Our concern is that the imprecision in accident labeling may lower detector’s performance if it is trained on such data.

To address the problem we attempt to correct the data and provide a more accurate labeling. We accomplish this using an auxiliary detector that is trained on the original (misaligned) data. For each accident in the training set, we use its first “hit” as the correct accident time stamp. To reduce the probability of making a correction that loses precious positive examples, the auxiliary detector should be a high sensitivity detector. To summarize, before the SVM model is learned on the training data, an auxiliary high-sensitivity pre-detector is applied to data sequences containing an accident.

Figure 8 shows the results for the SVM learned on cleaned (pre-labeled) data. We see a small, but very-low-variance improvement in the detection statistics.

8 Summary and future work

A simple support vector machine learning scheme was able to outperform the model underlying much current traffic detection practice: the California 2 model. Its major advantage is the elimination
Figure 7: Performance at low FAR: (a,b,c) California 2 (d,e,f) the SVM detector without the persistence check. (g,h,i) the SVM detector with persistence check: incident must be detected in two consecutive datapoints for an alarm to be raised. Note the inevitable small loss in time-to-detection.

Figure 8: Bootstrapping experiment. Left column: the original SVM performance. Middle column: the “accident” datapoints were first pre-labeled with a simple thresholding detector that fires if the average speed is 2 standard deviations below the trainset mean. Right, a detector of sharp drop in speed was pre-labeling.
of costly manual tuning of detection thresholds. Moreover, it achieves this performance in presence of noise in the accident labeling. Re-labeling of the data is a promising solution for correcting the accident label problem and for improving the detection performance by providing the learners with higher quality data.

A traffic accident is a dynamic event. The accident goes through two stages: a transient stage that is characterized by a rapidly changing traffic behavior (e.g., speed and volume drops) and a stable stage with steady but still anomalous behavior. We have tried and tested multiple solutions to model separately transient and steady stages hoping this split would let us to detect accident more rapidly. However, we were not successful in this effort. We attribute this to the fact that our data are five minute averages. We believe that transient effects on the section of the highway we analyzed are much shorter than 5 minutes, thus, averages prevent us to build upon them. We expect we will be able to refine this avenue of work when our ongoing collection effort for more fine-grained data with 30-second aggregation is completed.

References


