UNIVERSITY OF SOUTHAMPTON

Improving road incident detection algorithm performance with contextual data

by

Jonny Evans

A thesis submitted for the degree of
Doctor of Philosophy

in the
Faculty of Engineering and Physical Sciences
Civil, Maritime and Environmental Engineering

April 2020

UNIVERSITY OF SOUTHAMPTON

ABSTRACT

FACULTY OF ENGINEERING AND PHYSICAL SCIENCES
CIVIL, MARITIME AND ENVIRONMENTAL ENGINEERING

Doctor of Philosophy

IMPROVING ROAD INCIDENT DETECTION ALGORITHM PERFORMANCE WITH CONTEXTUAL DATA

by Jonny Evans

Road Incident Detection Algorithms (IDAs) help Traffic Management Centres (TMCs) detect, and hence respond to road incidents more quickly and effectively, minimising road network disruption, injury, and risk of secondary incidents. The focus of this project is on developing novel incident detection algorithm techniques to contribute to the field of incident detection.

A major problem faced by state of the art IDAs is the differentiation of incidents from contextual factors. Contextual factors (contexts) are factors that can be expected to cause disruption in traffic conditions in the future. Examples include sporting events, public holidays, weather conditions etc. Although some studies have addressed this problem, none have done so effectively on real-world data. TMCs commonly find that IDAs raise too many false alerts from contexts, and complain of IDAs requiring too much time, effort or expertise to implement.

This research project focuses on how incident detection algorithms can better differentiate incidents from contexts, in an effective and simple enough way to be used in TMCs. The proposed approach incorporates contexts within a traffic forecasting algorithm, which creates forecasts of traffic conditions that can be expected if no incident were to occur. The forecasting algorithm is found to be more accurate than a commonly used historical average predictor in forecasting average speed and flow data from loop detectors, by 4.4% and 4.0% respectively.

Incidents are then detected by an IDA that compares real-time traffic conditions with the forecasts. The IDA is evaluated in offline and online tests in order to ascertain whether incident detection algorithm performance can be improved with the incorporation of contexts. In the offline test, the IDA was shown to improve its performance by using contextual data, in detection rate from 94.4% to 96.7%, and in false alert rate from 1.75% to 1.50%. When tested online, in a TMC, 75 alerts were raised that were confirmed to correspond to incidents, and 49 of these alerts elicited a response from operators to manage the incident. Post-test interviews found that the majority found the developed IDA to a useful addition to their current incident detection methods, and would choose to continue to use the system. These results show that contextual data can be used to improve the performance of incident detection algorithms, in a way that is suitable for use in TMCs.

Contents

List of Figures

List of Tables

Acronyms

TMC	Traffic Management Centre
IDA	Incident detection algorithm
UTC	Urban Traffic Control
DR	Detection Rate
FAR	False Alert Rate
MTTD	Mean Time To Detect
RCID	RoadCast Incident Detection
CCTV	Closed-Circuit Television
ANPR	Automatic Number Plate Recognition
MSE	Mean Squared Error
RFID	Radio-frequency identification
ITS	Intelligent Transportation Systems
UTMC	Urban Traffic Management and Control

Declaration of Authorship

I, Jonny Evans, declare that this thesis entitled Improving road incident detection algorithm performance with contextual data and the work presented in it are my own and has been generated by me as the result of my own original research.

I confirm that:

1. This work was done wholly or mainly while in candidature for a research degree at this University;
2. Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated;
3. Where I have consulted the published work of others, this is always clearly attributed;
4. Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work;
5. I have acknowledged all main sources of help;
6. Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself;
7. Either none of this work has been published before submission, or parts of this work have been published as: Evans et al. 2019.

Signed:

Date:

Acknowledgements

This project was funded by Engineering and Physical Sciences Research Council (EPSRC) and Siemens Mobility Limited.

I would like to thank my primary supervisor, Dr Ben Waterson for his support, advice and challenging questions, which have been invaluable throughout the project. I would also like to thank my industrial supervisor, Andrew Hamilton for his insight, reassuring words and drive to help implement this project's outputs. I'm also very appreciative of the many people at Siemens and the University of Southampton Transportation Research group who have welcomed my research and been very helpful in providing feedback and guidance. In particular, thank you to those at Siemens who have been involved in the process of implementing this research project's outputs.

I would also like to thank my friends and family for their everlasting support over the last three years. In particular, I would like to thank my parents and my partner Georgie, who have helped me greatly through the ups and the downs, and heard more about traffic data than I ever could have asked for.

Chapter 1

Introduction

This introductory chapter describes the task of incident detection, and the role that incident detection algorithms have in assisting in this task. The research project's motivations, aims and objectives are then described.

1.1 Incidents

Since the 1950s, it has been known that road traffic conditions are affected by complex sources of traffic variation (Wardrop, 1952*a*). Some causes of variation are caused by the travel demand of certain events, others by the complex interaction between vehicles and road traffic signals. In the following years, many studies have attempted to better understand this variation for the purposes of improving traffic conditions, such as planning new road network layouts, and incident detection.

The most prominent causes of variation in road network traffic conditions can be categorised into two types. The first type is by contextual factors (or contexts). Contexts are referred to as external events that are planned in advance or predictable, and can be expected to cause variation in traffic conditions in the future. Examples include planned roadworks, sporting events, rush hours, schools closing and weather. The other type is by incidents. Incidents are referred to as unexpected events that cause variation in traffic conditions (Cambridge Systematics Inc., 2001). Examples of incidents include vehicle collisions, illegal parking and unloading, vehicle breakdowns and unplanned roadworks. Incidents cannot be expected to occur at any particular place and time, so the variation caused cannot be preempted. The point of difference between incidents and contexts is that incidents are inherently unexpected, and so the variation they cause in traffic conditions cannot be predicted beforehand, whereas contexts are expected to happen beforehand, and so their variation may be predictable to some extent. It should be noted that variation in traffic conditions is also caused by the variance between individual drivers' behaviour, and from measurement from collecting and processing data real-world detectors. These background sources of variation make differentiating incidents and contexts a challenging task.

1.2 Incident detection

Traffic Management Centres (TMCs) are responsible for managing the flow of traffic across a road network, enforcing road traffic regulations and accommodating servicing works. Part of a TMC's responsibility is the task of incident detection. That is, to ascertain that an incident has occurred at a particular place in a given road network. TMCs can do this in a number of ways, including:

- Monitoring Closed-Circuit Television (CCTV) cameras.
- Using communication from external services such as the emergency services, who may have relayed communication from road users.
- Using communication from road users directly (e.g. from social media).
- Using Incident Detection Algorithms (IDAs).

It should be noted that recent advances in technology have allowed for certain types of incidents to be detected quickly and automatically. For example, the eCall system has been made mandatory in all vehicles in the E.U. since March 2018 (European Global Navigation Satellite System Agency, 2018). The system is automatically activated when in-vehicle sensors detect a serious crash. Once activated, a message to the emergency services is sent, including the time of the crash, and the location and direction of the vehicle. This message can then be passed on to the local TMC. However, many other types of incidents, such as breakdowns and illegal parking, cannot be detected in this way. It should also be noted that these methods may also be used in combination. For example, a TMC may use an IDA to first indicate the presence of an incident, and then verify its presence and location using CCTV.

Incidents create significant costs to road network users, including delays, vehicular damage and personal injuries (Connelly and Supangan, 2006). As such, TMCs aim to detect incidents as quickly and effectively as possible, so that they can be responded to in a way that reducess their consequences. TMCs can respond to incidents in a number of ways, including:

- Notifying emergency services of incident occurrence and location.
- Traffic management strategies, including creating diversions and altering traffic signal control strategies.
- Notifying the public of incidents using radio bulletins, variable message signs, social media etc.

1.3 Incident detection algorithms

To aid TMCs in the task of incident detection, Incident Detection Algorithms (IDAs) can be employed. IDAs automatically analyse real-time traffic data across a road network, and once

such data is seen to be representative of an incident, raise alerts to TMC operators of the likely occurrence of an incident. IDAs can be employed to either automate, or aid TMC operators in the task of incident detection. In the case of aiding TMC operators, IDAs could be designed to raise alerts quickly at the first indications of an incident taking place, then an operator could verify the incident's presence (often by observing CCTV), before responding to the incident.

The benefit of IDAs to TMCs is that they can analyse data more quickly across a road network than an operator could manually. This empowers TMCs to detect incidents more quickly, accurately and/or over larger road networks. By helping TMCs detect, and hence respond more quickly and effectively, IDAs ultimately mitigate the consequences of incidents, including delay, injuries, and the risk of secondary incidents (i.e. incidents caused by preceding incidents (Wang et al., 2005)).

There are a wide range of types of data that IDAs use to detect incidents. Most IDAs use numerical variables, such as flows, occupancies and average speeds. These variables can be collected from detectors such as inductive loops, Automatic Number Plate Recognition (ANPR) cameras or probe vehicle detectors. Loop detectors detect the absence or presence of a vehicle at a point on a road, typically at a high sampling rate such as 0.25 seconds. The detector consists of a loop wire underneath the road surface which creates an electrical circuit. If a vehicle passes, the inductance of the circuit decreases. The absences and presences are aggregated to form metrics of flow and occupancy (percentage of time that a vehicle is present). Federal Highway Administration (2006) provides further detail of the theory of loop detector operation. ANPR detectors identify number plates from images of vehicles as they go past. When the same vehicle passes two ANPR cameras, travel time data can be calculated. Probe vehicle detectors submit position and speed data from individual vehicles, typically from smart phones or connected vehicles. A common IDA approach is to compare values of these numerical variables at upstream and downstream detectors, that is, pairs of nearby detectors in the same direction on the same road. Other IDAs' inputs include images from CCTV video, and signals from audio detectors.

The common objective of IDAs is to aid TMCs in achieving the task of incident detection as much as possible. That is, to empower TMCs to detect incidents as quickly and effectively as possible. This objective is typically evaluated in terms of IDA's detection rate, false alert rate, and average time to detect incidents. However, other attributes and features may aid TMCs further in achieving the task of incident detection, and so must also be considered. For example, operators could get an idea of the certainty of the algorithm's alerts by using a prediction interval, which is an estimate of an interval for which there is a certain probability that future observations will fall into.

In this study, and throughout the literature, it is assumed that variation caused by contexts and incidents should be differentiated by IDAs, and alerts produced because of variation from contexts are to be classified as false alerts. This is due to the fact that contextual factors can be expected to occur beforehand, and hence such alerts are unwanted by operators. For example, a detector nearby a school may be congested after school finishes every weekday during term

time, but an operator would not benefit from being alerted of this disruption every time due to its predictability. Despite this, it is expected that the desired type of alerts produced by an IDA may change based on the preference of the TMC. For example, for TMCs that do not have operators actively monitoring the network, information of disruption due to contexts may be beneficial in order to automatically alter traffic signal strategies. Hence, in practice, an IDA may be most widely beneficial if it were able to differentiate contexts from incidents, and display the type of alerts desired by the given TMC. However, in this study IDAs will be evaluated solely on their incident detecting abilities.

1.4 Motivation for the study

The key motives for this research are highlighted in the following subsections.

1.4.1 Road network delay

In the U.K., vehicle miles travelled has been increasing over the past 60 years, and appears likely to continue with the advent of new technologies such as connected and autonomous vehicles (UK Department for Transport, 2017). This presents a great challenge to road networks to manage the flow of traffic, in particular during incidents. Improved IDAs are required to more effectively manage incidents, minimising the disruption caused to the road network.

It is difficult to quantify the amount of traffic disruption caused by U.K. incidents and their consequences, but it is clearly very significant. On U.S. roadways, it was estimated that traffic incidents account for one quarter of all congestion (National Traffic Incident Management Coalition, 2017). The consequences of congestion due to incidents is not only large in terms of loss of time, but of emotional consequences, such as anxiety and frustration from a missed flight, appointment or delivery (Yass, 2017). Also, the stationary traffic on major roads caused by incidents is in itself a hazard, that can increase the risk of further incidents.

1.4.2 Safety

The number of deaths on U.K. roads has stayed between 1,700 and 2,000 in the seven years up to 2019 (UK Department for Transport, 2019). That's an average of five deaths per day. The U.K. ranked as the 5^{th} safest country in 2013, in terms of road deaths per 100,000 inhabitants (World Health Organisation, 2017). However, it appears that this statistic could be improved upon by responding to incidents more quickly. A study by medical experts found that in European high-income countries, about 50% of road traffic fatalities occurred within 60 minutes of the incident, either at the scene or while in transit to hospital. Of those taken to hospital, around 15% occurred between 1-4 hours after the incident, and 35% occurred after four hours (Buylaert, 1999). Evanco (1996) found that in 1992, the average time between crash occurrence and medical service notification was 4.28 minutes on U.S. interstates, freeways and expressways (Evanco, 1996). It was estimated that if this figure was reduced to three minutes with the use of

effective incident detection algorithms, 449 lives and $267 million monetary cost would be saved. This monetary saving would mainly from reducing delays from disruption, resulting in a small productivity increase to many travellers.

England's Department of Health has set standards for ambulance services to attend to the scene of 75% of life-threatening calls within 8 minutes, but this target was not met in 2015/16 (NHS England, 2017). If incidents were detected more quickly, emergency services could be contacted sooner, and the disruption caused could be reduced, meaning there would be less delays in reaching the incident. Clearly then, improvements in incident detection could reduce the risks and consequences of injuries.

1.4.3 Environmental

Incidents cause disruption to the surrounding road network in the form of congestion, which in turn leads to an increase in the amount of pollution from vehicles. The UK's current target is a reduction in greenhouse gas emissions of at least 80% by the year 2050, relative to 1990 levels (Gummer, 2016). Such policies have put pressure on the transport industry to help reduce emissions on road networks. This research project addresses these pressures by proposing algorithms that help operators reduce the disruption caused by incidents, reducing incident congestion and in turn, carbon emissions.

1.4.4 Financial

There is a financial cost involved in many of the consequences of incidents. Medical and ambulance costs are incurred to respond and treat victims of incidents. Delays cause a lack in productivity that can be represented as a monetary value. Police and traffic management services incur costs to manage the disruption from incidents. The Department for Transport estimated that the total value of prevention of accidents that occurred in the U.K. in 2018 came to £35.5 billion (Department for Transport, 2018). By improving incident detection and hence response times, IDAs can help to mitigate the consequences of incidents, which in turn can help reduce the financial pressure on the services involved.

1.4.5 Opportunities from technology

A number of areas of technology have progressed quickly in recent years, which has created new opportunities in incident detection research. As previously implemented incident detection algorithms have been limited by past technology, recent improvements in technology provide motivation to improve on the performance of previous incident detection algorithms.

It was estimated that the number of connected devices (to the internet) in the world rose from 100 million in 1992 to 14.4 billion in 2014 (CompTIA, 2016). Meanwhile, the cost to store a gigabyte of data has fallen from $569 in 1992, to $0.03 in 2012 (Hagel and Seely Brown, 2013). There has also been an exponential decrease in the cost of computing power, from $222 per

million transistors in 1992, to $0.06 in 2012. Hence, more and more data has been collected and stored in recent years, leading to trending phrases such as 'the internet of things' and 'big data'. The increase in computing power has made the analysis of such 'big data' more practical. Each of these advances in technology has provided opportunities for IDAs to improve, as data from these connected devices can be used as inputs to IDAs.

An example of IDAs taking advantage of this trend is the rise of machine learning IDAs. Machine learning is a type of artificial intelligence that gives algorithms the ability to learn without being explicitly programmed. In recent years, applications have led to advances in many fields, from autonomous vehicles, to the prediction of diseases (Kourou et al., 2015, Bojarski et al., 2016). Incident detection is no exception, many of the algorithms developed in the past 20 years incorporate a form of machine learning. This is largely because they require large amounts of historical data to be analysed quickly, and so have become more suitable in recent years. However, one must not assume that each IDA employing new technology will necessarily benefit incident detection. Each IDA reviewed and developed will be evaluated solely on its ability to meet the objective of aiding TMCs detect incidents.

1.5 Research approach

This section describes the research approach taken in this research project.

Firstly, this research project's problem is defined. From this, the aim and objectives of the research project are described.

Next, a review of incident detection algorithms, both in the literature and in practice, is undertaken. Where necessary, this review is supplemented with interviews with key stakeholders in the industry, in order to validate the relevancy of the review's findings. This review will highlight the limitations, and opportunities for improvement in the current state of the art.

Based on this review, hypotheses are identified that address the aim of this research project. An approach to testing this hypothesis, in this case the high-level approach of an IDA, is then developed.

The following chapters describe the development and testing of the proposed IDA. In this case, the development of the IDA includes the development and testing of a traffic forecasting algorithm, which is key to the IDA. To ensure that the IDA is thoroughly evaluated, it will be evaluated in offline and online tests. The offline test will involve the collection of historical traffic and incident data to train and test the IDA on. The developed IDA will be compared to the state of the art in order to understand its effectiveness. The online test will involve TMC operators using the IDA and providing feedback in real-time, as well as giving interviews before and after the test. This test will be used to understand the performance that can be expected of the IDA in practice. Based on the findings of these tests, an evaluation will be made as to whether the

aims and objectives of this research project have been met.

Figure 1.1 shows a flow chart of the research approach taken in this project.

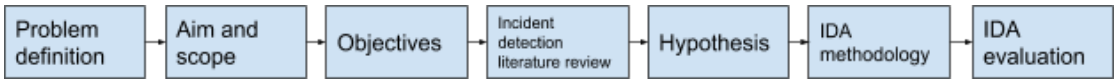


FIGURE 1.1: Flow chart of this project's research approach.

1.6 Problem definition

Incident detection algorithms have been developed since the 1970's, but studies have found that when implemented in TMCs, many have been disabled or simply ignored (Parkany and Xie, 2005). The most commonly stated reasons for this are the excessive time, effort or expertise required for calibration, and the poor operational performance when implemented, particularly in terms of the number of false alerts raised. For a time, this was not of great concern because manual incident detection methods could perform adequately in most situations (Guin, 2004). However, the size and scope of road networks under TMC monitoring have grown while staffing levels and TMC resources have been cut, meaning manual methods are becoming less and less viable (Guin, 2004). As such, IDAs must be developed to empower TMCs to detect incidents across large networks quickly and effectively.

It has been shown that a major problem with state of the art IDAs is their inability to differentiate incidents from disruption caused by contextual factors, such as rush hours, sporting events or public holidays. When such factors disrupt traffic conditions in a way that is similar to an incident, unnecessary false alerts can be created. These false alerts waste TMC operators' time in verifying incidents, and can lead to a feeling of mistrust, in some cases leading to disabled or ignored IDAs. This problem is also very common in the literature, particularly amongst IDAs tested in urban networks. This problem appears to be one of the biggest limiting factor in state of the art IDAs' performance. It also appears to be a somewhat neglected problem, and somewhat tractable given the recent advances in data collection, computing power and algorithm development.

1.7 Aim and scope

The aim of this research project is to develop a novel incident detection algorithm that is able to differentiate incidents from contexts, and uses this ability to improve on the state of the art. To do this, a review of IDA and relevant literature will be undertaken, and a novel IDA will be developed. This algorithm will then be evaluated on real-world data to identify the extent to which the limitation(s) has been accounted for, and whether the state of the art has been improved upon.

Due to the practical availability of data, the scope of this research project is limited to U.K. urban road networks. However, where possible, the approaches will be developed to be transferable to all types of road networks.

1.8 Objectives

The objectives of this research project are to:

1. To systematically review the 'state of the art' in incident detection, and highlight any limitations and opportunities for improvement.
2. To develop an IDA that addresses the issue of differentiating incidents from contexts, and implement it on a real-world traffic dataset.
3. To evaluate the developed IDA on real-world data and make comparisons to the state of the art in order to determine whether the IDA has addressed the issue identified, and has improved on the state of the art.
4. To provide recommendations based on the findings of this research project, including details on the developed IDA's limitations, opportunities for improvement and implementation requirements.

1.9 Expected contribution and impact

As stated in the problem definition in section 1.6, a major problem faced by state of the art IDAs is the differentiation of incidents from contextual factors. Although some studies have addressed this problem, none have done so effectively on real-world data. This is a problem which can be approached in a number of different ways, such as better understanding typical road traffic patterns, incorporating past incident data to better understand their effects, or incorporating contextual data in order to predict their effects, or a combination. It is not clear what the most effective method is to solve this problem, nor whether one exists that would be suitable for use in an IDA that could be used effectively in a TMC.

In this project, a novel approach to the problem above is presented. It is hoped that this approach will be effective at addressing the problem, but also be suitable for use in an IDA that is effective in TMCs. This approach incorporates contextual data into a traffic forecasting algorithm. The aim of this is to create an understanding of traffic patterns that can be expected to occur in the future, given expected disruption from contexts. A real-time incident detection algorithm will use this algorithm's forecasts as input in order to detect incidents. At the end of this research project, an understanding of the extent to which this approach solves the issue of differentiating incidents from congestion will be gained. The trade-offs and limitations of this approach will also be detailed. This understanding will aid future researchers to develop more effective IDAs, and will be a positive step towards more effective IDAs in TMCs.

This research project is expected to contribute to both academia and industry by meeting its aims and objectives. By meeting the aim of addressing the issue of differentiating incidents from contexts, future research on developing state of the art IDAs could build on this research project's findings of how the issue could be addressed. By meeting the objective of providing recommendations based on the findings of this research project, future research projects could use these recommendations to more readily improve on state of the art IDAs.

This research project will take advantage of the recent trends in computing power, connected devices and algorithms (discussed in section 1.4.5) to contribute to knowledge by developing and applying cutting edge algorithms to tackle problems faced by state of the art IDAs.

1.10 Thesis structure

This section provides a brief summary of each chapter.

Chapter two reviews the literature of state of the art incident detection algorithms, highlighting the limitations with the current methodologies and areas which could be improved upon.

Chapter three discusses the performance of state of the art incident detection algorithms in practice. This includes a review of TMC surveys in the literature, and the findings of a survey of TMC operators in the U.K., which was undertaken as a part of this research project. The findings aim to reflect operators' and TMC managers' views of IDAs that have been implemented in practice.

Chapter four develops the approach that this research project's IDA will take, and provides justification for it.

Chapter five describes the methodology of a traffic forecasting algorithm, RoadCast, is developed with the aim to understand whether forecasts can be improved with the incorporation of contextual information. This algorithm will be key to this project's proposed IDA.

Chapter six evaluates RoadCast's performance. RoadCast is tested offline on loop detector data from Southampton, U.K. A number of different scenarios are tested, including various amounts of training data and various forecast horizons, as well as the ability to interpret its decision making process. The potential of using RoadCast as the basis for this research project's IDA is found.

Chapter seven describes an IDA, RoadCast Incident Detection (RCID), which is developed to understand whether incident detection can be achieved using RoadCast's forecasts. In particular, it is studied whether RCID could use contexts to better differentiate contexts from incidents. It is tested offline on loop detector data from Southampton, U.K.

Chapter eight describes an online test of RCID in a Traffic Management Centre (TMC). RCID is implemented on loop detector data from Bristol City Council's TMC, and is evaluated from TMC operator interviews and real-time feedback from RCID's alerts. This test investigates whether RCID is suitable for use in practice.

Chapter nine presents the conclusions drawn from the work in this thesis, including the contributions made, the necessary implementation considerations, and possible future research directions.

Chapter 2

The evolution of incident detection algorithms

2.1 Introduction

To get a clear understanding of how the state of the art in incident detection can be improved upon, a review of all types of IDA was seen as necessary. This includes every type of approach taken, those designed for urban networks and motorways, those tested only on simulators and those implemented in TMCs. This review also addresses objective one of this research project (see section 1.8). This review is not exhaustive of all IDAs presented, but instead highlights IDAs that were clearly methodologically different.

It should be noted that most IDAs presented are designed for and tested on motorways, i.e. controlled access roads with two or more lanes. A small proportion of algorithms are designed for use in urban networks, i.e. urban arterials, streets and junctions. However, a significant proportion of road incidents occur in urban networks. Somerset Intelligence (2017) found that 37% of fatal and 54% of slight injury incidents in Somerset, England, occur on urban roads (Somerset Intelligence, 2017). Urban networks present many extra challenges for incident detection that are not faced on motorways. Diverse sets of travel demands and modes coexist in urban networks, short and long distance trips are taken by commuters, business and leisure travellers alike (Weijermars, 2007). This in combination with the more complex network topologies (many types of roads, junctions and signal layouts), make traffic movements and congestion propagation less predictable, and hence incident detection more difficult (Siripanpornchana et al., 2015). As such, IDAs typically perform better when tested on motorways than on urban networks (Zhang and Taylor, 2006). This chapter reviews both IDAs designed for urban and motorway networks.

The following sections highlight each type of approach taken by IDAs to detect incidents, before discussing examples of each type in turn. Then, limitations found with presented IDAs are described, and research addressing these limitations is reviewed.

2.2 Classification of incident detection algorithms

Depending on how IDAs approach the task of incident detection, algorithms can be classified into different types. These types have been defined differently in previous incident detection review papers (Balke, 1993, Mahmassani et al., 1999, Ozbay, 1999, Martin et al., 2001, Parkany and Xie, 2005, Deniz and Celikoglu, 2011). The classification used in this study has been created in an effort to best reflect the current landscape of existing incident detection algorithms. The algorithm types are comparative, time-series, machine learning, image and audio processing, and data fusion. Although the classification used does not tightly categorise all existing IDAs (some IDAs may use multiple approaches), it does help to describe and differentiate the approaches taken by existing IDAs.

Comparative algorithms compare variables from real-time traffic data against pre-set thresholds to identify incidents. Commonly used variables include flow, occupancy and average speed. Alerts are raised when real-time traffic data is observed to fall outside of the given threshold. The majority of early IDAs (1970s to 2000) were comparative algorithms.

Time-series IDAs model traffic variables as a time-series. They use recent historical observations of traffic variables to forecast future values, and raise alerts when these values are sufficiently different to real-time values. Time-series IDAs differ from comparative algorithms in that the threshold varies based on past values of variables. Time-series algorithms assume that traffic data follows a predictable trend over short periods of time during incident-free periods, and that this trend is disrupted when an incident occurs.

The application of machine learning techniques to real-world problems has increased dramatically in the past couple of decades, incident detection being no exception. In fact, the majority of urban IDAs developed in the last 20 years have used a form of machine learning. The typical approach of a machine learning IDA is to use historical data to 'learn' what conditions are representative of an incident and non-incident. The algorithm then takes real-time traffic observations as input to predict the incident state (incident or non-incident).

Rapid advancements in computer vision research, a sub-field of machine learning, have led to many image processing traffic applications in recent years (Kastrinaki et al., 2003). Instead of using numerical metrics of traffic conditions, image processing IDAs analyse real-time videos of the traffic itself (specifically, the pixel values of each video frame), and raise alerts when incidents occur on video. The algorithms first train on historical videos of incidents to 'learn' the features of incident and non-incident scenarios, then detect incidents when real-time video appears similar to the 'learnt' incident scenarios. Audio processing is similar in approach to image processing, except incidents are detected based on the noise from nearby audio sensors.

Data fusion algorithms combine multiple types of data, and sometimes algorithms, to detect incidents. Typically, separate algorithms first detect incidents on each data source independently.

The outputs are then combined using another algorithm to raise alerts when certain conditions are met.

2.3 Evaluating incident detection algorithm performance

As discussed in section 1.3, the objective of IDAs is to aid traffic operators as much as possible in achieving the task of incident detection. However, the evaluation of this objective is not immediately clear. Throughout the literature, IDAs are typically evaluated on three measures, their average time to detect incidents, detection rate, and false alert rate. However, the definitions of these measures vary, and additional features and attributes of IDAs may also be used in evaluation. To arrive at the definitions of these performance measures, incidents and a method of incident verification must also be defined. As such, the following subsections describe the definitions used in this study.

2.3.1 Incident definition

To be able to evaluate IDAs ability to detect incidents, a clear definition of an incident must first be found. Many previous studies have presented algorithms which are successful in identifying a change in traffic state from the input data, but this change may not be caused by incidents, and so may not be of interest to TMCs. A clear definition of an incident must capture all the causes of changes in traffic state which are of interest to TMCs. However, not all previous studies clearly define an incident, and the definitions that do exist are inconsistent throughout the literature. On the basis of the literature studied, the following definition of an incident has been created for this study.

"Incidents are unexpected events that cause variation in traffic conditions"

By unexpected, it is meant that the event could not have been predicted to occur beforehand. The variation caused could be of any magnitude, and so may not be disruptive enough to be noticeable in nearby detectors' data. This presents a challenge in evaluating the performance of IDAs, because those based on the disruption to traffic conditions may not be able to detect certain incidents.

This definition is thought to best capture the types of events that TMCs need to detect. Although the definition of an incident varies throughout the literature, and is often unstated, this is the most commonly used definition. Examples of incidents include vehicle-on-vehicle impacts, vehicle breakdowns, illegal parking or unloading and emergency works.

2.3.2 Incident verification

Once a clear definition of an incident has been established, an IDA's alerts must be compared with verified incidents in order to evaluate its performance. In previous studies, many papers have used simulators to artificially replicate such incidents (Sheu and Ritchie, 1998, Zhang and

Taylor, 2006, Ghosh and Smith, 2014). Incident free traffic data is used as input to the simulator, and conditions replicating an incident (such as a parked vehicle blocking a lane) are artificially simulated. The IDA's alerts can then be directly compared against the simulated incidents. An advantage of simulated incidents is that it is a theoretically unlimited data source of incidents because simulations can be run any number of times. This is especially useful for this application because if real-world data were to be used, a large amount of data would be needed because of the fortunate rarity of incidents. Another way to evaluate performance is to identify incidents from TMC logs or police records. The IDA is then run on the associated traffic data, and the alerts raised are compared with the logged incidents (Cherrett et al., 2002).

Singliar and Hauskrecht (2010) state that IDAs that work well on simulated incidents do not always give the same performance when implemented on field data (Singliar and Hauskrecht, 2010). This is because simulations do not always produce environments representative of the real-world locations an IDA may be implemented in. Difficulties include simulating contextual variations, developing representative origin-destination matrices, and creating incidents that have realistic locations, frequencies and disruptions. This can mean that although IDAs can be trained on simulated data such that they get very high performance in simulated environments, this may not translate to their performance in real-world networks. On the other hand, TMC incident logs have the drawback of being difficult to obtain, and very temporally sparse due to the fortunate rarity of incidents. Singliar and Hauskrecht (2010) comment that TMC incident logs are also often subject to bias and noise. For example, the time that incidents are logged can have a 'variable delay' from the actual time of the incident occurring, because incidents are only logged once an operator has verified them. This delay can therefore be detrimental to IDAs that require historical incident logs to 'learn', because the incident label may be misaligned with the incident itself, and hence the change in traffic state caused. Singliar and Hauskrecht (2010) propose a dynamic Bayesian Network to realign the timing of each incident log. Firstly a sample of incident logs from a particular motorway segment are manually realigned. Then once the Bayesian Network is trained on such logs, they use it to pre-process the rest of their incident log dataset. However, tests on real-world data with incident logs is generally seen to be the more reliable environment in which to evaluate IDAs, as this gives a closer representation of what could be expected when an IDA is implemented in a TMC. Of course, the most effective evaluation of an IDA would be from a real-world trial of the IDA in a TMC, where operators could judge the effectiveness of the algorithm for themselves. However, this approach would have difficulty in comparing state of the art IDAs, unless multiple operators in the same TMC were asked to use different IDAs.

There must also be a clear method of deciding whether an IDA was successful in identifying an incident. This allows for the differentiation of false alerts and true detections. To do this, an area of detection and maximum time to detect could be established. IDAs that compare upstream and downstream detectors traffic often state this as the area between the detectors, meaning any incident occurring in this area should be detected. While predicting congestion across Greater Seattle, Horvitz et al. (2012) classified their predictions as successes if an actual bottleneck occurred within 15 minutes of a predicted bottleneck time. In the following review of incident detection algorithms, incident verification methods vary, and are often left unstated.

2.3.3 Incident detection algorithm performance measures

The common objective of IDAs is to aid TMCs as much as possible in achieving the task of incident detection. To evaluate IDAs, quantitative measures are typically used. Many different measures have been used in the literature, but the most common are detection rate, false alert rate, and the mean time to detect. Many different definitions of these measures exist, but below are the most commonly stated. In this review, unless stated otherwise, the performance metrics are defined as below. These definitions are also used later in this study to evaluate the developed IDAs.

An IDA's detection rate is the percentage of the number of correctly detected incidents in a given time period and area, to the total number of verified incidents that occurred in the same time period and area. An IDA will be judged as correctly detecting an incident if an alert was raised at any point during an incident.

$$Detection\,rate\,(DR) = 100 \times \frac{Number\,of\,correctly\,detected\,incidents}{Total\,number\,of\,actual\,incidents\,in\,the\,dataset} \tag{2.1}$$

An IDA's false alert rate is the percentage of the number of messages for which an alert was raised but no incident was occurring, to the total number of messages for which no incident was occurring (i.e. false positive rate). It should be noted that here, a message is used as a term to represent a collection of traffic metrics that cover a particular time period at a detector (or detection location).

$$False\,alert\,rate(FAR) = 100 \times \frac{Number\,of\,messages\,where\,an\,alert\,was\,raised\,falsely}{Total\,number\,of\,messages\,where\,an\,incident\,did\,not\,occur} \tag{2.2}$$

Mean time to detect is the mean time taken (in minutes) by an IDA to raise the alert for a correctly detected incident, over a given time period and area.

$$Mean\,time\,to\,detect(MTTD) = \frac{1}{n}\sum_{i=1}^{n}(A_i - O_i) \tag{2.3}$$

Where n is the number of verified incidents, A_i is the start time of an IDA's alert being raised, and O_i is the start time of the corresponding incident.

It should be noted that false alert rate, detection rate and mean time to detect are closely linked, and an improvement in one may be transferable to degradation in others (Ghosh and Smith, 2014). For example, an IDA may be able to lower its sensitivity of raising alerts in order to

reduce its false alert rate, but it would come at the cost of reducing its detection rate and increasing its mean time to detect.

In the literature, less commonly stated measures of performance included:

- The feedback of TMCs, including thoughts on IDAs' operational performance, usability, ease of implementation. Although this measure will be subjective, it is an important factor affecting the usefulness of IDAs in TMCs.
- The time needed to calibrate to a new location. That is, to go from the raw data required, to detecting incidents in real-time.
- Once implemented, the frequency and time taken to re-calibrate the IDA to maintain performance.
- If trained on field data, the time span of the training data required.

2.4 Review of incident detection algorithms

The following subsections discuss presented IDAs, both for urban and motorway networks. The IDAs are ordered in a way to document the evolution of each type of IDA. Rather than describing every presented IDA in detail, this review aims to describe the significant differences made as each type of IDA evolved. The IDAs are grouped as comparative, time-series, machine learning, video-image processing, audio processing, social media and data fusion.

2.4.1 Comparative algorithms

2.4.1.1 HIOCC and PATREG

The Transport and Road Research Laboratory (TRRL) developed the High Occupancy (HIOCC) and Pattern Recognition (PATREG) algorithms to detect incidents over loop detectors on motorways (Collins et al., 1979*a*). It was envisaged that HIOCC AND PATREG would be implemented and work together, as PATREG would detect more quickly, but HIOCC would be more reliable and better at detecting the end of an incident. However, in online tests in the literature, PATREG was tested by itself, rather than in combination with HIOCC.

The premise of the HIOCC algorithm was that vehicles slow considerably during incidents, resulting in high occupancy values on loop detectors. Alerts were raised when a vehicle occupied a detector for two consecutive seconds (i.e. 100% occupancy for two consecutive one second periods).

The PATREG algorithm worked on pairs of detectors (around 500 metres apart) in the same lane on motorways. The idea was that a drop in average speed of vehicles between detectors

would be indicative of an incident. Firstly, the average speed of vehicles was estimated using a pattern recognition technique using occupancies at the first detector and flows. Predetermined upper and lower thresholds of speed for each pair of detectors are set, and alerts are raised when real-time estimations of vehicle's speed are outside these thresholds for 20 seconds consecutively, i.e. every vehicle in a period of 20 seconds is outside the threshold. When calibrated on an urban motorway near London, typical upper and lower thresholds were 74 and 48 miles per hour (mph) respectively.

It should be noted that the PATREG algorithm employs the commonly used technique of a persistence test to help raise incident alerts. Instead of raising an incident alert as soon as a condition is met, the idea here is to wait until this condition is met for a certain duration, or number of times periods. This is typically employed when an IDA is particularly sensitive to noise in the traffic variables used. By introducing a persistence test, the number of false alerts from noise can reduce, but the time taken to detect incidents increases.

PATREG was tested using a number of staged incidents, organised between two detectors spaced 530 metres apart, on the Boulevard Peripherique (controlled-access dual-carriageway ring road) in Paris (Collins et al., 1979*b*). 12 incidents were staged, lasting between three and 12 minutes. HIOCC detected each of these incidents, and raised no false alerts. The detection time ranged from 20 seconds to 130 seconds. However, the PATREG algorithm was unable to detect any of the incidents. It was reported that the IDA failed during flows exceeding 1,500 vehicles per hour per lane, due to the flow conditions being too random in nature to provide a recognisable pattern.

Both the HIOCC and PATREG algorithm require fixed thresholds to be manually calibrated at each site depending on the local traffic conditions. This may result in poor performance in urban networks due to the IDA being unable to account for the variation in traffic patterns between detectors in different locations. Another downside is that it only includes traffic data as input, meaning that some contextual factors, such as football matches, would be difficult to account for. If certain contextual factors produced disruptions in traffic conditions that appear similar to incidents' disruption, false alerts could be created.

2.4.1.2 California algorithms

The California algorithms are amongst the most commonly cited and replicated IDAs. Many variations of the original have been presented and compared (Payne and Tignor, 1978). But each use pre-set decision trees based on many traffic variables, to classify real-time traffic conditions into incident and non-incident states. Typically the traffic variables are derived from detector's occupancy. The rest of the paragraph describes the original algorithm presented (Payne and Tignor, 1978). The variables used were upstream and downstream occupancy, their difference, the difference relative to the upstream occupancy, and the change in the downstream occupancy over time relative to the downstream occupancy. When real-time values of these variables fell into certain thresholds, the algorithm would raise an alert. The algorithm was tested on one minute values of occupancy data from loop detectors in Los Angeles, U.S.A. It was compared to the

double exponential smooth algorithm (described in section 2.4.2.2). Figure 2.1 show the results found. While tuning the parameters to alter the false alert and detection rate, the California algorithm performed better in most cases. The average time to detect was not reported.

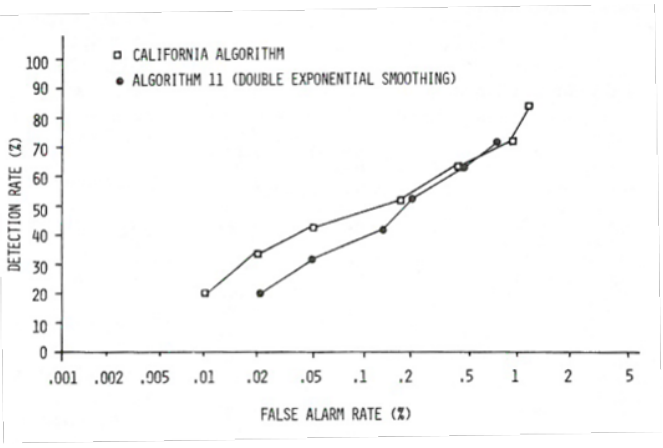


FIGURE 2.1: Performance of the California algorithm in comparison to the double exponential smoothing algorithm (Payne and Tignor, 1978).

A drawback of the IDA presented is that only occupancy variables were used, meaning it would likely only be able to detect congestion, rather than differentiating incidents from contexts. Payne and Tignor (1978) acknowledged this by saying that significant differences in occupancy values can also be caused in normal conditions, such as by 'geometric bottlenecks'. But they argue that this was 'distinguished by the fact that (during incidents) the downstream occupancy decreases rather abruptly'. As was found later in this review, similar approaches implemented in urban networks reported false alerts caused by contexts. Another drawback of most variations of the California algorithm is that pairs of detectors are required (and need to be identified), and their thresholds need to be calibrated manually.

Because of their simplicity, many studies have used the California algorithms as a benchmark for comparison, and many others have iterated on the first version presented to improve its performance and limit its drawbacks.

2.4.1.3 McMaster algorithm

Catastrophe theory classifies events characterised by sudden shifts in behaviour arising from small changes in circumstances. Persaud and Hall (1989) used a catastrophe theory model as the basis for the McMaster IDA, with the premise that as traffic conditions change from an uncongested to congested state, there is a sudden drop in speed. The definition of a congested state was based on the value of occupancy, flow and average speed. The IDA's premise is shown in figure 2.2.

In the basic version, a congested state was observed when real-time traffic conditions had low average speed, or were within certain defined areas of the flow/occupancy graph. Areas two and three of figure 2.3 represented the congested state. During calibration, the areas were defined manually with a set of training data. Then, incident alerts were raised when real-time traffic

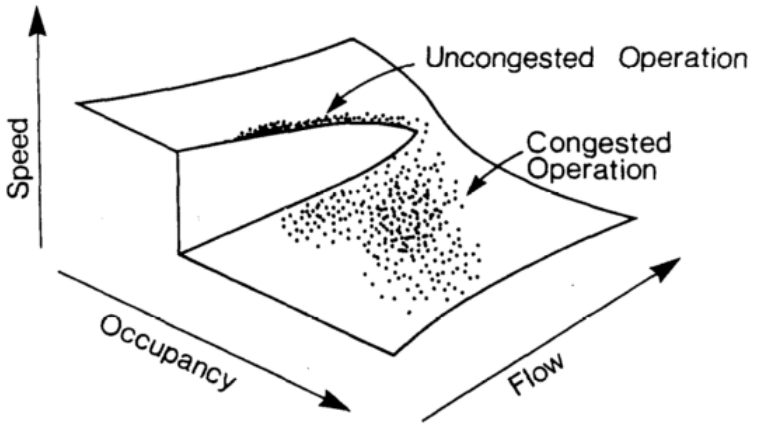


FIGURE 2.2: Conceptualisation of traffic operations on a catastrophe theory surface (Persaud and Hall, 1989).

conditions showed low speeds, or were within areas two or three for two consecutive time periods (i.e. one minute). The alert would then stop when the conditions returned to higher speeds, or area one, for three consecutive time periods (i.e. 1.5 minutes).

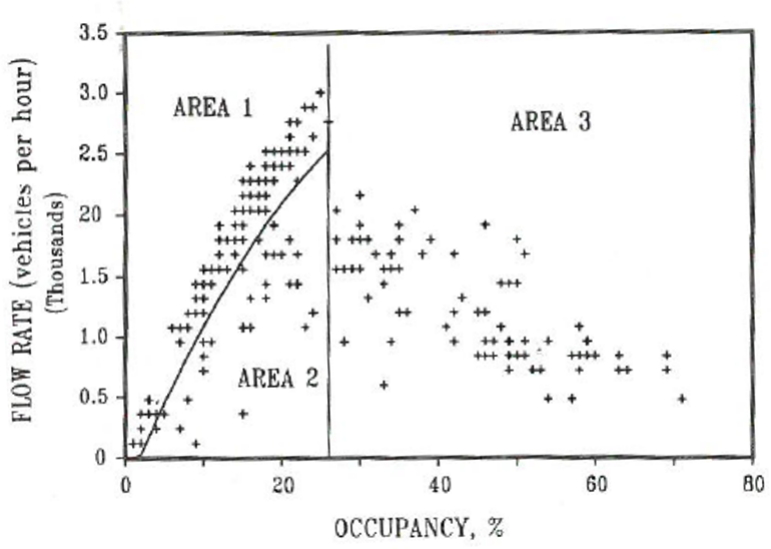


FIGURE 2.3: Defined areas of 30-sec data from Skyway Station NB-7, Ontario, Canada (Persaud et al., 1990).

The algorithm was implemented as a trial on a freeway in Mississauga, Canada. In 39 days, 28 incidents were recorded by operators. The algorithm detected 15 of these incidents, resulting in a 62% detection rate. However, the authors claimed that seven incidents had no effect on the traffic, and hence the detection rate should be 88%. The false alert rate was 0.0012%, and average time to detect was 2.2 minutes (Balke, 1993).

These results were impressive for the time. However, a major limitation of the algorithm was the manual calibration needed to define the areas of the flow/occupancy graph that represent incident conditions. This likely made the method difficult to implement in new locations without effort,

time and expertise to define the areas of the flow/occupancy graph at each location. However, Weil et al. (1998) developed a method to automatically determine these areas using a set of historical data. Persaud et al. (1990) stated that 'the algorithm is basically congestion detection logic' and so can only be used 'where it is not important to identify automatically the cause of the congestion'. As such, further logic would need to be incorporated in order to differentiate incidents from contexts.

2.4.1.4 Bayesian algorithms

Many IDAs have applied Bayes theorem to model the probability that an incident has occurred based on observations of traffic conditions. An advantage of this approach is that prior probabilities can be incorporated, such as an operator's knowledge of the local network.

An early example of a Bayesian algorithm was presented in 1978 (Levin and Krause, 1978). The algorithm used 20 second aggregated values of average speed and occupancy data from loop detectors. This data was processed into seven features, relating to recent changes in average speed and occupancy at pairs of upstream and downstream detectors. For calibration, it required a database of incidents with historical traffic data. It was tested on data from an expressway in Illinois, U.S.A, which contained 17 incidents. It reported a 100% detection rate, 0% false alert rate, 3.9 minutes average time to detect. A version of the California algorithm was used for comparison on the same dataset, it achieved a 100% detection rate, 0.11% false alert rate, and 1.5 minutes time to detect. The stated advantage of the Bayesian algorithm was that rather than just indicating whether an incident was occurring or not, the inferred likelihood of an incident occurring could be indicated, i.e. the confidence of the algorithm's detection could be given to operators.

Zhang and Taylor (2006) presents IDAs on both motorways and urban arterials. Both used occupancy and flow variables as part of a Bayesian network, as shown in figure 2.4 (Zhang and Taylor, 2006).

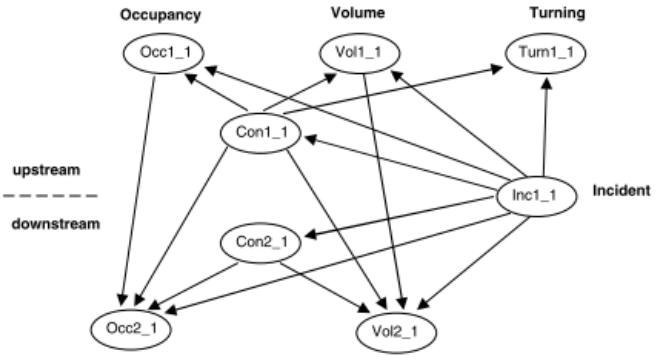


FIGURE 2.4: Basic Bayesian network used for arterials (Zhang and Taylor, 2006).

In the arterial algorithm, raw traffic values of volume and occupancy were first processed with traffic signal data, to form discrete values of low/medium/high volume and occupancy values. Then for both the urban and motorway IDA, a dynamic Bayesian Network was implemented

to detect incidents using the processed volume and occupancy values (as shown in figure 2.4). It should be noted that a different network was used in the urban and motorway IDA. The networks were 'dynamic' in that they used a persistence test, raising alerts only when multiple detectors were indicating the presence of an incident over an entire traffic signal cycle. A number of different complex networks were used, but each detected incidents by using the values of occupancy and flow to move between states in the network, resulting in the probability of an incident occurring. For more information on the mathematics behind Bayesian networks, including the use of conditional probability tables, see (Nielsen and Jensen, 2009). The arterial algorithm was tested on a transport simulator, and achieved a detection rate of 88%, false alert rate of 0.62% and mean time to detect of 178 seconds. Unfortunately, the IDA was not fully evaluated on field data. As such, it is uncertain whether the results achieved are transferable to what could be expected in practice in a TMC. This is because traffic simulators are simplifications of real-world traffic networks, and so the patterns of traffic behaviour in incident and non-incident scenarios may not be transferable.

There also appeared to be many constraints making its real-world implementation difficult. Firstly, the IDA required traffic signal timing data, because otherwise it found it difficult to find incident patterns. Also, the networks' conditional probability tables for the probability between states were initialised using prior knowledge of the networks, meaning expert knowledge would be needed for implementation.

Abdulhai and Ritchie (1999) then used Bayes theorem as a part of a more complex, machine learning IDA. As such, it is described later, in section 2.4.3.

2.4.1.5 INGRID

Bowers et al. (1996) presented INGRID, an IDA to detect incidents on urban networks, particularly on urban streets with traffic signals. Firstly, historical traffic data from loop detectors was stored in ASTRID, an on-line database. This data was averaged, giving more weight to recent data, in order to find 'expected' values of traffic parameters at each detector location. INGRID then detected incidents by comparing real-time values with the historical references retrieved from ASTRID. Alerts were raised when flows were lower and occupancies were higher than the 'expected' values. A final algorithm was used to generate a confidence level of an incident occurring, which increased as the number of detectors effected, consecutive time periods, and variables raising alerts increased. When tested on 540 hours of data from 420 detectors in Southampton, U.K., 100% of 'severe' incidents were detected (i.e. those seen to cause a large amount of disruption to local traffic conditions), 0.77 false alerts were raised per hour, and the mean time to detect was 6.5 minutes. The results suggested that performance was most dependent on the distance between detectors, the level of traffic and the reduction in capacity caused by the incident.

2.4.1.6 RAID

Like INGRID, RAID was a comparative algorithm designed to detect incidents on urban networks with traffic signals (Cherrett et al., 2002). However, RAID differed from INGRID in that it was a single-detector algorithm (i.e. it performed on detectors independently), meaning it did not require a high density of detectors. Although this approach may result in a lower performance than could have been achieved by comparing adjacent detectors, it may be more suitable for implementation across large road networks (with some areas of low detector density). Oskarbski et al. (2016) found that if the location of upstream and downstream detectors are not consistent, single-detector approaches are often found to be more effective than algorithms on upstream/downstream pairs of detectors. RAID used average loop-occupancy time per vehicle (ALOTPV) and average time-gap between vehicles (ATGBV) as parameters to detect incidents (Cherrett et al., 2002). ALOTPV is the average time that each vehicle spends occupying the road space above a loop detector, and ATGBV is the average time period in-between each vehicle occupying this road space. By comparing real-time values with pre-defined thresholds (which were intended to be configured by the traffic operator), RAID could accurately identify when detectors became congested. Congestion could be observed as a step-change in ALOTPV and ATGBV values. Once an alert was raised, it was then the traffic operator's responsibility to identify whether this congestion was caused by an incident or a context, often done by observing CCTV. Although not stated explicitly, IDAs that do not make this differentiation also effectively leave the responsibility of verification with the operator. As part of the PRIME project, RAID did automate part of the verification process by automatically displaying video from the CCTV camera closest to the detector that had raised an alert.

RAID was tested in Southampton, U.K. over five months, on 74 detectors along two urban arterials. Incidents were defined as either vehicle-on-vehicle impacts, vehicle breakdowns, illegal parking or unloading and emergency works. 181 and 334 alerts were raised on each of the arterials respectively, creating detection rates of 69% and 92% respectively. These detection rates were better than those reported in the online test of the McMaster algorithm. When the cause of the false alerts was investigated, it was found that 55% of the first arterial's alerts could be attributed to contexts, including bad weather, special events and football matches. However such alerts were seen to be beneficial because one in every 3.8 alerts resulted in either a radio traffic bulletin being issued, or the change of a Variable Message Sign (VMS). VMS are electronic signs on roadways which inform road users of events such as incidents or congestion. No average time to detect was stated.

2.4.1.7 TRISTAR

TRISTAR was an IDA designed to detect incidents on urban arterials (Oskarbski et al., 2016). From Bluetooth and Wi-Fi scanner probe vehicle data, vehicle journey times were calculated and used as input to the algorithm. Historical reference values were then determined for different times and types of day (e.g. weekday, bank holiday etc.). Recent observations of average travel times were compared to these historical reference values, and alerts were raised using an algorithm based on the Kalman filter when a sufficient difference was recognised. The threshold of this difference could be modified by the traffic operators using them. TRISTAR was constrained

in that its mean time to detect could be as high as 15 minutes when detecting incidents on long road sections. This is because the algorithm used journey time data, and so could not detect an incident until the vehicles effected had travelled from the first to the second detector, a problem particularly when a road section is completely blocked. Other challenges included the penetration rate and aggregation of the devices detected (e.g. many devices detected on a bus at the same time). In their pilot implementation, approximately 30% of vehicles could be detected, and a classification scheme was used to identify each type of vehicle passing. Such challenges will be common amongst all IDAs that use journey time data from Bluetooth detectors.

The majority of early IDAs were comparative, and this type of algorithm is still being used in recent years. As such, large numbers have been developed, tested and compared. Typically, the approach for comparative IDAs designed for motorways and urban networks is the same. However, in more dense urban environments such as streets and junctions, comparative algorithms require significant manual calibration and are typically less effective, because of the complexity of urban road networks and diversity of traffic patterns. A disadvantage of comparative algorithms' fixed threshold approach is that temporal variations, such as traffic signal noise, may go unaccounted for, leading to unnecessary false alerts. Another limitation of many IDAs is that they aren't able to differentiate incidents from contexts (such as sporting events or planned roadworks). This means that many false alerts are generated when implemented, particularly in urban road networks, causing such algorithms to fall out of favour with many traffic operators (Parkany and Xie, 2005). Using fixed thresholds on traffic parameters to raise alerts may be effective in identifying queuing or slow moving traffic, but it is much more difficult to infer the cause of the congestion. RAID avoided this by making it the operator's responsibility to differentiate incidents from contexts. Clearly, these IDAs could be improved upon if they were able to make this distinction automatically.

2.4.2 Time-series

2.4.2.1 Standard normal deviate

One of the earliest and simplest IDAs was the standard normal deviate algorithm (Dudek et al., 1974). The algorithm was developed for motorways, and used occupancy data to detect the 'shock wave' (i.e. sudden change to lower speeds) in traffic caused by incidents. Dudek et al. (1974) tested a number of different values of parameters, but occupancy was found to produce the best results.

First, the mean and standard deviation of occupancy values for the past five minutes were stored. The standard normal deviate (SND) was then calculated as:

$$SND_{n+1} = \frac{X_{n+1} - \bar{x}}{s} \tag{2.4}$$

Where X_{n+1} is the next occupancy value, $\bar{x}$ and s is the mean and standard deviation of the last five minutes occupancy values. Incident alerts were raised when the SND was above a certain

threshold for two consecutive time periods. The time period used was one minute.

The algorithm was tested on loop detectors on the Gulf Freeway, Texas, U.S.A. 35 incidents occurred in the time period. A 92% detection ratio, 1.3% false alert rate (defined as false alerts per time period per detector during peak hours), and average time to detect of 1.1 minutes was reported. This method detected abnormally high values of occupancy, which would indicate queuing traffic, which would indicate the occurrence of an incident. However, as stated in Dudek et al. (1974), although the 1.3% false alert rate appears low, 'the number of false alarms can become very significant in an operational system' (Dudek et al., 1974). This high rate may be because the IDA only uses occupancy values, and so can only detect congestion, rather than differentiating incidents. It would also not be able to detect incidents upstream of detectors (where low flows may occur), and did not take into account spatial patterns to detect incidents (e.g. nearby detectors raising alerts raising likelihood of an incident occurring).

2.4.2.2 Double exponential smoothing

Cook and Cleveland (1974) presented the double exponential smoothing IDA. It was similar to the standard normal deviate (see section 2.4.2.1), but used a more complicated forecasting method. The target variable values were first 'smoothed' so that recent observations were weighted more heavily than less recent observations, making the algorithm less sensitive to variation from noise. Then, the next value of the target variable is forecasted using double exponential smoothing (for more details, see (Brown, 2004)). If real-time values differ from this forecast by a sufficient amount, an incident alert is raised. The algorithm was tested with 13 different variables, but found volume and occupancy yielded the best results. The algorithm was tested on a freeway in Los Angeles, U.S.A., that had 50 incidents over a 13 month period, it had a 42% detection rate with false alert rate of 0%, or 100% detection rate with 8% false alert rate, depending on how the parameters were set, which altered the sensitivity of the algorithm in raising alerts. Most incidents were detected within one minute of the onset of congestion after the incident.

2.4.2.3 Low volume

In 1975, an IDA was designed specifically for low volume traffic conditions on motorways (Dudek et al., 1975). The algorithm was one of the first to detect incidents using a rudimentary form of journey time.

Using pairs of detectors, when a vehicle passed over the first detector, its speed was used to estimate an expected time of arrival at the second detector. A range of possible times for each vehicles was estimated using the fastest possible speed (assumed to be 100mph), and a potential reduction in speed of 10%. Using this logic, a range of the expected number of vehicles arriving at the second detector over the next 30 second period was calculated. If the actual number of vehicles fell outside this range, an alert would be raised. The IDA was tested on a computer simulation program under a number of different scenarios. With a 300 meter detection spacing at 500 vehicles per hour, the IDA detected 86% of incidents, with an average time to detect of

one minute.

This IDA would only be suitable for pairs of detectors on sections of road where no entries or exits existed. As such, its operation is limited to motorway networks with a high density of detectors. It was concluded from the simulations that the IDA would be suitable for 3 lane motorways, with 300 meter detector spacing, for flows of up to 500 vehicles per hour. When implemented, the IDA was less suitable in higher flow conditions, because with more vehicles passing the second detector, higher times to detect would be needed (it is more difficult to estimate the expected arrival time under high flow conditions). The IDA also struggled with vehicles changing lanes, and detecting trucks, which could be counted as two vehicles.

2.4.2.4 ARIMA

Auto-Regressive Integrated Moving-Average time series (ARIMA) models use recent observations of a traffic variable to create a prediction of its 'expected' value (i.e. conditions that would occur if no incident occurred) in the near-term future. If real-time values significantly deviate from this prediction, an incident alert is raised.

ARIMA was used to detect incidents using occupancy data on freeways in Detroit, U.S.A. location (Ahmed and Cook, 1982). 95% confidence intervals were used for the predictions of occupancy, and incident alerts were raised when real-time values were outside this interval. The IDA was tested on 50 incidents on the Lodge Freeway, Detroit, Michigan, U.S.A. The IDA had 100% detection rate, 1.4% false alert rate, and a time to detect incidents of 23 seconds, and false alert rates of 2.6%. Here, the false alert rate was defined as the percentage of false incident messages to the total number of incident messages generated by the algorithm.

ARIMA models are commonly found to be effective in forecasting traffic variables in the short-term future. However, the forecast would not be of 'expected' traffic conditions if the recent observations are influenced by incidents. When used in an IDA, this could lead to incidents going undetected. The model is also known to be less effective during sudden changes in traffic parameters, e.g. rush hour. As part of an IDA this leads to many false alerts being created, which has meant ARIMA models have fallen out of favour in recent years (Martin et al., 2001). Pan et al. (2015) found improved forecasting accuracies when combining an ARIMA model with a historical average. Using a decision tree, the historical average predictor was typically used at times of sudden change in traffic conditions, such as during rush hour. However, this ARIMA model would still not account for other contextual causes of sudden changes in traffic variables, such as sporting events. This combined model was not developed into an IDA.

Thancanamootoo and Bell (1988) presented one of the first IDAs designed specifically for urban networks. It used volume and occupancy data to detect incidents between pairs of upstream/-downstream detectors. An exponential smoothing scheme was used to set threshold 'expected' values of volume and occupancy on each detector. An alert would then be raised if real time flow and occupancy values fell below their thresholds on the downstream detector, and flow fell below

and occupancy rose above its threshold on the upstream detector. Results from simulation and field tests reported that the IDA had a 'lack of robustness', i.e. performance (accuracy, time to detect and false alert rate) varied significantly across different locations of incidents with respect to the detector pairs. Difficulty in differentiating incidents from contexts was also reported (Parkany and Xie, 2005).

Sheu and Ritchie (1998) presented a modified sequential probability ratio tests algorithm for use in urban networks. It included three procedures, a knowledge-based rule set for identifying the symptoms of an incident, signal processing for real-time prediction of incident-related traffic conditions and pattern recognition for incident detection. As well as detecting incidents, the IDA also claimed the ability to estimate lanes blocked, queue lengths in blocked lanes and the duration that an incident would disrupt local traffic conditions (although this may not be possible in real-world implementations). Unfortunately, this algorithm was only tested on simulated incidents, and so it is unclear whether this approach would be suitable for real urban networks.

Lee and Taylor (1999) also detected incidents on urban streets, but by applying a Kalman filtering algorithm to find sudden changes in traffic variables on a two-lane arterial's detectors. This approach was designed to be simple and require little calibration, while being dynamic enough to account for traffic signal noise. First, a discrete-time linear Kalman filter was used to forecast detector's speeds and flows with upper and lower limits. Then if any observed values fell outside of these limits, an incident alert would be raised. Although shown to detect an incident successfully, not enough data was available for a thorough performance evaluation. The algorithm only took recent traffic observations as input, meaning changes in traffic variables caused by contexts, may cause false alerts.

The main factor differing time-series from comparative algorithms is that the threshold used to raise incidents will vary based on recent local conditions. This gives an advantage because it means temporal variations (such as peak periods) can be more readily accounted for automatically, and less manual calibration is required. However, each time-series IDA reviewed only used inputs of traffic data for their forecasts, and so contexts occurring at irregular intervals (such as football matches) could not be accounted for.

2.4.3 Machine learning

2.4.3.1 Neural networks

Neural Networks take inspiration from the human brain in that they represent a system of 'neurons' which exchange data between each other to 'learn' how to do a particular task. They are best used in estimating functions that depend on a large number of unknown inputs. Over the past 30 years they have been increasingly used in an expanding number of applications, including natural language processing, image recognition and incident detection.

In the early 1990s, the first neural network approaches to incident detection were presented. Ritchie and Cheu (1993) presented a multi-layer feed-forward neural network for incident detection (the first and simplest type of neural network). It compared results for a number of different traffic variable inputs to the network. The best performing network used 13 hidden layers, and took inputs of the real-time upstream volume and occupancy, and the downstream volume and occupancy for real-time and the last two time periods (1 minute). The IDA was trained and tested on datasets from a simulation of a motorway, each with 123 incidents. It had a 100% detection rate, 0.012% false alert rate, and average time to detect of 69 seconds. A version of the California algorithm (described in section 2.4.1.2) was also implemented for comparison, which achieved 98.6% detection rate, 0.19% false alert rate, and 130 seconds average time to detect. As the algorithm was not tested on field data, it is unclear whether the IDA had over-fit (i.e. fit its model too closely on a limited dataset) to the patterns found in the training simulated dataset. Another clear drawback of this algorithm is that a large number of incidents with traffic data are needed for training, meaning that in practice either large amounts of historical data would need collecting before implementation, or entire networks would need to be created in a simulator to train the algorithm.

A similar multi-layer feed forward neural network was developed and tested on field data in 1997 (Dia and Rose, 1997). The loop detector dataset used was from the Tullamarine and South Eastern Freeway in Melbourne Australia. It had 475 incidents, but only the 100 that caused visible disruption to the detector data were considered. 60 were used in training, 40 in testing (believed to be the largest set of incident data at the time). The algorithm used three hidden layers, and took inputs of real-time values of average speed, flow and occupancy (aggregated over 20 seconds) from upstream and downstream detectors. It used a persistence test of two time periods to detect incidents. The evaluated algorithm had a 82.5% detection rate, 0.065% false alert rate, and 203 seconds mean time to detect. 1456 of 2171 false alerts were found to be from the algorithm mistaking incidents for 'rubber necking' or congestion from other causes. These results show that neural network models can be trained and implemented on field data, but it is clear that large training datasets of incidents are required to do so. This stems from the algorithm's approach to 'learning' what traffic conditions could be expected in both incident and non-incident scenarios.

In later years, progress in machine learning research resulted in more complex types of neural network, which in turn have been developed as incident detection algorithms. Abdulhai and Ritchie (1999) developed a Bayesian-based probabilistic neural network IDA. The IDA used inputs of the past 2.5 minutes of volume and occupancy values (aggregated over 30 seconds) from upstream and downstream loop detectors. When tested on a simulator, and loop detectors on a California freeway, it showed a similar performance to a multi-layer feed forward neural network in term of detection rate, false alert rate and mean time to detect. However, the Bayesian neural network took less time to train, and it could re-train continuously, meaning manual re-calibration of the algorithm wouldn't be needed if traffic conditions at a site were to change.

Khan (1997) presented a modular neural network to detect incidents on urban arterials. The idea

of the modular network was to develop a neural network for different tasks, detecting lane blocking incidents, special events incidents and detector malfunction. The outputs of these algorithms were then combined using a gating network. When tested on loop detector data from Anaheim, California, U.S.A, the algorithm performed better than a multi-layer feed forward network. It achieved an 80% detection rate with 0.7% false alert rate. As well as improved performance, an advantage of modularising the problem is improved understanding for operators of the situation, i.e. is there an incident occurring or has the detect malfunctioned.

Khan and Ritchie (1998) presented a modular neural network algorithm to detect incidents on signalised urban arterials. Modular neural networks decompose the task of the neural network to several modules. These modules are in themselves neural networks, which each solve a sub-task of the problem with the best possible architecture. A gating network then combines these results and outputs the predicted incident state. Modularising the problem allows the network to be trained more quickly and provides a more understandable representation of the problem. The modules were lane blocking incidents, special event incidents and detector malfunction. The algorithm was tested on simulated incidents that used loop detector data from Anaheim and Los Angeles, California, USA. After comparing their modular network with a multi-layer feed-forward neural network and a projection neural network, they found that the modular network achieved the best results. For field data the modular network achieved an 80% detection rate with 0.7% false alarm rate. They also found that their IDA performed best during higher flows and on incidents with greater queue lengths. However, it is unclear whether these results are replicable on real incidents, and how much calibration is required to implement the algorithm in practice.

2.4.3.2 Fuzzy logic

Fuzzy logic has been used to detect incidents on signalised diamond interchanges (Lee et al., 1998). Fuzzy logic is an effective method for models that require approximate reasoning, require real-time operation, and exhibit uncertainty. The objective of the IDA was to provide incident alerts that would aid traffic operators in adjusting signal control strategies. The IDA consisted of four modules, normality inference, incident location inference, incident severity assessment and incident termination inference. It took inputs of lane-by-lane volumes, occupancy and speed, each derived from a video system. The IDA was tested on a simulated diamond interchange. It detected 74 of 102 simulated incidents (73%), produced 34 false alerts from 6120 decisions (0.56%), and had a mean time to detect of 4.1 minutes. Although the model was designed and tested on signalised diamond interchanges, it was claimed not to be limited to such locations (although results on other road types were not presented). It is unclear whether the results stated would be replicable on field data, and if so how much calibration would be involved in implementation.

2.4.3.3 Support vector machines

Support vector machines (SVMs) classify points by forming an 'optimal' hyperplane between training data points. In testing, new data points are classified based on the side of the hyperplane they fall, in this case incident or non-incident. The learning of the algorithm to form the hyperplane is done using kernels, in a similar way to neural networks. Steinwart and Christmann (2008) provides further background on the mathematics involved in support vector machines.

In recent years, SVMs have become more commonly used in incident detection. Yuan and Cheu (2003) used an SVM to detect incidents on sections of urban arterial with high loop detector coverage. The inputs to the algorithm were the real-time and three previous intervals (1.5 minutes) of flow and occupancy for six nearby detectors. It was tested on 30 second aggregated loop detector data from a freeway in the San Francisco Bay area, California, U.S.A. The dataset had 45 incidents, 22 of which were used for training. The best results were achieved when using a polynomial kernel, with a 91.3% detection rate, 0.13% false alert rate, and 4.5 intervals (2.25 minutes) average time to detect. A multi layer feed-forward neural network was used for comparison, which achieved 82.6% detection rate, 0.06% false alert rate and 6.5 intervals (3.25 minutes) average time to detect. The Bayesian neural network developed by Abdulhai and Ritchie (1999) was also implemented (described in section 2.4.3.1) for comparison, which achieved a 95.6% detection rate, 0.3% false alert rate, and 7.7 intervals (3.85 minutes) average time to detect. This shows that SVMs are competitive in terms of performance with neural networks.

Support Vector Machines (SVMs) have also been developed for urban networks. An SVM presented by Yang et al. (2009) used inputs of volume and occupancy from upstream, downstream and 'medium' loop detectors for each link (i.e. at the start, mid-point and 50-100m from the end of each link), average speeds from probe vehicles, and individual reports from travellers on the road network. The algorithm was tested on both field and simulated data. The IDA outperformed a multi-layered feed forward neural network in terms of detection rate, false alert rate and average time to detect on both field and simulated datasets.

In recent years, the same types of machine learning approaches (neural network, bayesian, fuzzy logic) have been developed into more complex IDAs for both motorways and urban areas, but the types of approaches have remained the same. Ghosh and Smith (2014) customised four machine learning based motorway IDAs for use in urban areas with signalised junctions, namely a multilayer feed-forward (MLF) neural network, a probabilistic neural network, a fuzzy-wavelet radial basis function neural network, and a SVM. The IDAs were tested on simulated data from an urban network with signalised junctions. Each IDA used inputs of flow and occupancy from loop detectors. It was found that the SVM provided the best performance, closely followed by the MLF neural network. However, the MLF neural network was found to be far less computationally intensive in training and easier to implement, and so was recommended as the most suitable choice for implementation.

2.4.3.4 Discussion

The typical assumption made by these machine learning algorithms is that traffic conditions vary between incident and non-incident periods, and the distinction between the two can be 'learnt' from historical traffic data. Once the distinction is 'learnt' from training data, alerts are raised when real-time data appears to be most similar to incident conditions (Khan and Ritchie, 1998, Lee et al., 1998). Due to the fortunate rarity of incidents, this approach requires collection of large amounts of data to obtain sufficient samples of incident conditions. As such, algorithms with this approach are typically trained on incidents in a transport simulator. It is unclear whether this means that when implemented in a new location, a simulated network would have to be created first. If this is the case, such algorithms may require too much calibration to be feasibly implemented across large road networks. As was discussed in section 2.3.2, simulated data can be unrepresentative of traffic conditions that can be expected in real-world networks, and so the performance of IDAs trained on simulated data may not generalise to the performance in TMCs.

Machine learning IDAs state some of the best results in terms of average time to detect, detection and false alert rates. Such algorithms aim to 'learn' the conditions of an incident, and so may be able to differentiate incidents from contexts. However, unfortunately few machine learning IDAs have been thoroughly evaluated and compared because so few have been implemented on field data. Further work is required to verify that the stated results and abilities are achievable when implemented on real road networks.

2.4.4 Video-image processing

Kamijo et al. (2000) designed a hidden Markov model IDA specifically for urban intersections using CCTV imagery. It dealt with the problem of occlusion among vehicles by using a Markov random field to help its tracking. The algorithm tracked between 93% and 96% of vehicles at junctions, and was demonstrated to be feasible for incident detection. However, only a small number of incidents occurred during their observation period, so further work is required to determine the validity of their method. Ki (2007) also used a tracking algorithm to detect, record and report traffic incidents at junctions. However, only four incidents occurred during their test period, so further testing would be required for a full evaluation.

Zou et al. (2009) also presented a hidden Markov model classifier, but this time was tested on 500 samples of video. After extracting image sequences from a traffic surveillance system on an intersection in China, compressed features were generated through several image processing steps. Then the hidden Markov model was used to detect incidents. When tested, the algorithm achieved an 84% incident detection rate. The false alert rate and mean time to detect was not stated, and it is unclear if such results would be replicable on other junction and urban road types.

Recently, image processing algorithms have been implemented commercially, demonstrating their viability (TrafficVision, 2017). As well as having high incident detection performance, their

approach holds some advantages over traffic variable based algorithms (i.e. those based on values of variables rather than images or sounds). For example, traffic variable based IDAs rely on incidents causing traffic disruption to be detected, but this disruption may be small in heavy or light traffic conditions (Cherrett et al., 2002). As image processing IDAs do not rely on such disruptions, its unlikely they'll suffer from this issue, and they'll be less susceptible to mistaking incidents for traffic signal noise or contexts. They also have the ability to highlight the exact location of incidents to operators. However, image processing algorithms are unlikely to ever form the entire solution to incident detection because few areas have CCTV coverage dense enough to rely on them entirely.

2.4.5 Audio processing

Bruce et al. (2003) explored the use of digital audio signals for incident detection algorithms on intersections. They found that audio signals are a cost effective, computationally efficient data source. To train and test the algorithm, 66,176 3-second samples of audio data were collected at intersections using a tape recorder, which were combined with further crash audio data. Many feature extraction and incident detection approaches were compared. Wavelet transforms were used for feature extraction, and a maximum-likelihood classifier was found to be best at determining the traffic state. When tested on their audio samples, a dynamic wavelet transform was found to be the most suitable technique for feature extraction due to high accuracy and computational efficiency. When used with a maximum likelihood classifier to detect incidents, accuracies of over 95% were found. Further evaluation is needed to assess this approaches usability, such as the number of microphones required, and the performance around background noise such as weather or construction.

2.4.6 Social media

Social media has been used more and more frequently in recent years as a platform to report recent incidents occurring. Reports of incidents come most commonly from public authorities, media companies, and road users themselves (Gu et al., 2016). Social media is now being used by some TMCs as an additional incident detection data source for operators to monitor. This section describes research undertaken to explore whether social media data could be used as an input to an IDA. Such an IDA would need to obtain social media messages in real-time, encode the data into a machine-readable format, then determine whether the message was indicative of an incident at a particular location, and whether it was reliable. This determination may be possible with a machine learning algorithm trained on historical social media messages.

Gu et al. (2016) attempted to detect incidents from Twitter data on all types of road in Pittsburgh and Philadelphia, U.S.A. Incident related tweets were extracted, and the time and location of incidents were inferred. This extraction process filtered out posts that were judged not to provide the location, road direction or incident situation well enough. These incident related tweets were compared with incident data from the local TMC's incident logs, and 911 Call For Service data. 71% of the 'incident related tweets were within 30 minutes and one mile of the

incident logs. This method was not developed into an IDA that could detect incidents in real-time. Incident related tweets were reported far more frequently in the day (particularly during peak hours), and in the centre of the cities (due to the greater volume of messages). As such, it was suggested that Twitter based incident detection would have a lot more coverage in urban networks.

Of the incident related tweets, it was stated that 60-70% were posted by public agencies or the media. Another common source was from road users themselves. The media and public agency tweets are simply second-hand data sources from others' incident detection methods. These sources may be useful to TMCs as the social media companies are effectively providing a platform to share incident reports. However, the only new incident detection data source provided by social media companies appears to be the posts from road users themselves. Such posts are only useful if the location, road direction, situation and time of incident are described. Also, as many countries now ban the use of mobile phones whilst driving, travellers' data is becoming reliant on passengers. In England, the average car/van occupancy has fallen to 1.55 in 2016 (62% single occupancy rate), and so the amount of incident related social media data from travellers may be falling (UK Government, 2017). However, this trend may be reversed by the arrival of fully autonomous vehicles in the future. As such, there is an unanswered question over the current and potential significance of road users' incident related posts on social media platforms for incident detection.

Nguyen et al. (2016) explored social media IDAs by filtering incident related Twitter data on all roads across New South Wales, Australia, and presenting the most relevant tweets to TMC operators on an online map. The methods used to filter incident related tweets (a message on Twitter) and determine their usefulness to operators was done with a variety of machine learning algorithms. Training of these algorithms was done with the help of 5000 historical tweets which were manually labelled by TMC operators as relevant or not relevant. They tested their algorithm by comparing the tweets raised to operators with TMC incident logs. The results showed four case studies of incidents, in which the tweets were raised many hours earlier than logs were created by the TMC. However, these results suggest that the incident detection methods in the TMC were far slower than other techniques in this review. A comparison of the presented algorithm to other state of the art IDAs would be needed for a full evaluation. The three Twitter accounts that posted the most incident related tweets were stated. These were a public authority, a local radio station, and a charity that provides a radio network manned by volunteers, which receives radio calls from road users, and reports the findings of these calls on Twitter.

It appears that an IDA based on social media data would have lesser performance in terms of detection rate, false alert rate and average time to detect. Difficulties include the time between incidents occurring and reports being made, the requirements for training, and the coverage of incidents. However, it may provide additional information that other sources could not, such as the cause of an incident (such as 'oil spill covering two lanes...'), where congestion is propagating, or which diversion routes were being taken. Social media IDAs also have the advantage that they can detect incidents across entire road networks, and so have more coverage than IDAs based on road-side detectors. It could also be used for sentiment analysis of travellers reactions to

incident disruptions, which could be used for evaluation and planning of future incident response strategies. As such, social media based IDAs appear best used to complement other IDAs based on conventional sources (such as loop detectors, or videos of traffic).

2.4.7 Data fusion

Westerman et al. (1996) developed an algorithm that incorporated data from probe vehicles and loop detectors to detect incidents on motorways. Many separate algorithms were used as part of the compound IDA. Firstly, an algorithm using solely probe vehicle data and an algorithm using solely loop detector data would each have to indicate the presence of an incident in a certain location. If both did, three further incident detection algorithms would use inputs of both data sources to verify the presence of the incident. The outputs of these algorithms would be combined to form a probability that an incident truly had occurred, and if this percentage was above a certain threshold, an alert would be raised. The loop detector algorithms were modified versions of the California algorithm, and probe vehicle algorithms compared real-time average vehicle speeds to a historical average. The algorithm was not tested on field data, but initial results on a simulator indicated that the compound algorithm improved on the algorithms used on just one data source.

Ivan (1993) designed an urban IDA that combined three separate algorithms from different data sources using a data fusion process. The individual algorithms used data from probe vehicles, fixed-detectors and anecdotal sources. The probe and fixed-detector algorithm compared real-time data to historical averages. The fixed detector algorithm used occupancy data from inductive loop detectors, whereas the probe algorithm used travel times from probe vehicles. Real-time values significantly different from historical averages would result in the algorithm indicating a greater likelihood of an incident downstream. The anecdotal source algorithm used data from emergency dispatch reports (e.g. from ambulances and the police), travellers cell phones, emergency patrol vehicle reports, and construction reports. The algorithm interpreted this data and computed the likelihood of an incident given the source, content and number of reports received. A final data fusion process was used to combine the output of the three algorithms, resulting in an estimated likelihood of an incident at a particular location. The process weighed up the reliability, precision, validity and ageing rate of each algorithm's output. The IDA was not evaluated.

Ivan et al. (1995) presented a similar, but more complex urban IDA than above. The IDA was made up of two algorithms, one on fixed detectors and one on probe vehicle detectors, each implemented on sections of road with probe vehicle coverage and a fixed detector present. However, each of the algorithms were multi-layer feed-forward neural networks, with one hidden layer of five units. The outputs of these algorithms, incident or no incident, were used as input to a data fusion process. This process was an identically structured neural network, which outputted the final incident or no incident state. The input to the fixed detector algorithm was the ratio of the detector's current flow to the average flow under non-incident conditions over the training period, and the same ratio but for average speed. The probe vehicle algorithms' input was the same ratio but for average travel time over the road section. The algorithm was trained and tested on simulated urban streets and arterials. This IDA was compared with each of the individual

algorithms, and a multi-layer feed forward neural network with a single hidden layer of five units, which simply took inputs of the three ratios (travel time, flow and average speed), and outputted the incident state. This comparison was made to determine whether IDAs would perform better by using inputs from multiple data sources, or by using algorithms on data sources separately and then combining outputs. The neural network which took inputs from all data sources was found to offer no improvement over each of the individual algorithms on single data sources. But the IDA that combined the individual algorithms' outputs using the data fusion process performed 'much better'. This IDA was reported to achieve a 100% detection rate and 0% false alert rate, no time to detect was reported. However, as this IDA was only tested on a simulator and was not compared with other state of the art IDAs, it is hard to draw conclusions from the stated results.

In theory, data fusion algorithms could achieve more accurate and reliable incident detection performance by combining many types of data and/or algorithm. However, the trade-off of this benefit is that more algorithms need developing and implementing, meaning that compared to using just one algorithm, a greater complexity, cost (to develop and implement) and calibration time is required. Also, if the final output of the IDA relies on the output of many individual algorithms, it will be as slow as the slowest individual algorithm, or slower if there is a combination process at the end, hence such IDAs will be slower than the individual algorithms.

2.4.8 Research addressing incident detection limitations

From the review of state of the art IDAs undertaken, it is clear that there still exists a number of outstanding limitations that have not been solved. However, some research studies have identified and specifically addressed these limitations. The following subsections describe these research studies, and discuss whether each identified limitation has been resolved.

2.4.8.1 Accounting for traffic signal noise

Few IDAs have been designed for use in urban networks, and even fewer for signalised junctions. However, many (and often the most detrimental) urban incidents occur at or nearby junctions, resulting in large amounts of disruption to the surrounding traffic network (Zhang and Bruce, 2004). The closer a detector is to a signalised junction, the more noise is caused from its traffic signals, making incident detection more challenging. Ghosh and Smith (2014) observed that traffic signals at stop line detectors can create incident-like conditions. In particular, low flow and high occupancy was observed during red periods of traffic signals. Such behaviour can make urban IDAs prone to creating false alerts by mistaking incidents for traffic queuing at signals.

Some of the earliest IDAs considered the importance of noise caused by traffic signals. Thancanamootoo and Bell (1988) calculated their thresholds by finding the means and variances of volume and occupancy at the end of each signal cycle. Bowers et al. (1996) took traffic signal timings into account by comparing real-time traffic data with historical data at the same signal timing.

Some research has addressed the challenge of traffic signal noise by smoothing real-time traffic data before running IDAs. Ghosh and Smith (2014) matched signal timing data to adjacent stop line detectors, and artificially changed the values of flow and occupancy at each detector depending on whether the time-stamp was during a red or green phase. This modified data was then used as input to test state of the art IDAs that were originally designed for motorways. The IDAs tested included a support vector machine and three types of neural network, probabilistic, multi-layer feed-forward and a fuzzy-wavelet radial basis function neural network. It was found that the original IDAs produced high false alert rates in urban networks, but this rate slightly improved when the IDAs were modified using the traffic signal data.

An extension of TRISTAR (discussed in section 2.4.1) was developed with aim of making the IDA feasible for use on signalised junctions (Oskarbski et al., 2016). To do this they ran simulation studies to understand the variability of incident congestion at junctions. Oskarbski et al. (2016) used traffic and traffic signal data to argue that when a stop-line detector's occupancy remains sufficiently high during a green cycle, vehicles must not be able to enter the junction because either congestion at a junction exit restricts it, or an incident has occurred at the junction. The former could be indicated by high occupancy values at the detectors at the exit of the junction downstream. Simulation results showed how incident detection at junctions could be improved by considering traffic signal data, but it is unclear how much further calibration would be required to implement this in the field. The results also indicated that TRISTAR would raise many false alerts in urban streets at peak times, i.e. the IDA wouldn't differentiate incidents from contexts.

The most effective method found to account for traffic signal noise appears to be to use traffic signal data as an additional input (Zhang and Taylor, 2006, Stephanedes and Hourdakis, 1996). This method is often done with the aim of gaining a basic level of understanding of the context surrounding detectors. If appropriate insights can be gleaned from this extra data, IDAs could perform better throughout urban networks. However, this comes at the cost of requiring such data to be collected at each implementation location. A traffic operator would benefit most from an IDA that was designed to account for traffic signal noise, but did not require traffic signal data. This could be done by using a duration period longer than a signal cycle, or 'learning' the pattern of traffic signal cycles.

2.4.8.2 Differentiating incidents from contexts

Despite the evolution of IDAs, there still appears to be an outstanding challenge in differentiating incidents from contexts. In urban networks this is especially difficult, because their complex topologies serve diverse travel modes and purposes, meaning traffic behaviour during incidents and contexts are often less predictable. Many urban IDAs have high false alert rates, often due contexts, which has led to dissatisfied traffic operators (Parkany and Xie, 2005). With the ability to differentiate incidents from contexts, IDAs could improve by producing fewer false alerts.

Many of the IDAs discussed rely solely on traffic data as inputs to detect incidents. Therefore, to be able to differentiate incidents from contexts, these IDAs must make the assumption that

traffic data during incidents is characteristically different than during contexts (Payne and Tignor, 1978, Persaud et al., 1990). On motorways, it was found that when incidents and congestion occur, an upstream drop in flow and rise in occupancy is observed. This change is accompanied by a drop in upstream average vehicle speed, which is more sudden when incidents occur (Gall and Hall, 1989, Persaud and Hall, 1989). Gall and Hall (1989) also found that capacity is maintained downstream of incidents, but volume is reduced. Hence, low occupancy and flow are observed. During contexts, however, the entire congested area exhibits similar conditions, but areas downstream of the congested area show high flows and occupancies (higher than during incidents). By contrast, Cook and Cleveland (1974) found that for motorways' flows and occupancies, the conditions displayed during incident and non-incident situations were not exclusive. Many IDAs tested on field data in urban networks also found this. Cherrett et al. (2002) found that many of their IDA's false alerts came from contexts, which displayed similarly congested conditions to incidents.

Of the IDAs that assume that incidents and contexts are different in terms of traffic conditions, some make the assumption that queuing will occur upstream of an incident but not downstream (Payne and Tignor, 1978, Persaud et al., 1990). These algorithms attempt to make the distinction by utilising comparisons of upstream and downstream detectors. However, this approach requires the identification of pairs of detectors, and would not be possible in more complex or less dense networks where such pairs do not exist. Payne and Tignor (1978) assumed that congestion occurred more abruptly during incidents than during contexts, and so developed their IDA to only raise alerts when congestion occurred abruptly. The IDA developed was similar to the California algorithm, but used a decision tree based on this logic. A test on field data with 118 incidents revealed that the developed IDA outperformed the California algorithm in terms of detection rate and false alert rate (although exact values were not stated). Cherrett et al. (2002) focused on detecting congestion only, and left the responsibility with the traffic operator to identify whether detected congestion was caused by an incident or a context.

It appears that accounting for contexts is still an outstanding limitation with state of the art IDAs. Many attempts have been made to account for contexts, but it is still a limitation found by IDAs tested on field data. As such, further research is required to better differentiate incidents from contexts. This would significantly improve IDAs' performance by limiting the number of unnecessary false alerts being created.

2.4.8.3 Estimating incident location and disruption propagation

Many IDAs simply raise alerts that an incident may be occurring in the vicinity of a detector's location. Some algorithms on fixed detectors raise alerts of incidents occurring between upstream and downstream detectors. Ideally though, an IDA would be able to infer the exact location of an incident, and even estimate the propagation of its disruption. This would allow for faster incident verification, and improved response strategies, particularly in situations where verification is not possible (e.g. no CCTV available) (Lee et al., 1998, Hawas and Mohammad, 2015).

To achieve this, Lee et al. (1998) presented an incident location inference module as part of their their IDA designed for signalised diamond interchanges. Once an incident had been detected, the location was estimated by comparing observed traffic data against expected incident patterns. Hawas and Mohammad (2015) detected incident location by looking for inconsistencies in traffic flow between nearby loop detectors. They found that for accurate location inference, a high density of detectors was needed for the comparisons. Unfortunately, the effectiveness of both of the proposed methods was not validated on field incident data.

In the literature, many models have been presented which attempt to understand and predict how congestion propagates (Saeedmanesh and Geroliminis, 2017, Hu et al., 2009). However, there are few that are specific to understanding how congestion propagates during incidents. Literature that addressed this issue could be useful in developing an IDA that could not only detect incidents, but also predict their disruption in the surrounding area. This tool could help TMC operators to pre-emptively take measures to reduces incidents disruption.

The design and effectiveness of incident location and congestion propagation features appears largely dependent on the type of algorithm and data used. For example, comparative algorithms that use upstream/downstream pairs of loop detectors could detect which pair an incident occurred between, then approximate where between the pair it occurred. Whereas image processing IDAs could detect the exact incident location, but only when incidents occur within view of CCTV cameras (location inference is otherwise not possible). When IDAs are developed, the priority is typically to maximise performance and reliability, meaning these features become secondary priorities. However, it is found that methods exist to incorporate these feature within existing IDAs, allowing IDAs to improve their usability for traffic operators.

2.4.9 Conclusions

This chapter reviewed the progress made in designing an effective IDA. Limitations with presented IDAs were highlighted, and research aimed at addressing these limitations were discussed. This review shows that incident detection is not a solved problem. Despite progress being made through the years, state of the art IDAs are found to still have outstanding limitations.

Several IDAs have been presented in the past few decades, but very few of the IDAs reviewed have been implemented for use in real networks. Due to the difficulty in obtaining sufficient real traffic and incident data, many have only been tested on simulators. Hence, it is currently difficult to evaluate the true performance of urban IDAs. Significant research is needed to draw conclusions of the performance and usability of many of the IDAs reviewed. It is also difficult to directly compare the performance of IDAs because of the variety of datasets used. Comparisons could be made more easily if more datasets and codes were published. Further progress could be made with the use of a platform to post data and algorithms, allowing researchers to share, compare or compete their IDAs (see (Kaggle, 2017, ImageNet, 2017)).

Of the IDAs based on traffic variables (rather than images or sounds), the best stated performances are achieved on transport simulators using recent machine learning techniques. Transport simulators provide simplified versions of real-world networks, which often do not account for real-world disruptions such as emergency vehicles passing at high speed, erratic driving, or major sporting events. Hence, transport simulators typically output more predictable traffic data values, meaning IDAs can perform better. As such, it is unclear whether results on simulated data are replicable on field data, and if so how much calibration would be required. From studies of those implemented in the field, traffic variable based IDAs appear well suited to motorways and arterials, but find difficulty in accounting for traffic signal noise and contexts within urban streets and junctions.

Image processing IDAs appear feasible for use across various types of network. Promising field results have been stated by many image processing IDAs, but further research is needed to thoroughly evaluate such approaches and directly compare them to IDAs based on traffic variables. Considering the rapid advancement of computer vision in recent years, image processing algorithms appear to have the potential to achieve the best performance when implemented in the field. However, such algorithms will only form part of the solution to incident detection due to the lack of CCTV coverage in many areas. Audio signal processing IDAs may have potential in the future, but more research is first needed to develop and evaluate such IDAs.

Social media based algorithms appear best suited in complementing IDAs on other data sources. The algorithms are able to detect incidents across entire road networks, but they are limited by their detection rate, calibration requirements and time in which it takes incident reports to be posted.

It is clear that limitations still exist which limit the performance of IDAs, particularly on urban networks. Perhaps the most clear and most commonly cited limitation is that many IDAs are unable to differentiate incidents from contexts, resulting in a high false alert rate. Noise from signals on junctions can cause congestion similar to that of an incident, leading to false alerts in traffic variable based IDAs. Finally, many IDAs are only capable of indicating when an incident has taken place in the vicinity of a detector. Traffic operators could respond more effectively if the exact incident location and expected congestion propagation could be estimated. These features are closely related to incident detection, and could be accounted for within the design of IDAs to aid operators further.

The findings of this literature review will be used to shape the rest of the research in this study. Firstly, they will be used to identify an outstanding limitation of IDAs that will be the focus of this research project. Then, they will be used to aid the development of a methodology to address this limitation.

Chapter 3

Incident detection algorithms in practice

3.1 Introduction

The goal of IDAs is to aid operators in achieving the task of incident detection. As such, they must perform well in practice, as well as in theory. However, few algorithms presented in chapter 2 have been evaluated on field data, and even fewer have been implemented in TMCs. As such, research into the state of incident detection and IDAs in TMCs was considered crucial, in order to understand; what is required from an IDA, whether this is currently being met, and to identify any room for improvement. As such, a review of TMC surveys was undertaken. Then, to ascertain whether the findings of these surveys still hold today and in the U.K., interviews were undertaken with key stakeholders involved in the detection and management of incidents in TMCs.

3.2 Review of TMC surveys

As was found in section 2.4, many IDAs have been developed, but few have been implemented in TMCs. Given that IDAs' objective is to aid TMCs as much as possible in achieving the task of incident detection, it is important that IDAs can be implemented easily, and perform well operationally in TMCs. This section reviews surveys of TMCs, with the aim of understanding what incident detection methods are being employed by TMCs, how operators are currently using IDAs in TMCs, and what attributes TMCs require of IDAs.

3.2.1 Description of surveys

Three surveys of TMCs have been carried out, in 1997, 2004 and 2005 (Ritchie and Abdulhai, 1997, Guin, 2004, Parkany and Xie, 2005). The surveys had 7, 32 and 24 responses respectively. The surveys each cover a variety of topics, including how the task of incident detection is handled

in TMCs, and how IDAs fit into this task. Each survey was a written questionnaire sent by email or the internet, and each was conducted in the U.S. As such, the findings of these studies may differ from the current state of incident detection in TMCs in the U.K., particularly as many state of the art IDAs have been developed since 2005 (see section 2.4). Each responder's position in the TMCs varied, and in some cases was unclear. This factor may have influenced each survey's results. For example, traffic operators may prefer IDAs to empower operators in detecting incidents more effectively, whereas TMC managers may prefer IDAs to automate the task of incident detection to save costs.

3.2.2 Method of incident detection

Parkany and Xie (2005) and Guin (2004) found that CCTV monitoring was the most commonly used method of incident detection, closely followed by witness calls, police reports and motorway patrols. Parkany and Xie (2005) also found that IDAs were commonly used. Other methods included the use of call boxes by the roadside, aerial detection (such as helicopters) and notification from other agencies. It should be noted that these methods and networks may have changed since these studies took place. For example, road network sizes and traffic volumes have been increasing (see section 1.4.1), so there may have been a further shift from manual methods to automatic methods.

Guin (2004) found that 70% of respondents considered their current incident detection methods to be insufficient in meeting their current demands, and a further 20% expected they'd be insufficient in meeting future demand. This was thought to be because the size and scope of road networks under TMC monitoring were growing faster than TMC staffing levels and resources. This shows that at the time of the surveys, many TMCs that relied on manual methods for incident detection (such as CCTV monitoring) were unable to cover the increasing road network sizes. As this trend of increasing network size has continued, it is expected that the demand for effective automatic incident detection methods would have increased. The findings indicate the need for research into automatic incident detection methods, that work effectively when implemented across large road networks, using the limited resources available.

Guin (2004) stated that the most predominant factor in determining the speed of incident detection is the method used, and so questioned TMCs on the average time each method took to detect incidents. The lowest average time to detect incidents was found to be by IDA, taking just under four minutes. The average non-IDA incident detection time was found to be 8.5 minutes. Clearly many other factors also influence the speed of detection, such as the size of the network, the design of IDA etc. For example, IDAs would likely detect incidents more quickly in networks with high flows because of the greater disturbance in traffic conditions. As such, these findings may differ greatly between TMCs.

These findings highlight the opportunity for IDAs to aid TMCs in the task of incident detection. IDAs can reduce the average time of detection by being the first method to indicate the presence of an incident, and can allow TMCs to cover large road networks by providing an automated

method of quickly analysing large amounts of traffic data.

A combination of IDA and manual methods in detecting incidents may be most effective in current TMCs. It appears that many TMCs are becoming less able to detect incidents using only manual methods, because road network sizes are increasing and TMC resources are limited (e.g. too many CCTV cameras to be monitored by operators simultaneously). In these cases, operators are required to analyse vast amounts of data continuously. IDAs can analyse this data more quickly, and so could be used to first indicate the presence of an incident by raising alerts. Alerts could then be verified by operators to confirm the presence of an incident (as current IDAs do not achieve 100% detection rate and 0% false alert rate). This would be done manually by analysing data relevant to the alerts (such as checking relevant CCTV cameras). Compared to manual methods, this approach would require operators to analyse significantly less amounts of data.

3.2.3 IDA use

The surveys of Guin (2004) and Parkany and Xie (2005) investigated the use of IDAs in TMCs, the satisfaction of TMCs with implemented IDAs, and the reasons behind this satisfaction.

Of the 32 TMC responses in Guin (2004), 15 had not implemented an IDA, seven had a disabled IDA, two had an IDA which was being ignored, four were partially using an IDA, and only four had a fully operational IDA. Parkany and Xie (2005) found that half of those that had an implemented IDA were satisfied, and half were not. Some TMCs stated preferring developing their own algorithms for incident detection, which could be tuned to fit operators' preferences and local traffic conditions.

Guin (2004) found that "the primary and most commonly cited reason (for preventing IDA usage) was an unacceptably high rate of false alarms". These false alerts caused discomfort and distraction for operators, which usually outweighed the benefit of the algorithms' faster detection (Guin, 2004). Other deterrents included IDAs' difficulty in calibration, and low detection rates. The difficulty in calibration was found to be from IDAs being too complicated and time consuming for implementation in many TMCs. Parkany and Xie (2005) reported that some TMCs had to discard their IDAs due to high false alert rates and long detection times, resorting instead to CCTV monitoring and witness reports. The long detection time of IDAs contrasted with Parkany and Xie (2005), but this may be the result of IDAs being tuned to improve false alert or detection rates e.g. increasing the number of over threshold messages required before alerts are raised.

In a separate study, seven TMCs across the U.S.A were visited in 1993 (Balke, 1993). The purpose of the visits was to understand the role of IDAs in TMCs, and how they are currently being used. Four of seven TMCs were actively using an IDA, all of which were a modified version of the California algorithm (Balke, 1993). The three others used to use a version of the California algorithm, but subsequently discontinued its use due to the high number of false alerts produced. The most reported cause of false alerts was from improper calibration. Many TMCs set the same

threshold values for the entire network, but as traffic conditions vary from location to location, calibration must be undertaken for each detection location. For the most part, those with an IDA were pleased with its performance, but it was not relied upon heavily for the task of incident detection (relying mostly on CCTV and radio reports instead).

Section 3.2.2 highlighted the opportunity IDAs have in the task of incident detection, which has been reflected by the interest shown by TMCs. But the studies above have found a lack of use, and general dissatisfaction with implemented IDAs, principally due to high false alert rates (Balke, 1993, Guin, 2004, Parkany and Xie, 2005). It can be seen that the lack of IDA use is partly from the dissatisfaction felt by TMCs, principally due to high false alert rates. This is a limitation that must be addressed if IDAs are to meet their objective in aiding TMCs in the task of incident detection.

3.2.4 Required performance of IDAs in TMCs

Guin (2004) and Ritchie and Abdulhai (1997) then asked TMCs what performance would be acceptable for an implemented IDA, and which performance metrics should be prioritised. These questions improve the understanding of how to evaluate the performance of IDAs, and why so many TMCs have been dissatisfied with implemented IDAs. Ritchie and Abdulhai (1997) asked operators what detection rate and false alert rate would an IDA have to achieve for it to be considered acceptable for them to use in their TMC. The average of their responses was that the detection rate would have to be at least 88.3%, and the false alert rate at most 1.8%. By the same method, Guin (2004) found an acceptable false alert rate would be for an operator to check at most three false alerts per hour, and 10 per day. IDAs must also be simple and automated enough in terms of calibration to be used in TMCs. To be acceptable, it was found that implementation and ongoing calibration of IDAs must be achievable either automatically or by TMC staff (Guin, 2004).

Abdulhai and Ritchie (1999) used the results of Ritchie and Abdulhai (1997) to define a set of attributes which an IDA would need in order to be 'universal', i.e. to meet all the needs of TMCs in the task of incident detection. It was thought that IDAs should aspire to achieve these attributes in order to meet their objective of aiding TMCs. They include:

- High performance in terms of detection rate, false alert rate and mean time to detect.
- Minimal time, effort and skills required for implementation.
- Transferable performance to different locations
- Minimal training data required
- Able to account for the prior probability of incidents
- Capable of providing an estimate of incident duration and severity, and detection certainty.

None of the IDAs reviewed in section 2.4 possessed all of these attributes. If IDAs cannot sufficiently attain all of the attributes required by TMCs, IDAs must prioritise these attributes. To thoroughly evaluate IDAs, tightly defined performance measures based on attributes desired by TMCs must be created.

It should be noted that some IDA performance measures can be seen as a trade-off. Including the most commonly used metrics, detection rate, false alert rate and the average time to detect. Generally speaking, the detection rate is directly proportional to the false alert rate and average time to detect, and the false alert rate is inversely proportional to the average time to detect.

Many IDAs can be tuned to alter their performance of these metrics, which could be done to benefit the needs of the TMC using it (e.g. a low false alert rate may be preferred over a high detection rate). For example, some IDAs are capable of being tuned to reduce the time to detect, but in doing so would raise the false alert rate, and lower the detection rate (Ghosh and Smith, 2014).

Guin (2004) concluded that because so many methods to detect incidents exist in TMCs, it's unlikely that any incidents will go undetected. Hence, 100% IDA detection rate was seen to be not critical. Instead, the objective of IDAs could be simplified to minimising the average time to detect incidents, while maintaining acceptable rates of detection and false alerts. This would give IDAs the best opportunity to aid operators by being the first method to detect incidents. Other methods could then be used to manually verify the incident afterwards.

3.2.5 Summary

Three surveys of TMCs were reviewed in order to understand what incident detection methods are being employed by TMCs, how operators are currently using IDAs in TMCs, and what attributes TMCs require of IDAs. The surveys took place in the late 1990s and early 2000s, and each was a written questionnaire carried out in the U.S. As such, the findings of this review may not be representative of the current state of incident detection in U.K. TMCs.

The review found that as the size and scope of TMC responsibilities grow, TMCs have a growing need for automated incident detection methods, such as IDAs. Many TMCs have shown an interest in implementing IDAs, but once implemented, many IDAs have failed to satisfy.

As was found by Parkany and Xie (2005), and stated in section 2.4, there appears to be little connection between algorithms that have only been tested offline or in simulators, and those that have been implemented and used in TMCs, both in terms of design and performance. Those that have been implemented in TMCs have tended to be on the less complex end of the IDAs presented in the literature, and performances found in TMCs have typically been worse than in offline tests.

The most commonly used IDAs in TMCs appear to be the simplest. Each TMC that implemented an IDA in the survey of Balke (1993) used a California algorithm. Parkany and Xie (2005) found that operators often preferred developing their own IDA. This indicates what TMCs need from an IDA, i.e. to be simple to implement, and to perform reliably. Many IDAs do not meet these requirements as they perform poorly, and require too much time, effort and knowledge to be implemented and maintained. TMCs' most commonly reported lack of performance was the excessive false alert rate, which distracted operators, leading to them becoming dissatisfied and abandoning their IDAs.

In section 2.4, the results found when IDAs were tested on simulated data were often better than when tested on field data. The surveys report that when IDAs are implemented, the principal cause of dissatisfaction is from high false alert rates, which is often caused by mistaking incidents for contexts. It may be that some of the results found on simulated data were better than they would have been on real world data because the simulator did not reflect all the complexities of real world networks, such as the disruption caused by contexts, and complex vehicle behaviours (such as ambulances passing). Other reasons include the likely worse quality of field data, which may have errors from the detector itself or the process of calculating the variables. For IDAs to achieve their objective, they must perform reliably on field data when implemented.

The surveys indicated that the most effective way for IDAs to aid TMCs in achieving the task of incident detection should be to be the first identifiers of an incident, reducing the overall time taken for incidents to be detected. Once identified, incidents could then be verified by operators using manual detection methods such as CCTV monitoring (as IDAs do not currently achieve 100% detection rate and 0% false alert rate). If this was the case, the objective of IDAs could be simplified to minimising the time to detect incidents, while maintaining acceptable detection and false alert rates, and being sufficiently simple for TMC operators to calibrate and use.

For IDAs to achieve their objective, they must perform well when implemented so that they aid TMCs in the task of incident detection. The surveys suggest that IDAs must require minimal calibration when implemented in new networks, and achieve acceptable performance levels in terms of detection rate, false alert rate and average time to detect. However, these requirements are subjective, and unprioritised. As such, a set of clearly defined key performance indicators will be created to thoroughly evaluate IDAs.

3.3 TMC interviews

In section 3.2, a review of TMC surveys was undertaken to understand what incident detection methods are being employed by TMCs, how operators are currently using IDAs in TMCs, and what attributes TMCs require of IDAs. A number of findings were drawn from this review, which will have an important bearing on the design of the incident detection algorithm being developed in this research project. However, as the surveys took place in the U.S. over 10 years ago, they may not be representative of the current state of TMCs in the U.K. To affirm or dismiss the

findings of the review of TMC surveys, a number of interviews were undertaken.

The interviews were based off a set of questions, stated in appendix A. For all TMCs that have operators responsible for detecting and responding to incidents in real-time, representative stakeholders were attempted to be found. Three interviews were accepted, two of which took place over the phone, and one in person. A fourth participant was unable to take part in the interview, but gave information back by email. The interviews were conducted in December 2017 and January 2018.

The first interviewee was a TMC operator from Cardiff Councils TMC. The TMC in Cardiff consists of a team of operators, two of which are full-time, who are responsible for incident management across the TMC's road network. Another part of their role is the CCTV monitoring and reporting of any kind of 'abuse' across the network, both on and off the road. The network area is Cardiff Council, which covers Cardiff city centre and the surrounding boroughs. All types of roads are managed in this area except for motorways, which are covered by Highways England.

The second interviewee was a Principal UTMC Engineer from Bristol City Councils TMC. They were responsible for managing a team of operators that manage incidents across the network. The network area is Bristol City Council, which covers the city centre and the surrounding boroughs. Again, all types are managed in the area except for motorways.

The final interview was with Siemens' Head of UK Consultancy Services. The interviewee's role is to manage a team of 26 engineers who work with TMC operators to implement and maintain their traffic management software, and in some cases operate the TMC themselves. The interviewee also has prior experience of operating a TMC. As such, the interviewee was able to comment on the incident detection practices on each of Siemens' 32 U.K. tenants, i.e. TMCs that use Siemens' traffic management software.

The fourth response to the request for interview was from a Network Operations Manager from Essex Highways. They were unable to take part in the interview, but they did respond by giving information by email on how Essex's TMC responds to incidents. The TMC is responsible for managing a network that covers the whole of Essex, but again motorways are not covered.

Firstly, there appeared to be a general consensus that the role of an operator was to monitor and react to traffic disruption on a given network. Incident detection is a part of this responsibility. The most common methods of detecting incidents were found to be using CCTV monitoring (all four respondents used this). Other common methods in use by the respondents, or were known by respondents to be commonly used elsewhere included manually monitoring traffic data (such as from Google or Siemens), contact from road users (e.g. via emergency services call), and from communication with other organisations, such as the Police or Highways England. Siemens' Head of UK Consultancy Services stated that the approach to incident detection varied largely based on the size of the TMC (i.e. number of staff). Larger TMCs, i.e. those manned by operators (such as Bristol and Cardiff), are more likely to use CCTV monitoring as they have more staff

dedicated to incident management, and more detectors and cameras. In smaller TMCs however, there is typically less time and resources dedicated to incident detection and management. It is often the responsibility of traffic engineers, who cannot spend much of their time monitoring CCTV for incidents as they have a wide array of other responsibilities, such as working on new transport schemes. As such, fewer incidents are detected, and methods such as getting reports from external sources, such as the Police, are more common.

When asked about how incidents were responded to once detected, a number of common responses were again found. The respondents from Bristol and Cardiffs TMC stated that they would alter traffic signal strategies. Siemens' Head of UK Consultancy Services stated that the approach of responding to incidents was said to again depend on the size of the TMC. Smaller TMCs' traffic engineers typically change a traffic signal plan on their own merit, but wouldn't make any other response. Larger TMCs typically follow a pre-set set of guidelines, which include changing traffic signal strategies, and communicating the information to road users and emergency services.

Stakeholders from Cardiff, Essex and Bristols TMC stated that no IDA was currently being used, and there was no known use of an IDA in the past. Siemens' Head of UK Consultancy Services stated that around half of Siemens' 32 tenants have an IDA, INGRID, within their traffic management software, but only three or four have attempted to use it (see section 2.4.1.5). INGRID is a simple comparative IDA that was developed over 20 years ago. The interviewee did not know of any other IDA being implemented. The interviewee spoke of many TMCs becoming dissatisfied when using INGRID. The main reason for this was the high false alert rate. Operators simply did not have the time to check all of the alerts being produced by INGRID. Because not every alert could be checked, the alerts became a distraction from the operators' other responsibilities, and so they would simply ignore or turn off INGRID's alerts. Another problem was that both TMC operators and Siemens' engineers were unable to re-calibrate the IDA because it required too much expertise of the algorithm's parameters. INGRID was also described as being known to 'miss many incidents', i.e. had a low detection rate. Finally, TMCs disliked that the alerts said 'possible incident occurred', rather than giving a confidence level (or having 100% detection rate).

Stakeholders from Cardiff and Bristols TMC stated that the best role an IDA could play in their TMC would be to be the first indicator of an incident, which could then be verified by an operator checking CCTV. This was stated to be because it would speed up the TMC's incident detection times, and allow incidents to be detected over a larger network. The respondent from Cardiff went on to say that if the IDA could be implemented on a currently lesser used data source, in their case loop detector data, then the IDA could benefit by providing additional information to the TMC. Siemens' Head of UK Consultancy Services believed that an IDA implemented in a TMC should play a different role depending on the size of the TMC. For large TMCs, an IDA should be the first indicator of an incident, which could then be verified by operators monitoring CCTV. For smaller TMCs that use traffic engineers, an IDA should only alert them when it is '99%' sure that an incident has occurred, because engineers often do not have the time or means to verify many false alerts.

Interviewees were then asked about what the most important features an IDA should have to be implemented in a TMC. The respondent from Cardiff stated that the TMC prioritises detecting major incidents and monitoring incident hot spots, but it was emphasised that an IDA would not need to make the same priorities. This is because incident hot spots are already covered sufficiently by existing methods, and major incidents are known of quickly either by CCTV monitoring or reports from the police and road users. Siemens' Head of UK Consultancy Services expressed the idea that an IDA based on the disruption of traffic conditions on loop detector data could raise an alert indicating the possibility of an incident, which would then cause a CCTV camera to automatically change its view onto the location of the loop detector, and then an image-processing IDA could raise an alert if it believed an incident had occurred. This combination of approaches could improve on a single IDA because each IDA could verify each other's alerts. It could also improve the coverage area of a single image-processing IDA because CCTV cameras can only see one view at a time.

When interviewees were asked to give a performance level that would be suitable for use in their TMCs, many found it hard to think of suitable performance metrics and give exact figures. When prompted with examples of commonly used performance metrics, it appeared that they were difficult for practitioners to translate into the experience that they could expect in TMCs. The representative from Bristols TMC was the only to respond with exact figures, which were that the IDA would need to have a mean time to detect between five and 15 minutes, and a detection rate of at least 85%.

During the interviews, both the Cardiff and Bristol respondents mentioned that the TMC would welcome an IDA if it could achieve the desired performance levels, and hence aid them to detect incidents more effectively.

3.3.1 Summary

The interviews undertaken proved very insightful in understanding the current state of incident detection practices in TMCs in the U.K., and hence could be compared to the findings of the review of surveys undertaken in section 3.2.

The interviews indicated that many TMCs in the U.K. have not implemented an IDA, and even fewer are currently using one. The only IDA used in Siemens' TMCs was INGRID (see section 2.4.1.5). INGRID was found to be ineffective principally because of its high false alert rate, which distracted operators and ultimately led to them stopping using the IDA.

The most common incident detection method was by operators monitoring CCTV, and in smaller TMCs was to receive reports from external sources. Other methods included manually monitoring graphs of detector data, and using a stopped-vehicle detector.

The most common responses were to alter the traffic signals strategy and to disseminate the information.

The interviews were undertaken to affirm or dismiss the findings of the review of surveys undertaken previously. The findings of the interviews did not dismiss any of the findings of the review, but did affirm some. The following findings were consistent between the review of surveys and the interviews undertaken:

- The most commonly implemented IDAs have been the simplest.
- TMCs have been dissatisfied with implemented IDAs.
- The main cause of this dissatisfaction has been the high false alert rate, which distracts operators.
- Other causes of dissatisfaction include the low detection rate, the expertise required for calibration, and the lack of a measure of an alert's confidence.
- For TMCs manned by operators, there was consensus that the best role an IDA could play in aiding the task of incident detection in TMCs is to be the first indicator of an incident, which could then be verified by more reliable means, such as an operator checking CCTV. This would speed up TMCs' incident detection times.

3.4 Conclusions

This chapter reviewed previous TMC surveys to understand current practices in incident detection and the role played by IDAs, and described interviews with key stakeholders to affirm or deny the findings of these reviews in the U.K. presently. Many findings of the interviews conducted were found to be consistent with the findings of the surveys reviewed. It can be concluded that IDAs are not currently fulfilling their potential in aiding TMCs to detect incidents. IDAs have many areas for improvement in practice, which if achieved could result in wider adoption in TMCs, and hence benefits in achieving the task of incident detection.

This chapter, along with chapter 2, has addressed this research project's first objective of providing an up to date review of IDAs, and highlighted limitations with the current state of the art. The findings of this chapter will be used to shape the methodology of the IDA to be developed, in particular with regards to its ease of implementation, maintenance and expected performance in TMCs.

Chapter 4

Approach justification

4.1 Introduction

Objective two of this research project involves developing an IDA that is able to differentiate incidents from contexts. This chapter first reviews literature related to contextual data in order to understand how an IDA could address objective two. A hypothesis is then described on how contextual data could be used in the proposed IDA. This hypothesis is then used to develop the high-level approach that will be taken by this research project's IDA. Finally, the originality and potential benefits of the identified approach are described.

4.2 Context related literature

Given the problem of this research project (stated in section 1.6), it is necessary to investigate studies that have incorporated external data sources to incident detection and contextual data related studies. This literature will help to understand how best to tackle the problem of differentiating incidents from contexts. In particular, this literature will be studied to understand how contexts affect traffic conditions, to what extent, and how this can be accounted for. Traffic variation, incident occurrence and IDA studies will be reviewed. Unlike the review in section 2.4, which focused on the major differences in IDA methodology, this review focuses on how external data sources have been incorporated within algorithms and analyses of traffic conditions and incident occurrences.

4.2.1 Traffic variation studies

Many papers found that spatial and temporal variation in traffic conditions can be caused by contexts, such as public holidays, major events, and adverse weather conditions (Stathopoulos and Karlaftis, 2001, Thomas et al., 2008). Trends in variation in traffic conditions were found as early as the 1950s (Wardrop, 1952*b*).

At a minute-to-minute scale, the largest cause of variation in urban areas was found to be from traffic signals (Weijermars, 2007). At this time scale, a number of other factors can be seen as noise for forecasting and incident detection applications. Noise could come from inaccurate detectors, or individual vehicle's unpredictable behaviour (e.g. a vehicle speeding). For example, Chin et al. (2004) estimated that delays from delivery vehicles parking on the roadside caused just under one million vehicle hours of delay in 1999. Variation from individual vehicle behaviour is particularly noticeable on detectors with low flows.

Daily variation in traffic conditions appears to be most determined by commuters' behaviour. Weijermars (2007) found that flows in Almelo, Netherlands had an AM and PM peak on weekdays coinciding with the typical 8am-4pm work day, but a bell-shaped profile at weekends. On working days, an average of 7.5% and 8.7% of the day's traffic volume occurred in hour long periods in the AM and PM peak respectively. On loop detectors in Athens, Greece, Stathopoulos and Karlaftis (2001) found that on average throughout the year, flows were higher in the morning toward the city centre, but higher in the afternoon away from the centre.

Major events and holidays have also been found to disrupt traffic conditions. Weijermars (2007) found that in Amsterdam, Netherlands, major events such as national and international football matches and concerts resulted in higher traffic flows on nearby roads. In Almelo, Netherlands, holidays showed lower than average traffic flows, particularly during the morning when rush hour would otherwise cause a sharp increase in flow. Song and Miller (2012) found much lower flows than normal (i.e. average weekdays) on Thanksgiving day, and the the morning after. However, a small peak period at noon on Thanksgiving day was observed, which was reported to be caused by people visiting families and friends.

Roadworks can also disrupt traffic conditions. Weijermars (2007) found that in Almelo, Netherlands, sometimes roadworks would cause a great deal of disruption, but other times caused only a slight difference, e.g. at detectors on roads further from the roadworks or on a diversion route. Long term roadworks could easily be mistaken for seasonal effects, particularly when they had noticeable, but little disruption. Almelo is a city of 70,000, and has many road network features, such as a ring road and signalised intersections, that are typical of cities in the U.K. It is therefore thought that the findings above could be transferable to urban locations in the U.K.

The impact of weather on traffic conditions has been extensively covered in the literature (Andersen and Torp, 2016). The majority of data driven studies found that urban flows reduce during adverse weather conditions such as rainfall (Al Hassan and Barker, 1999, Changnon, 1996, Goodwin, 2002). Adverse weather is also thought to reduce road capacity, increasing the likelihood of congestion (Zhang et al., 2015, Kamga and Yazici, 2014, Lam et al., 2013). However, forecasting studies have found that weather information contributed little to forecast accuracy and could even lead to over-fitting (Bajwa and Kuwahara, 2003, Zhou et al., 2014).

Longer term and seasonal factors have also been found to affect travel demand, and hence traffic flows. Li et al. (2004) found correlations between seasonal traffic patterns and contextual factors,

including the ratio of seasonal households to permanent households, hotel visitors, ratio of retail employment to population and the percentage of retired households with high incomes were important factors. In Athens, Greece, Stathopoulos and Karlaftis (2001) found that average flows did not change significantly between 1997 and 1999. Seasonal variation were also not strong, but July and August had significantly lower flows because of residents taking vacation. In the Twin Cities metropolitan area of the U.S.A., Kim et al. (2008) found that average traffic flows increased from 1997 to 2006, but the rate of increase decreased. The highest flows were seen in August, and lowest in January, reported to be caused by the change in weather. Flows on the I-94 were found to decrease as a result of a light rail service opening in 2004.

From this review it can be seen that many types of contexts cause variation in traffic conditions. As such, an understanding of conditions under these contexts would be required to address this issue. It can also be seen that many of these contexts occur irregularly, but are scheduled, or are predicted to occur beforehand. As such, to account for variation from contexts, contextual data (such as the contexts' schedules) would need to be incorporated. Unfortunately, few of the variation studies reviewed have built on their research to create forecasting algorithms.

4.2.2 Incident occurrence studies

In a similar way to the traffic variation studies reviewed, many have investigated the contextual factors affecting incident occurrences. These contexts differ from the contexts of section 4.2.1 because they describe the context of incidents, that is, factors that are known of in advance that could be expected to influence the likelihood of an incident occurring. To find the patterns of these contexts, some studies have matched historical records of incidents with contextual datasets, such as weather, daylight, geometry, and social datasets such as Twitter.

Some studies have studied the link between incident occurrence and the interactions on social media, such as Twitter. Mai and Hranac (2013) found that in California, tweets are posted on average 5 hours after an incident occurring, and between 10 and 25 miles of the incident's location. This indicates that social media platforms are unlikely to be a useful data source for detecting incidents.

Other studies have found that the frequency of traffic accidents is correlated with weather conditions. Yuan et al. (2014) found that the highest number of traffic accidents in Hong Kong occur in the first hour of rainfall and the first hour after rain (however, Hong Kong gets substantial rainfall that may affect incidents more so than other locations). Nofal and Saeed (1997) studied accidents in Riyadh, Saudi Arabia between 1989 and 1993. Accidents were directly correlated with the temperature, and inversely correlated with adverse weather conditions such as rain, snow and hail. Accidents were most common in the summer season, particularly between noon and 3pm, which was explained by the heavy traffic, intense sunlight and high temperatures faced. Ahmed et al. (2012) found distinct differences in the patterns of incidents during the snow and dry season on the I-70 highway in Colorado, U.S.A. During the snow season, the likelihood of

incidents doubled, particularly on steep sections of road. Precipitation, visibility, and the variation of vehicle speeds before incidents were also found to be significant factors. This contrast between studies in Hong Kong, Saudi Arabia, and the U.S.A. shows that the same context, in this case weather, can affect the likelihood of incidents differently in different road networks.

Clearly then, the likelihood of incidents occurring is affected by contexts. Data on these contexts can be collected prior to incidents occurring, and incorporated within IDAs. The benefit of incorporating this type of contexts is that they could be tuned to be more sensitive at times/locations where contexts indicate an increased likelihood of incident, such as rain or tight corners.

4.2.3 Incident related data included in incident detection algorithms

Incorporating external data (i.e. data that does not describe road traffic conditions) within incident detection algorithms is not novel. However, previous studies have incorporated such data in an attempt to better understand the likelihood of an incident occurring (i.e. a prior probability), rather than understanding traffic conditions that can be expected to occur. Based on this likelihood, the IDA's sensitivity to raising alerts would be automatically adjusted, increasing performance. Such data could include the weather, road geometry and speed limits.

An example of this was demonstrated by (Lam et al., 2016). The authors first studied data from video traffic detectors in Hong Kong, and found that as the intensity of rainfall increases, the estimated capacity (vehicles/hour/lane) and free-flow speed (km/h) falls, across all road types. An IDA was then developed that incorporated recent traffic observations, speed limits, and rain data. It was argued that such external factors affect the performance of IDAs, and so different thresholds would be used during different contexts. The IDA itself was comparative, and was an extension of the standard normal deviate IDA (reviewed in section 2.4.1). When tested on the video detectors in Hong Kong, the performance of the IDA improved when rainfall context was included, particularly by reducing the false alert rate. The IDA (rainfall data included) achieved a detection rate of 91.9%, false alert rate of 4.0% and mean time to detect of 4.2 minutes. There were however a number of limitations apparent in the approach of this IDA. Firstly, the IDA aimed to understand what behaviour could be expected at each detector under incident, non-incident, rainfall and non-rainfall scenarios. Because of the lack of messages where an incident occurred, and when rainfall occurred, the understanding that could be gained for certain scenarios was limited. It could be expected that if more external data sources were incorporated, this approach would require very large amounts of historical data to be calibrated effectively. Also, their calibration process required road speed limits, and some manual calibration to set detection thresholds under different scenarios. Ideally, external data would be incorporated in an automated way, so that the cost, time and knowledge required for implementation wouldn't increase.

4.2.4 Summary

External data (including contextual data) have been used in a number of studies because they have been found to cause variation in traffic conditions and incident occurrences. Traffic variation and incident occurrence studies highlight the wide range of contexts, and their influence on different road networks. The contexts that most effect traffic conditions either can be predicted or occur irregularly but to a schedule. As such, the IDA developed in this research project would have to incorporate data on these contexts' schedules or predicted times to account for the variation caused. As some contexts have similar effects on the traffic as incidents, an IDA would not be able to differentiate the two without an indication that a certain context was occurring.

Although some IDAs have incorporated external data, few have done so in an attempt to better understand what traffic conditions could be expected when no incidents occur. Instead, most IDAs attempted to learn the characteristics of incidents, and have used external data to improve the understanding of the likelihood of an incident. This method requires large amounts of incident data, which is typically either done by collecting large amounts of historical data, or simulating incidents which requires calibration in modelling the road network. Ideally, IDAs should be able to incorporate external data without requiring large amounts of training data, and to do so in an automated way, so that the only calibration required would be to collect and input the traffic and context data.

4.3 Hypothesis

As was stated in section 1.6, the problem that will be the focus of this research project is the problem of IDAs differentiating incidents from contexts. Addressing this problem will address objective two of this research project.

It has been found that the conditions displayed during contexts and during incidents are not exclusive (Cook and Cleveland, 1974). Also, contexts are scheduled in advance, or can be predicted to be disruptive, but often occur at irregular times. As such, it is thought that the most effective way for an IDA to account for contexts is to incorporate data on their schedules. Such an IDA would be able to infer from contextual data that a context's disruption could be expected at a particular time, and so would be more capable of differentiating this disruption from incidents. As such, the research hypothesis can be stated as follows:

IDAs can better differentiate disruption from contexts and incidents if contextual data is used to better understand traffic conditions that can be expected to occur.

It is thought that the incorporation of contexts within IDAs will lead to a reduction in the number of false alerts from contexts. For example, a football match may lead to congestion, or cause a sudden change in traffic conditions, that could be falsely detected as an incident. By understanding the expected amount of disruption each context causes, such false alerts wouldn't

be raised. However, incidents during contexts would remain detectable, such as lower than expected flows on roads downstream of an incident during rush hour. During low flows, many incidents do not cause sufficient disruption to be detectable by IDAs that rely solely on traffic data (Cherrett et al., 2002). With sufficient understanding of the context at the time, incidents may become more recognisable in this case, increasing the detection rate. For example, context incorporated IDAs may detect incidents better when no contexts are affecting the traffic, and an incident causes lower flows downstream, or higher flows on a nearby road.

This hypothesis will be tested by developing an IDA that incorporates contextual data. It will be designed to use contextual data in a way that will be as effective at detecting incidents as possible, while remaining simple and easy enough for TMCs to implement, use and maintain. The IDA will be evaluated with and without contextual data, and compared against other state of the art IDAs.

4.4 Approaches considered

There are a number of ways that contextual data could be incorporated within an IDA to detect incidents. As such, each type of IDA (reviewed in chapter 2.4) was considered to decide what type of approach would be taken in this research project. The following sections consider each type of IDA, and explains why each type would or wouldn't be suitable for the proposed approach.

4.4.1 Comparative

Comparative IDAs effectively work by defining what 'expected' traffic conditions are, then raise alerts when conditions fall outside this definition. However, comparative algorithms use a fixed threshold at each detection location to define the 'expected' traffic conditions, and so do not account for temporal or context caused variation, and cannot account for spatial variation without significant amounts of calibration. Time-series IDAs differ from comparative IDAs in that their definition of 'expected' conditions varies based on recent observations, and so can better account for these variations. As such, the comparative algorithm was not seen as suitable as it could not account for spatial and temporal variation, including variation from contexts.

4.4.2 Time-series

Time-series IDAs use a forecast of 'expected' traffic conditions, and raise alerts when this differs from real-time observations sufficiently. However, time-series IDAs base their forecast on models of traffic evolution, relying on recent observations. These recent observations may be during incidents, and so forecasts could be influenced by these incidents' disruption. This could lead to the IDA missing incidents because forecasts would be more similar to real-time conditions during an incident. As such, it was decided that the IDA developed would not use recent observations, in order to ensure that only 'expected' traffic conditions would be forecasted. However, this IDA type's approach of forecasting 'expected' traffic conditions appeared somewhat suitable,

because the approach demonstrated that a forecasting algorithm could benefit from incorporating contexts.

4.4.3 Machine learning

The approach of machine learning IDAs described in chapter 2.4 is to 'learn' the conditions of incidents and non-incident periods. This approach appeared limited because the fortunate rarity of incidents means that large amounts of data would need to be obtained for the algorithm to 'learn' the conditions of an incident. In the literature, the common approach to avoid this limitation was to train the algorithm on a transport simulator with simulated incidents. This means that when implemented, the given road network would first be simulated, then the algorithm would train on the simulated network's data (with simulated incidents), then the algorithm would be implemented on the real road network's data. If this approach were to be taken, large simulated models would need to be created, and significant manual calibration would be required to incorporate contexts within the models. As such, this approach was not taken. However, it appeared that machine learning algorithms were capable of incorporating external data sources (such as traffic signal strategies), and so may be suitable for incorporating contexts. Machine learning algorithms could also be seen as powerful tools for analysing large amounts of historical data effectively for incident detection.

4.4.4 Image and audio signal processing

Finally, the approaches of image and audio signal processing IDAs were deemed unsuitable for a number of reasons. Firstly, this type of IDA can only be implemented in locations that have sufficient CCTV or audio detector coverage. The coverage of detectors using numerical metrics of traffic conditions (e.g. loops and ANPR using flow and average speed) is far greater than CCTV and audio-detector coverage in the U.K. (see section 3.3), and so an IDA based on numerical metrics was seen as more suitable for this research project. There was also a lack of availability of image and audio data, meaning such a type of IDA would be more difficult to develop. As research into image and audio processing IDAs is less mature than numerical metric based IDAs, the performance and limitations of such IDAs could not be fully understood from the literature review undertaken (see section 2.2). It is also unclear whether such IDAs suffer from the problem of differentiating between incidents and contexts, and so may not be relevant for this study's hypothesis.

4.4.5 Summary

After considering each type of IDA, none appeared entirely suitable for the proposed hypothesis. However, elements of the time-series and machine learning IDAs showed potential to be suitable. Machine learning algorithms appeared capable of incorporating contextual data in an effective way, without requiring significant calibration, because the data could be automatically 'learnt' from. The most suitable approach appeared to be the time-series method of forecasting 'expected' traffic conditions, and then raising alerts when real-time conditions differed. The main downfall found in the literature for this type of IDA was that the forecasts were not accurate

enough during contexts' disruption, resulting in false alerts. If the forecast of 'expected' conditions could be improved upon with the incorporation of contextual data, the performance of this approach could be shown to improve.

Based on this evaluation of previously presented IDAs, an idea of the approach of the IDA in this research project was formed. Firstly, an algorithm will create a forecast of 'expected' traffic conditions. Contextual data will be incorporated within this algorithm as input features. Then, the IDA will raise alerts in the same manner as previous time-series IDAs, that is by comparing the forecast to real-time traffic conditions, and raising an alert when these differ sufficiently.

The key to this IDA will be the accuracy of the forecast of 'expected' traffic conditions, particularly during times at which contexts disrupt traffic conditions. To assess the feasibility of this idea, relevant traffic forecasting literature will be studied in the next section.

4.5 Traffic forecasting literature

One of the key components of the proposed IDA idea is the traffic forecasting algorithm. This algorithm will need to accurately forecast conditions that could be expected if no incident occurred, particularly during contexts' disruption, so that incidents could be more effectively differentiated. Therefore, a review of literature was considered necessary to understand how such forecasts have been developed in the past.

The field of traffic forecasting is large, and there have been many different approaches to the problem. Time-series (Van Der Voort et al., 1996, Williams and Hoel, 2003), statistical (Crawford et al., 2017), Bayesian (Zhou et al., 2014), neural network (Goves et al., 2016), random forest (Leshem and Ritov, 2007, Zarei et al., 2013) and support vector machine (Vanajakshi and Rilett, 2004) methods are some of the most common approaches. However, a large portion of traffic forecasting algorithms are not suitable for the proposed IDA idea. The IDA needs to forecast using target variables relevant for incident detection, such as flow speed aggregated over 30 seconds, and it needs to forecast conditions that could be expected if no incident were to occur. Finally, it needs to forecast contexts' disruption, so that the IDA could differentiate this from incidents. It should be noted that these requirements are described in more detail in section 5.2.

The focus of this review is to highlight algorithms that are suitable for the proposed incident detection algorithm idea. As such, the following section explains why some approaches are unsuitable. The subsequent sections then focus on techniques used in forecasting algorithms that are suitable.

4.5.1 Unsuitable forecasting techniques

Many forecasting algorithms are based on inference from recent observations of traffic conditions (Vlahogianni et al., 2004, Leshem and Ritov, 2007, Zarei et al., 2013). When incidents disrupt

traffic conditions, these forecasting algorithms would take these conditions as recent observations, and use them to forecast conditions that were disrupted by incidents in the near future. That is, the algorithm would not be forecasting 'expected' traffic conditions, but incident influenced conditions. For IDAs that aim to detect incidents by looking for differences between real-time conditions and forecasted conditions, this would lead to incidents being missed. If recent observations were to be used, the horizon used would need to be longer than the duration of the incident, in order to assure that the forecasts were not influenced by the disruption caused by incidents.

Another issue with using algorithms based on recent observations for incident detection is that they are less effective at the edge of disruption caused by contexts. This is because they have no way of knowing that a sudden change is about to occur. Pan et al. (2012) demonstrated this effect in average speed data. They then developed a forecasting algorithm that would use ARIMA the majority of the time, but would switch to using a historical average when sudden changes were expected to occur. When tested on 5-minute loop detector average speeds from motorways and arterial streets in LA County, U.S.A, the combined historical average and ARIMA model was found to forecast up to 78% (mean absolute percentage error) more accurately than each of the individual models at a horizon of 30 minutes.

For the reasons described above, it was decided that the developed forecasting algorithm will base its forecasts on historical patterns of traffic conditions, rather than recent observations.

Another area of the traffic forecasting field focuses on forecasting aggregated travel behaviour (such as yearly vehicle miles travelled) many years into the future. This area of research is unsuitable because the target variables being forecasted are sampled over too large time periods, such as monthly or yearly vehicle miles travelled. Real-time incident detection algorithms use data sampled more frequently, such as a message every 30 seconds. Only data sampled frequently would allow an IDA to achieve a reasonably low mean time to detect. As was stated in section 3.2.2, a survey of TMC operators found that the average time to detect of IDAs was reported to be four minutes (Abdulhai and Ritchie, 1999).

When forecasting at a horizon of at least an hour, historical averages based on time of day and day of week information are typically used (Chrobok et al., 2000, Syrjarinne, 2016). These approaches have been incorporated within IDAs previously (Bowers et al., 1996, Westerman et al., 1996), but many perform poorly when implemented, in part due to high false alert rates. Many generate false alerts during disruption caused by contexts that appears similar in nature to incidents. This occurs because the historical average predictor cannot forecast contexts accurately. As such, attempts to improve forecasts of traffic conditions during contexts are also reviewed.

4.5.2 Potentially suitable forecasting techniques

This section reviews forecasting algorithms that develop methods presented in the literature that showed potential to be suitable for the proposed IDA idea. In particular, many studies

have shown that forecasting algorithms' performance can be improved with the incorporation of contextual data. The algorithms presented were reviewed in detail in order to understand whether any would be suitable for the proposed IDA idea, or whether parts of the method could be used.

Firstly, many traffic forecasting algorithms have included contexts as an input in order to improve accuracy. The most simple way to do this is to attempt to isolate the disruption caused by a context, and to had this disruption to future occurrences of the context. Thomas and van Berkum (2009) separated the expected flow at the times of event contexts into two parts, the demand generated by those visiting the event, and the background demand. The background demand was estimated using a historical average of days in which the event did not occur. The context demand was estimated by taking the previous 10 days of event contexts, and finding the difference between the average of these values and the historical average values. The limitation of this approach is that the 'learning' of how contexts affect traffic conditions is done in a very manual manner. Influential contexts need to be identified for each detector being forecasted, and the algorithm may be unable to determine the conditions expected when multiple contexts occur at one time (e.g. a football match during Christmas holiday) or when contexts occur at different times (e.g. football matches starting at different times on different occasions). Pan et al. (2012) furthered their ARIMA model by incorporating event data by calculating the average difference in speed during event occurrence compared to a historical average. For each event, the type, start-time, location, direction, and affected lanes were recorded. For each combination of these event attributes, the average difference in speed from the historical average was recorded. The forecast was then the historical average minus the average difference. When forecasting at a horizon of 30 minutes, this model was found to improve on the historical average's prediction accuracy (mean absolute percentage error) by up to 91%.

Other forecasting algorithms have incorporated weather contexts in order to improve accuracy. Jia et al. (2017) incorporated rainfall data within an ARIMA model and three types of neural network, deep belief (DB), back-propagation (BP) and long short-term memory (LSTM). The target variable being forecasted were flows from detectors on a motorway in Beijing, China. The predictors used recent observations of rainfall intensity and traffic flows to forecast future flows at a horizon of 10 minutes and 30 minutes. Each neural network performed better when rainfall data was incorporated, but the ARIMA model performed worse. The LSTM neural network performed best, followed by the DB network, followed by the BP network. As these algorithms relied on recent observations, their prediction horizons were at most 30 minutes.

There are also examples of traffic forecasting algorithms that incorporate multiple types of context. Zhang et al. (2015) developed an online optimisation algorithm to identify the influence of both public holiday and weather conditions. This algorithm was developed into a traffic forecasting algorithm which was found to be more accurate than a Support Vector Regression algorithm in terms of mean absolute error. Zhang et al. (2015) claimed that this was the first time average speed had been predicted using inputs of short, and long term historical data as well as contexts. However, this algorithm relied in part on recent traffic observations for accurate traffic forecasts, and so had a horizon of only 10 hours. Another example presented a

hierarchical Bayesian network that forecasted average travel times using inputs of time, day of week, holidays, event and weather data (Zhou et al., 2014). Data was collected from four RFID detectors on a rural motorway in Western Massachusetts, U.S.A., from 1st January 2010 to 30th June 2013. The target variable being forecasted was a measure of congestion, based on the travel time between pairs of detectors. The last 150 days of data was used for testing, while the rest was used for training the algorithm. When forecasting at a horizon of one hour, the algorithm's accuracy was found to improve when incorporating the contextual data. However, the weather data was found to contribute little, and even 'jeopardises some nodes with light traffic'.

Finally, attempts have been made to forecast the disruption caused by planned roadworks, by incorporating contextual data related to the roadworks as input. Hou et al. (2015) developed such a method by using input features such as the roadwork type, duration, length (spatially), start and end time, the number of open and closed lanes, and the roadwork area speed limit. Roadworks differ from many of the previous studies' features in that roadworks principally disrupt the capacity of roads, rather than the travel demand. The most important roadworks features were found to be the roadwork area speed limit and the length of the roadwork area. A collection of predictors were implemented for both short-term (up to one hour) and long-term (24 hour) horizons, including a multi-layer feed-forward neural network, non-parametric regression, regression tree and a random forest. A random forest was found to be most accurate in both short-term and long-term cases, and was found to improve by using the roadwork contextual features. This shows that, with sufficient data on planned roadworks, it may be possible to forecast the disruption they will cause in the future.

Another approach to incorporating contexts within a traffic forecasting algorithms is to form different groups of time periods using clustering techniques, and to train and forecast on each group separately. Chrobok et al. (2004) and Yasdi (1999) both manually analysed historical traffic data in order to split different types of day into groups. Chrobok et al. (2004) split days into four groups; Sundays and public holidays, Saturdays (except holidays), Fridays and days before holidays (except holidays), and Mondays-Thursdays (except holidays or days before holidays). Yasdi (1999) split days into 'ordinary' Mondays, Tuesdays/Wednesdays/Thursdays, Fridays, Saturdays, Sundays/holidays, and 'special event' days. For each presented traffic forecasting algorithm, the appropriate group for the day being forecasted would be selected, and a forecasting algorithm trained only on such days in the training data would be implemented. For Chrobok et al. (2004), this algorithm was a historical average, and for Yasdi (1999) this was a Jordan neural network. Unfortunately, Yasdi (1999) did not test their algorithm on a horizon of over one week. When Chrobok et al. (2004)'s algorithm was tested using two years of flow data from 350 loop detectors in Duisburg, Germany, the historical average was found to outperform a short-term forecasting algorithm at horizons of over two hours, but performed worse at shorter horizons.

More complex, automated methods of clustering traffic data have also been presented (Chung, 2003). The Small Large Ratio (SLR) clustering algorithm was proposed to cluster similar AM peak and PM peak travel times between ultrasonic detectors on a motorway in Tokyo. Once clustered, rainfall, holiday and day of week data were used to explain the clusters, and create

groups of AM and PM peaks based on these contexts. For example, the clusters revealed that AM peaks could be grouped into a weekday group, Saturday group, and Sunday and Monday holiday group. When using these groups as a base for a historical average predictor, this predictor was found to be more accurate with contexts by 0.1% mean absolute percentage error. However, this was an unsupervised machine learning algorithm which clustered data based only on the traffic data. The contextual data was only used to explain the outputs of this algorithm. As such, this method did not automatically 'learn' the patterns of travel times with respect to contexts, but instead relied on human intuition to interpret the outputs of the algorithm, meaning significant manual calibration would be required for implementation. In general, the clustering method was deemed unsuitable for the application of incident detection because of the manual calibration needed in many of the presented methods to identify clusters. Another issue is that when time periods have been clustered, no insights can be gained between clusters. For example if one detector is affected by different combinations of contexts than another, pre-set clusters will be unsuitable for at least one of the detectors. However, the presented clustering methods do show the potential benefit of incorporating contexts into a traffic forecasting algorithm.

4.5.3 Summary

No existing traffic forecasting algorithms were found to be appropriate for the proposed incident detection algorithm idea. Each algorithm was either based on recent traffic observations, did not incorporate contexts, or required a large amount of manual time, expertise and effort to implement. As such, a novel forecasting algorithm will need to be developed for the purposes of this research project. This algorithm will need to forecast 'expected' traffic conditions accurately, and do so by incorporating contexts in an automated way. Although no algorithms reviewed were suitable for the proposed idea, there were elements of these algorithms that were suitable. It was shown that 'expected' traffic conditions can be forecasted (i.e. conditions that occur without an incident), and short-term forecasts can be improved with the incorporation of contextual data. This suggested that the development of an accurate context based forecast of 'expected' traffic conditions would be feasible.

4.6 Approach proposition

The literature review above indicates the possibility for an IDA to differentiate incidents from contexts by taking advantage of an accurate traffic forecast that took contexts into account. This forecast is of 'expected' traffic conditions, i.e. conditions that would be expected to occur if no incident occurred. Then, alerts are raised if real-time traffic conditions differed sufficiently from the forecast.

To evaluate this approach, a traffic forecasting algorithm will be developed in this research project. For simplicity, this traffic forecasting algorithm will be named RoadCast. The goal of RoadCast is to create an accurate forecast of 'expected' traffic conditions, i.e. traffic conditions when incidents are not occurring. This is particularly important during contexts' disruption,

which would otherwise lead to false alerts in IDAs. RoadCast will attempt to account for contexts' disruption by using contextual data as input. This could include schedules of sporting events, weather forecasts, future public holidays etc. Another key element of RoadCast is its requirements to set up and maintain, including the expertise, manual effort and time required. If RoadCast can predict traffic more accurately with the incorporation of contextual data, without needing requirements that would make it unsuitable, it will be considered suitable for the proposed IDA.

RoadCast's forecasts will form the basis of an IDA, named RoadCast Incident Detection (RCID). RCID will first use RoadCast to produce forecasts of future traffic conditions. It will then utilise comparisons between forecasts and traffic data in real-time, in order to raise incident alerts. Unlike many state of the art IDAs, this approach will attempt to learn what can be expected in a non-incident scenario, rather than both an incident and non-incident scenario. As was found in the literature review in chapter 2, this approach has been found to be effective (Guin, 2004), and holds an advantage in that incident data would not be required for training, making implementation more feasible in real-world networks.

It is hoped that by incorporating contextual information, RCID will have greater insight into what traffic conditions can be expected at a particular time (during non-incident conditions). RCID will still have no insight into when incidents would occur, or how they affect traffic conditions, but will assume that they cause a deviation from the expected traffic conditions. Hence, RCID is based on the premise that contexts' influence on traffic is predictable, but incidents influence is unpredictable, and causes sufficient disruption to be detectable.

RCID's ability to detect incidents will depend on RoadCast's accuracy in forecasting 'expected' traffic conditions. Indeed, the closer the predictions to real-time data during non-incident conditions, the easier it will be to detect the disruption caused by incidents.

Another key requirement of this IDA is that it will require minimal calibration, such that TMCs can easily implement and maintain it (see section 3.2). To achieve this, careful consideration will be given to RoadCast and RCID's implementation process, in particular to the incorporation of contexts.

The proposed approach is summarised in the flow chart in figure 4.1.

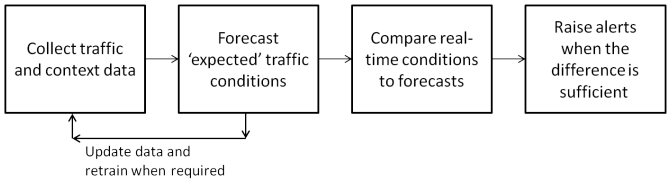


FIGURE 4.1: Flow chart showing the basic approach of the proposed incident detection algorithm.

4.7 Approach originality

The focus of this research project is to address the problem of IDAs failing to differentiate incidents from contexts caused disruption. The novelty of the proposed approach is in its use of contextual data. Rather than using contextual data to improve the understanding of the prior probability of an incident occurring, this approach uses the data to better understand what traffic conditions can be 'expected' to occur. This understanding makes it easier for an IDA to differentiate between disruption from contexts and incidents. The following paragraphs compare the approach to the most related research in the literature.

Contextual data has been used extensively to understand patterns in historical traffic and incident data, as was reviewed in section 4.2. These studies clearly show the effect contexts have on traffic behaviour and the regularity of incidents. A few of these studies have been extended to forecasting future traffic conditions. Similarly, a few traffic forecasting algorithms have incorporated contextual data in order to improve forecasts, as were reviewed in section 4.5.2. Of those that have, few have done so in a way that is applicable to incident detection, and many require significant calibration. The proposed approach differs from previous literature on contexts in that it uses an algorithm that incorporates contexts to forecast conditions that can be expected to occur, and detects incidents based on the difference of this forecast to real-time conditions.

Previously presented IDAs in the literature have incorporated data from external sources (i.e. i.e. data that does not describe road traffic conditions) in an attempt to better understand the likelihood of an incident occurring (i.e. the prior probability), and would change the sensitivity of the IDA accordingly. Although these IDAs demonstrate the improvement that can be made by incorporating contexts, they do not address the issue of differentiating incidents from contexts. The proposed approach differs fundamentally in that it uses contexts to better understand traffic conditions that can be expected to occur. This has the benefit that only an understanding of conditions in non-incident scenarios need be 'learnt' in training, and so not requiring a sufficient number of incidents to occur in training data.

Lam et al. (2016) presented an IDA is perhaps the most similar to the proposed approach, in that it incorporated contextual data into an IDA (it was reviewed in section 4.2.3). However, there are a number of key differences. Firstly, contexts were used because they were thought to affect the prior probability of an incident occurring. Hence, contexts were used to alter the sensitivity of the algorithm, rather than to understand what traffic conditions could be expected. Because different thresholds were used with different combinations of contexts, the IDA required a large amount of incident and traffic training data that would cover each incident state and each type of context. Because the proposed approach does not require incident data for training, and will attempt to 'learn' the effect of each context, it will need far less data for training, allowing for more feasible implementation in TMCs. Also, the presented IDA required manual calibration to incorporate each type of context, and so would require more time, effort and knowledge to implement in TMCs.

The presented traffic forecasting algorithm, RoadCast, is developed for the purpose of benefiting incident detection, but it is also thought to create a contribution to the field of traffic forecasting, which has applications in route guidance, and transport scheduling such as roadworks and logistics. RoadCast is novel in that it is the first to use contextual data within a machine learning algorithm that is capable of forecasting traffic conditions at a horizon of multiple days. It is also the first time that such a wide array of contextual data has been incorporated within a traffic forecasting algorithm.

The approach presented in this research project may not have been studied previously because of the significant amounts of data collection, storage, and computing power required, and the recent advancement of machine learning algorithms. This research project takes advantage of the recent technological advances described in section 1.4.5, to make original contributions in both the traffic forecasting and incident detection fields.

4.8 Approach benefits

A benefit of the proposed incident detection approach is that there is no need to learn the characteristics of incident patterns. Many machine learning IDAs detect incidents by looking for the characteristics of incidents in real-time data, which often results in complicated and time consuming calibration procedures (Williams and Guin, 2007). To learn the characteristics of local incident patterns, large amounts of incident data is often required, particularly in urban areas where congestion propagation is less predictable (as explained in section 2.1). Due to the fortunate rarity of incidents, this means that large amounts of real traffic data would be required to get a sufficient sample of incidents to learn from. As such, transport simulators are typically used to simulate incidents, but this in itself creates a calibration cost of time. By using an approach that only requires an understanding of conditions expected if no incident were to occur, these calibration requirements are avoided. The IDA would still need calibration in the form of collecting and training on sufficient amounts of historical traffic data. But the amount of traffic data required would likely be much less, and historical incident data would not be required. This would be more practical for implementing the IDA, particularly for locations without incident data, or where detectors had only been installed recently.

As discussed in section 2.4.8.2, it is thought that for an IDA to fully account for contexts, contextual data needs to be collected and incorporated. By incorporating contexts within a forecasting algorithm, a better idea of conditions that can be 'expected' at a particular time can be gained. By using the proposed approach, operators could be presented with a forecast of the 'expected' conditions, rather than just an alert. This could help in gaining an operator's trust and understanding of the algorithm. It may also be possible to indicate to operators how contexts are affecting the traffic. For example, if there is a queue over a detector, but the IDA has not raised an alert, it may be able to raise a message that it interpreted the queue as being caused by a nearby football match finishing. This would also aid an operators understanding of the algorithm and the current traffic state.

Finally, the proposed approach has an advantage in that it presents an opportunity to contribute to a number of different road transport applications, rather than just incident detection. Contexts describe all factors which could be expected to influence traffic conditions at a particular time. As such, they are key to a forecast of 'expected' traffic conditions. Such a forecast is key for many transport applications, including traffic signal strategies, roadworks scheduling, and public transport strategies. Typically, these applications use forecasts based on historical traffic data, and so do not account for contexts. By taking the opportunity from the technological trends described in section 1.4.5, this study aims to combine large amounts of traffic and contextual data to better estimate expected traffic conditions, which could provide contributions to the road transport applications described.

4.9 Conclusions

In this chapter, the problem focus of this research project was defined, and an approach to addressing it was developed. The basic approach taken is to develop an IDA that can incorporate contextual data in order to gain a better understanding of traffic conditions that can be expected to occur, such that it can more readily differentiate disruption from contexts and incidents. The proposed approach is slightly different to the existing IDA types reviewed, but takes elements from time-series and machine learning IDAs.

Traffic forecasting and context related literature were also studied in order to understand the tractability and originality of the proposed IDA, and to discover any techniques which may be effective within the proposed method. Firstly, recent advances in technology and algorithms suggested that the proposed approach was tractable. Some techniques in the literature appeared effective and suitable for the proposed IDA, but even so would need to be modified for the proposed approach, particularly in order to be suitable for implementation in TMCs.

The following chapters describe the development and evaluation of a methodology that follows the approach proposed in this chapter. Chapter 5 and 6 respectively describe the development and evaluation of the proposed traffic forecasting algorithm, RoadCast. Chapter 7 then develops this algorithm into the proposed IDA, and chapter 7 and 8 evaluates it in offline and online tests in order to prove or disprove this research project's hypothesis.

Chapter 5

RoadCast methodology

5.1 Introduction

A key part of the proposed IDA is an algorithm capable of accurately forecasting expected traffic conditions (named RoadCast). The aim of developing RoadCast is to forecast expected traffic conditions as accurately as possible, whilst remaining suitable for the proposed incident detection algorithm. In particular, it needs to be able to improve its accuracy by incorporating contextual data. If RoadCast achieves this, it will be developed into an IDA to address the research hypothesis, that IDAs can be improved with the incorporation of contextual data. The following sections describe the development of the RoadCast methodology.

5.2 Forecasting algorithm requirements

Before developing RoadCast, a set of requirements for the algorithm were set out. These requirements would ensure that RoadCast would be suitable for the proposed IDA.

5.2.1 Forecasting contexts

The algorithm should be able to forecast expected conditions accurately throughout non-incident situations, including during disruption caused by contexts. This is a key requirement that would allow an IDA based on these forecasts to differentiate this context from incidents.

As was discussed in section 4.2, it was seen as necessary to incorporate contextual data into the algorithm itself. This was because many contexts occur irregularly, and so schedules of such contexts would be needed so that their influenceon traffic conditions could be forecast accurately. As such, the forecasting algorithm would need to be able to incorporate contextual data in order to be able to improve its forecast accuracy.

To meet this requirement, RoadCast would need to be able to improve its forecast accuracy by incorporating contextual data. If successful, RoadCast would be developed into an IDA which would aim to improve its performance by incorporating contexts, and so this requirement would be key in addressing the research hypothesis.

5.2.2 Suitable target variable

The target variable being forecasted would need to be suitable for detecting incidents from. That is, the target variable must deviate significantly when incidents occur. The variable must also be over a short enough time period to detect incidents as quickly as possible. For example, monthly flows would clearly be unsuitable for incident detection. To ensure this requirement would be met, RoadCast would be developed on a target variable suitable for both traffic forecasting and incident detection.

5.2.3 Forecasting expected traffic conditions

The goal of the forecasting algorithm is to accurately forecast traffic conditions when no incidents occur, and inaccurately forecast conditions when incidents occur, allowing incidents to be detected from the error caused by the incident disruption.

Many forecasting algorithms are based on inference from recent observations of traffic conditions. These algorithms typically have horizons of up to an hour, and so are known as short-term forecasting algorithms (Vlahogianni et al., 2004). These algorithms would not meet the requirements set out in section 5.2 because during an incident, such algorithms could infer that the expected conditions would be similar to that of the recently observed incident. This could mean that no incident alert would be raised because the real-time data during incidents would be close to the algorithm's expected conditions. For the proposed methodology, during an incident, the forecasting algorithm should retain a forecast of expected conditions (i.e. conditions if no incident occurred), so that large errors would occur during incidents only.

The requirement of forecasting expected traffic conditions should be carefully considered when developing a forecasting algorithm. Each input to the algorithm must improve its accuracy for when no incidents occur, but not give the ability to forecast conditions accurately when incidents occur. The requirement means that many existing forecasting algorithms would not be suitable for the proposed incident detection algorithm.

To ensure that expected traffic conditions were being forecasted, RoadCast was developed to forecast traffic conditions at a horizon of up to one year. As no incident can last this long or be predicted to occur, this would ensure that RoadCast would not be able to forecast the disruption caused by incidents, and hence must only be forecasting expected traffic conditions.

5.3 Traffic data

Many decisions in developing the methodology of RoadCast were made based on the observations and preliminary tests made on real-world data. This decision was made to ensure that RoadCast would be suitable for the real-world application of incident detection.

The chosen datasets for this development was two years of loop detector data, and associated relevant contextual data, from Southampton, U.K. The first year of data was used to develop and train the algorithm, and the second was used to evaluate it.

This section describes the dataset used, and the reasons behind choosing it.

5.3.1 Location

Through the links made available by the University of Southampton, and this project's sponsor Siemens, there were a number of locations for which traffic data could be obtained for this project. This included Southampton, Cardiff, Coventry, Essex and Birmingham. Southampton was seen as the most suitable location, due primarily to the availability of data in this location. The data available covered a range of road, junction and land use types, a large time period of two years, and in a practical format for the task at hand. Southampton City Council provided the traffic data for this study.

The problem of congestion comes at a significant cost to Southampton, predominantly in the form of losses in productivity and fuel costs. According to a study by INRIX, it was estimated that Southampton drivers spent an average of 24 hours in delays in 2016, resulting in a cost to the city of £74 million, or £748 per driver (Cookson and Pishue, 2016).

However, Southampton was ranked as only the 18^{th} most congested in the U.K., in terms of average delay spent by motorists. Each of the top 25 U.S. cities were found to be more congested than Southampton, and each of the world's 25 most congested cities (covering 12 countries) were more than twice as congested as Southampton (Cookson and Pishue, 2016). Clearly then, the problem of congestion faced by Southampton is also faced across cities globally. Many ITS applications, including incident detection, aim to improve the state of congestion in these cities, and a key part of many of these applications is an accurate traffic forecast.

111 single inductive loop detectors around Southampton were used to collect traffic data for the study. Figure 5.1 shows the location of the detectors used.

The detectors used in this study were located on a range of urban road types, including urban arterials, streets and junctions. As stated in section 2.1, urban networks are more complex than other types of networks such as motorways, and so are thought to be more challenging to

forecast in. These detectors were installed primarily for the purpose of improving traffic signal management strategies, and so are installed on the stop-lines, approaches and exits of signalised intersections. As such, there may be a bias in this urban traffic dataset towards detectors in close proximity to signalised intersections.

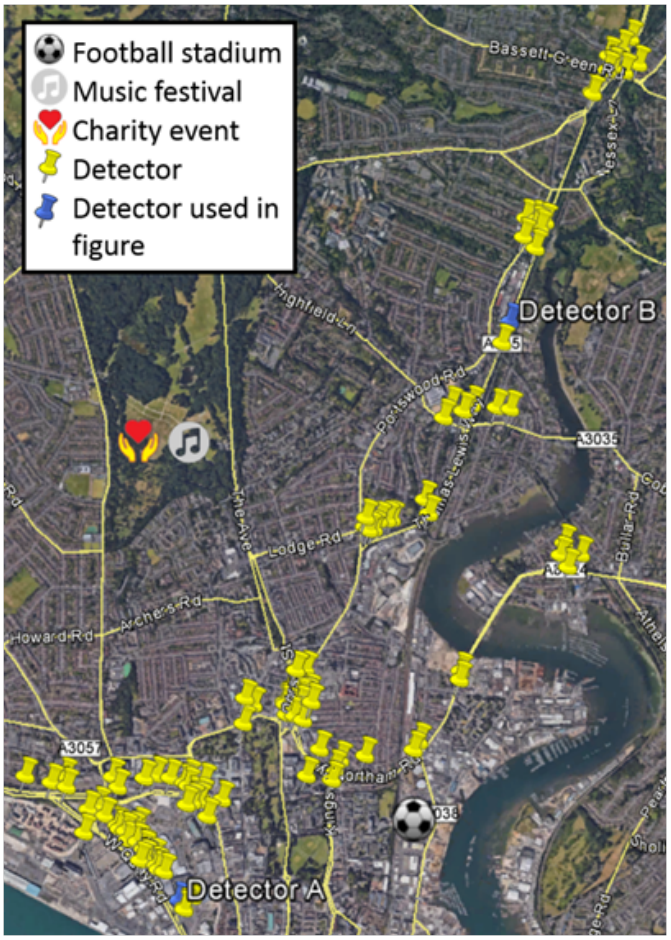


FIGURE 5.1: Locations of the detectors used in this study. This image was created with Google Earth.

5.3.2 Data description

726 days worth of loop detector data was collected from 16th March 2015 to 16th March 2017 (5 days of data were missing). The second year (16th March 2016 onwards) was to be reserved for testing the algorithm. As such, the first year was available for first developing the algorithm, conducting the preliminary analysis, and training RoadCast.

As described earlier, loop detectors detect the absence or presence of a vehicle at a point on a road by using the inductance of the vehicle to complete an electrical circuit. Every 250ms the detector next to the loop records the absence or presence of a vehicle as a "1" or "0". This data is then relayed to an Urban Traffic Control (UTC) system and processed into messages representing a five minute period. These messages include a number of variables, including flow, occupancy,

time mean speed (herein referred to as average speed), the average time gap between vehicles and the average loop occupancy time per vehicle. Flow and the average speed of vehicles in each five minute period (over the lane of the detector) were used as the target variables in this study. RoadCast would be implemented on each combination of detector and target variable separately.

Flow was estimated using the number of switches between one and zero from the detector in the five minute period, divided by two. The average speed values were estimations, generated using Cherrett et al's formula, as shown in equation 5.1 (Cherrett et al., 2001).

$$VS = 4(DL + VL)/N \tag{5.1}$$

where DL is the detector's effective magnetic length (meters), VL is the effective magnetic length of the vehicle (meters), N is the loop occupancy time of the vehicle (the number of ones produced each representing 250 milliseconds of occupancy), and VS is the vehicle's estimated average speed (meters per second) which is then converted to miles per hour. From this equation, average speed was estimated by taking the mean over the number of vehicles that passed in each five minute period. Because measured average speeds were found to differ slightly from these estimations, RoadCast should be seen as forecasting estimated values rather than true values of average speed.

These target variables were chosen because they were seen to be the most important for detecting incidents. Many previous studies use variables of flow and occupancy to detect incidents (Cook and Cleveland, 1974, Payne and Tignor, 1978, Persaud et al., 1990). Typically during incidents, congestion is caused upstream. The average speed variable can indicate this congestion as a drop in value. This variable was chosen over occupancy as average speed is more simple to interpret, and so the extent to which contexts' disruption can be forecasted can be seen more clearly. This congestion causes a drop in flow both upstream and downstream of the incident, which can be indicated by the flow variable.

Such five minute messages may be unsuitable for detecting incidents in the incident detection algorithm planned. This is because an incident would not be able to be detected until the next message had come in, which could be up to five minutes (longer than many reviewed in section 2.4 and those found to be used by many TMCs in section 3.2). However, this level of aggregation was seen to be sufficient for RoadCast (i.e. to evaluate the ability to forecast traffic conditions) because five minute aggregations were short enough to capture the variation caused by contexts (which vary more gradually than the sudden changes caused by incidents). If required, RoadCast could be retrained to make forecasts for data aggregated over shorter time periods.

5.3.3 Pre-processing

When studying the training traffic dataset, it could be seen that some detectors would at times return many consecutive messages of zero flow and average speed due to detector system fault. This could have been caused by detectors becoming faulty or being turned off. This could be seen clearly at times when multiple days of zero flow and average speed was observed, but at

other times it could be more difficult to judge.

To account for these unrepresentative messages, all messages with a flow value of zero were removed, along with the previous and next message (because a detector could start/end returning unrepresentative data at any point during a five minute period). This method would ensure that each message in the dataset would hold data returned by a detector, and so would reduce many unrepresentative messages. A number of other methods were considered to remove the unrepresentative data, including removing periods of zero flow of a certain length and/or times of day, using the degree of change in values and comparing values to historical averages. However, each of these methods would result in unrepresentative messages remaining in the dataset. As such, the described method was chosen for its simplicity, ease of implementation, and effectiveness.

However, clearly this method would also remove messages that were representative, i.e. when no vehicles passed the detector in the five minute period. This may introduce a bias in the results, for example as many representative messages at night time on quiet roads would be removed. However, it is thought that the comparison to other predictors would remain fair as the representative messages removed would not introduce an advantage for any of the predictors being tested. Of course in a real-world implementation, RoadCast would forecast every message in a certain time period into the future, but it could not be expected to accurately forecast the zero values returned when detectors are faulty or turned off.

After this, detectors which had fewer than 50 training messages or 50 testing messages were disregarded. Datasets of fewer than 50 messages for training or testing were seen as too small to get a representative view on the accuracy of the predictor. In this case, all 111 detectors had a sufficient number of messages. These messages passed each of the pre-processing steps described above, and so would be used in this study.

5.4 Approach choice

The methodology of the RoadCast algorithm was developed while studying the training data of the Southampton dataset, introduced in section 5.3. The aim was to develop an algorithm that could forecast expected traffic conditions as accurately as possible, whilst also meeting the requirements stated in section 5.2.

Firstly, a historical average predictor was developed as a benchmark for forecast accuracy. The best performing historical average was to take each combination of time of day and day of the week (i.e. forecast next Monday's traffic as the average of every previous Monday). This benchmark was chosen because of the commonality of use throughout the literature, both in the field of incident detection and traffic forecasting (Chrobok et al., 2000, Syrjarinne, 2016).

A number of different types of forecasting algorithm were developed and compared to try and improve on this benchmark, including regression techniques, neural networks, nearest neighbour,

decision trees and support vector machines. A preliminary test was conducted by using cross-validation on the training dataset of traffic flows, and with a performance metric of the mean absolute error. The results of preliminary tests of these algorithms is provided in table 5.1.

Algorithm	**Mean absolute error**
Linear regression	22.47
Support vector machine	10.25
K nearest neighbour	7.70
Historical average	6.66
Neural network (MLF)	6.52
Decision tree	6.25
Random forest	6.23

TABLE 5.1: Performance of implemented forecasting algorithms.

The only algorithm found to produce more accurate forecasts than the historical average predictor, while meeting the requirements of section 5.2, was the random forest. It was also found to consistently improve its accuracy when contexts were incorporated. As such, the random forest algorithm was chosen to be developed for use in RoadCast.

It should be noted that the method used to encode features was of critical importance for this problem, but each algorithm has different constraints as to how this can be done (e.g. neural networks cannot use categorical variables). Also, each type of algorithm can be designed and optimised in a countless number of ways. It was found that the feature engineering and design of the algorithm was of more importance in improving performance than the type of algorithm used. It would not be feasible to develop each type of algorithm to be as accurate as it possibly could be at the problem at hand. As such, the comparison of the accuracy of the developed algorithms presented above is not representative of the best possible performance of each type of algorithm for the problem at hand. The decision to go with the random forest after the preliminary tests was because it showed best potential to solve the problem, and so was developed further (into the final RoadCast algorithm).

Random forests were introduced by Breiman (2001), who took inspiration from Amit and Geman (1997). Random forests can be used to predict either a categorical target variable (classification), or in this case, a continuous target variable (regression). The method has been applied to a large and diverse set of applications in recent years, including cancer classification (Statnikov et al., 2008), image segmentation (Schroff et al., 2008), and aircraft engine fault diagnosis (Yan, 2006).

The number of variations of different types and designs of forecasting algorithm is vast, and so it would be infeasible to find the most accurate possible algorithm that met these requirements. As not every type and design of algorithm could be developed, there may exist a more accurate forecasting method that would also meet the forecasting algorithm requirements. The random

forest was chosen because it was consistently more accurate than the benchmark historical average, used contexts to improve its accuracy, and met the requirements for the proposed IDA. Once chosen, the random forest approach was developed to be as accurate as possible in forecasting expected traffic conditions. This algorithm, RoadCast, would help answer the question of whether IDAs can be improved by incorporating contextual data.

5.5 Benefits of approach

As well as meeting the requirements stated, and providing the most accurate forecasts found, a number of other benefits to using a random forest approach were apparent. The following sections describe these benefits.

5.5.1 Machine learning

The benefit of using a machine learning approach is that a great deal of the calibration process can be automated. This includes the process of 'learning' the influenceeach context has on traffic.

A key requirement of TMCs was found to be minimal time and effort required to implement an IDA (see section 3.2). As such, the calibration requirements of the forecasting algorithm were seen to be important, particularly in this case as many types of context data required incorporation into the algorithm.

Random forests are known as robust algorithms that can create accurate predictions with little calibration (Leo Breiman, 2017). They require very few parameters to be tuned during training, which can be set using trial and error over a training data set.

5.5.2 Prediction interpretation

Many complex and machine learning techniques can be seen as a black-box. That is, it can be viewed in terms of its inputs and outputs, but the understanding of how input is turned into output cannot be gained. However, for random forests there do exist methods to interpret its predictions (Strobl et al., 2007, Palczewska et al., 2014). These methods are later explored for RoadCast in chapter 6.

This method was found to be important for the incident detection application, rather than the forecasts themselves. The ability to understand why a particular traffic state was predicted was seen to be a beneficial feature to TMCs. For example, if sudden heavy congestion was predicted and observed, but no alert was raised, the algorithm could indicate that the congestion was expected because of a football match taking place. Or if heavy congestion was observed, an alert was raised, but a TMC operator thought that this was a false alert, the algorithm could justify the alert by indicating that this congestion was in fact not expected because it was during a bank holiday where congestion levels were typically lower.

5.5.3 Fast training and testing times

With the Southampton dataset, RoadCast took 8.1 minutes to train per detector. Predicting every message in the test data (363 days) took 0.25 seconds per detector. This was achieved on an Intel(R) Core(TM) i7-6700, 3.40Ghz, 16GB RAM.

This training time was faster than many other machine learning methods considered (this will be backed up when the results of the comparison of algorithms is completed). The quick training time of the algorithm was seen to be of particular importance if RoadCast would need frequent retraining. This may be the case if weather data was included, as RoadCast could be retrained whenever the weather forecast changed.

5.6 Random forest theory

Although the random forest algorithm had been chosen for use in RoadCast, there were still many open questions as to the exact design of the algorithm to be used. The RoadCast algorithm was developed using the Scikit-learn library in Python (Pedregosa et al., 2011, scikit-learn, 2017*a*). This section describes the random forest algorithm used in RoadCast.

The random forest algorithm used in RoadCast is described in the following algorithms. Algorithm 2 describes how many decision trees (algorithm 1) are combined into the random forest used in RoadCast. Each detector and target variable combination used a separate random forest. Louppe (2014) provides further explanation of the random forest method employed in this study.

Algorithm 1 Decision tree algorithm

1: **procedure** TRAINING(set of training messages Z^{tr})
2: Create a root node B_0 and assign all training messages Z^{tr} to it
3: **do**
4: Find the leaf node B_i with the most messages assigned to it
5: Create child nodes B_j and B_{j+1} from B_i
6: From a random subset of features of size S, find the attribute a to split B_i's messages into two subsets, such that the sum of each subset's variances (in terms of the target variable) is reducesd
7: Assign B_i's messages to B_j and B_{j+1} according to their value of a in the split
8: **while** every leaf has more than M messages assigned to it
9: **end procedure**

Random forests simply take the average of the forecasts of many decision trees. The idea of each decision tree is to form subsets of the training data that share the same (or similar) values of features, and similar target variable values. This means that to forecast a new message, the decision tree gives an average over values in the training data that have similar feature values.

First, all training messages are stored in a root node. These messages are then split into two subsets based on an attribute, i.e. their value of a feature. An example of an attribute could be

Algorithm 2 Random forest algorithm

procedure TRAINING(set of training messages Z^{tr})
  for a pre defined number of trees K **do**
    Create a bootstrap random sample Z_r^{tr} from Z^{tr} of size $|Z^{tr}|$
    Create a decision tree T_r with Z_r^{tr} using algorithm 1
  end for
end procedure
procedure TESTING(set of testing messages Z^{ts})
  for each message x in Z^{ts} **do**
    Predict a value y_i for message x using each of the decision trees $T_1 ... T_k$
    Return the mean of the predicted values $\bar{y}$
  end for
end procedure

'time of day'>13.5, meaning all messages that occurred after 13.5 (1.30pm) would be put into one subset, the rest in the other subset. The idea is to create a split that results in two subsets that each have the smallest variances in terms of the messages' target variables. This splitting procedure continues until a stopping criterion is met, i.e. the criterion in which it is decided that nodes will be split no further. Once the stopping criterion is met, the final decision tree is formed.

Then, to predict a new message's target variable, the message would move down the tree based on its value with respect to each of the splits, until it reached a leaf node, i.e. a node at the end of a tree. The decision tree's prediction is then the average of the training messages' target variable values in this leaf node.

A random forest is simply a collection of many decision trees, where each tree is created using a random subset of the training messages, and chooses a random subset of the available features to create each split. A random forest's prediction is simply the mean of each of the decision tree's predictions.

It is thought that random forests typically achieve superior results to decision trees by reducing variance and bias in the final prediction (Li, 2017). This is done in three main ways; by creating trees from bootstrap samples of messages, using random subsets of features at each split and by averaging over many trees. It should be noted that a bootstrap sample is a type of sample where samples are drawn with replacement (Singh and Xie (2008) provides explanation of the bootstrap method). Bootstrap samples and random subsets of features mean that the trees created are more independent, so the gains from averaging over a large number of trees can reduce variance substantially (Li, 2017). By using random subsets of features, each feature has more opportunity to be used in a split, meaning bias typically increases from splitting data on more randomly selected features early, but variance is reduced by averaging over more independent trees (that consider more features) (scikit-learn, 2017*b*).

It should be noted that the parameters of a random forest can be changed in order to change the behaviour of the algorithm. For example, the stopping criterion can take different forms. In this case, the stopping criterion was chosen to be the minimum number of messages each leaf

has assigned (referred to as 'min_samples_leaf' in figure 5.2), M, because this criterion resulted in consistently improved accuracies over other criterion's such as the maximum depth of the tree (referred to as 'max_depth' in figure 5.2), and the minimum number of messages required to split a node (referred to as 'min_samples_split' in figure 5.2). Figure 5.2 shows the flow cross-validation score of the training data using different stopping criterions, with pre-set values to be used across all detectors. As can be see, 'min_samples_split' and 'min_samples_leaf' stopping criterions consistently outperformed 'max_depth', and had clearly lower minimum average cross-validation scores.

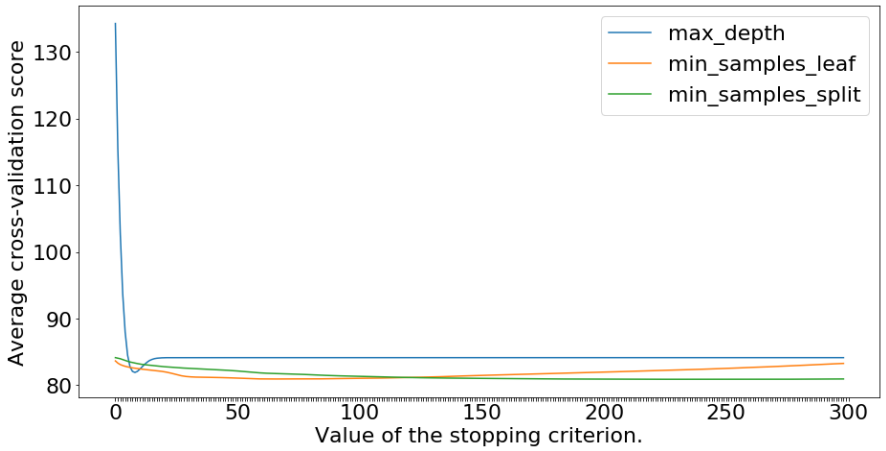


FIGURE 5.2: Cross-validation score when predicting flow, of the random forests with various stopping criterions used.

Parts of trees that were not during contexts' disruption were seen creating leaves representing a particular time of day and day of week. These leaves could not be split further because they were pure, i.e. no other contexts could be used to split them (as no other contexts occurred during these messages). Hence, the stopping criterion was thought to only make differences during contexts.

Using the number of messages in each leaf rather than the maximum depth was found to improve accuracies because it meant that contexts that occurred on more messages would be split further than those with few messages. This worked in this case because rare contexts (such as the half marathon) would reduces variance by averaging over many messages, but more frequent contexts (such as football matches) could be split further so that the sudden change caused could be 'learnt'. The maximum depth would instead result in subsets that may have needed further splits during frequently occurring subsets, or less during rarely occurring contexts.

Another key parameter used was the size of the random subset of features used for each split, S. A value of one would mean that the feature used to split messages at each node would be chosen randomly, but higher numbers would mean that all possible splits over many features would be considered.

5.7 Developing feature encoding methods

The development of contextual features would be crucial in enabling RoadCast to improve the accuracy of its forecasts, and hence allowing the proposed IDA to improve its performance with the use of contexts. The features would also need to be development in accordance with the the requirements set out in section 5.2, so that RoadCast would be suitable for the proposed IDA.

A preliminary analysis of the Southampton training dataset was undertaken to develop the contextual features. At the start of the analysis, RoadCast was found to be far less accurate with contexts than without. At this time contexts were encoded in a simple manner, such as a Christmas feature being encoded as the number of seconds until the start of Christmas day. RoadCast would overfit to contextual features because there were so few messages in which they disrupted the training data. Only with the use of the carefully encoded contexts that are described in the following sections, were improvements in RoadCast's accuracy found.

The following sections describe the findings of the preliminary analysis, and the final method used to encode each feature.

5.7.1 Time

There are two commonly used traffic forecasting approaches, using historical patterns of traffic conditions, or by inferring from recent observations of traffic. The first method attempts to find patterns in large amounts of traffic data, such as similar conditions on a particular day of the week or time of day. Forecasts are then made by matching the message being forecasted to historical data, with respect to the patterns found. The second method takes recent observations of traffic conditions, and models how the conditions would be expected to evolve in the near future. If using a machine learning algorithm, this method may 'learn' how conditions typically change in particular situations, such as average speeds typically falling after flows reach a particular value.

Chrobok et al. (2004) found that the most effective forecasting algorithm at a 5 minute horizon was short-term algorithms based on real-time and recent traffic data, but as the horizon rises to 120 minutes, the most accurate method changes to heuristics based on historical data. This is because traffic conditions vary little over short time periods, so the evolution can be modelled, but over longer time periods there is more variation, meaning the evolution is more difficult to model. Instead, inference from patterns in historical data becomes a more accurate method over longer time periods.

As stated in section 4.5.1, algorithms using recent observations could be influenced by recently occurring incidents, and so would not be forecasting the required 'expected' traffic conditions (set out in section 5.2). To ensure that 'expected' conditions were being forecasted, RoadCast did not use recent observations, but instead was designed to infer from historical patterns.

While studying the Southampton training dataset, the features with greatest correlations with the flow and average speed target variables were the time of day and day of the week (using the chi squared statistical test). These features could also visibly be seen to cause the most variation in flow and average speed values. After conducting preliminary tests on the Southampton training dataset, it was clear that these two features would be most effective as the base features of RoadCast's forecasts, and so were chosen for use. This finding is also consistent with the findings of the development of many other predictors in the literature (Stathopoulos and Karlaftis, 2001, Weijermars, 2007). The time of day was encoded as the $hour + minute/60$, and the day of the week was an integer ranging from 0 to 6. These features would be key in allowing the random forest to 'learn' the patterns of historical data between the time, day of the week, and contexts.

5.7.2 Events

Event contexts are typically local to a TMC's particular road network (e.g. a city), and can cause disruption in a variety of ways. All events that appeared to influence Southampton's traffic were included in RoadCast. This included a football, cricket, half marathon, boat show, music festival and charity event feature. Each type of event is a separate feature, and each has an encoding based on the time until the nearest start time of the event.

Event features were given a value of 10 if they were not within a particular time duration to the event. Event contexts that were expected to only disrupt traffic conditions on one day were called single day events, and were encoded as the time until the start time of the nearest event if on the day of the event, 10 otherwise. Event contexts that could cause disruption across multiple days, i.e. multiple day events, were encoded as the time until the start of the event if during the event, 10 otherwise. In the case of the Southampton dataset, the football, cricket, half marathon and charity event contexts were encoded as single day event features. The boat show and music festival contexts were encoded as multiple day event features.

The decision to encode each feature as 10 if not within a certain proximity to the start time of an event occurrence was made because RoadCast was found to overfit to spurious patterns in some values of contexts in the training data. For example, RoadCast could split a node based on higher/lower than 4.5 days to the nearest football match. One can see that this split was unlikely to have a real bearing on the traffic conditions at this time. However, RoadCast made splits such as this when there were few samples in a given node, and they happened to be correlated in this way. By using domain knowledge to set message's football context values as 10 if they were not on the day of a match, RoadCast was forced to not split on this feature for messages not on the day of a match. Hence, these spurious patterns in contexts were less likely to be found, meaning overfitting would be reduced.

From comparing cricket match schedules to RoadCast's errors, it was found that only international cricket matches had a noticeable effect on traffic (domestic matches had no effect). This highlighted a challenge in using contextual data relevant to the study area. Intuition is needed to find which contexts to collect. When implemented in a new study location, either

local knowledge or basic data analysis would be needed to initially decide which event contexts (such as particular music festivals or football stadium matches) could affect the location's traffic conditions, and so would be relevant to RoadCast.

School term dates were also encoded as a multiple day event feature, as detectors near schools typically had higher flows during term time. Detectors nearby the University of Southampton and Solent University were also studied. After developing various features related to term dates, graduation dates and 'moving weekends' (the weekends adjacent to the start and end of terms), no significant accuracy improvement was found. There are a number of reasons that could provide a reasonable explanation to this, including; the less centralised nature of Universities (unlike other contexts such as concerts), the diverse range of travel modes to Universities, and the wide spread of commuting times (student's timetables vary). Because these University features did not improve RoadCast's forecasts, and often led to over-fitting, none were included in RoadCast.

5.7.3 Holidays

Christmas, Easter and public holidays could visibly be seen to have noticeable effects on traffic, but each typically influenced traffic in different ways. As such, each type of holiday was included as a separate feature. Contextual features were developed based on the results of preliminary tests. It was decided that holiday contexts could be encoded in the same way as multiple day event contexts. This method was found to be effective, and would allow for a more generic set of methods of encoding contextual features. Easter and bank holidays were encoded as multiple day event features, i.e. the time until the start of the holiday if during the holiday, 10 otherwise. The holiday in these cases was defined as the start of the first non-working day of the holiday weekend, until the end of the last non-working day of the holiday weekend.

However, during preliminary tests it was found that the Christmas feature required encoding differently to the other holiday features because of the importance of the time until a Christmas Day. Because Christmas Day occurs on different days of the week each year, this feature was found to be far more effective when using the difference in time to Christmas Day than using the time until the start of the holiday. As such, this type of feature would be called a multiple day event (with reference). The encoding of this type of feature was decided to be the time until the reference date if during the event, 10 otherwise. In the case of Christmas, this means the time until Christmas Day if during the holiday, 10 otherwise. Here, the holiday was defined as the first non-working day before Christmas Day until the first working day after New Year's Eve.

On many detectors, RoadCast's first split of the data was done by the 'day of week' feature, meaning contexts that did not occur on the same day of the week in training and testing sets may not have been accounted for during testing. For example, Christmas occurred on a Friday in 2015, so the algorithm's decision trees may only have split on the Christmas context over messages that occurred on a Friday. This would mean that when tested on Christmas Sunday in 2016, the 'Christmas' feature would not have been used. To address this, a 'modified day of week' feature was created. This feature was 0 during multiple day events (with reference), and

1-7 otherwise (representing the day of the week). This modification tackled the problem above by ensuring that all messages during multiple day events (with reference) would be in the same part of each decision tree. For example, if the Christmas feature was the only multiple day event (with reference) at a particular detector, this change would allow the Christmas feature to split messages depending on their date with respect to Christmas, rather than their day of the week.

An 'any holiday' binary feature was also found to improve accuracies on detectors which exhibited similar traffic conditions during different holidays. This meant that variance could be reduced by averaging across messages from different holidays. However, when trained on periods that only covered one holiday, unrepresentative forecasts could be made. For example, when trained on data between November and February, and tested on data between February and June, forecasts of Christmas like conditions were created for Easter. Clearly this is a situation caused by a lack of representative training data, but in many real-world situations, a whole year of training data may not be available. To ensure that RoadCast would be transferable to such situations, the 'any holiday' feature was not used.

To account for any long term variation, seasonal features were also investigated. Features such as 'month of year', 'week of year' and 'day of year' made no improvements in RoadCast's accuracy. As such, seasonal features were not included in RoadCast in this study. However, it is thought that if multiple years of training data were available, longer term seasonal patterns may be found, which could improve RoadCast's accuracy. Improvements may also have been found if the study was in a location with more seasonal variations, such as popular tourist destinations. As such, seasonal features should be considered when implementing RoadCast in new locations, particularly if multiple years of training data is available.

5.7.4 Weather

The impact of weather on traffic conditions has been extensively covered in the literature (Andersen and Torp, 2016). The majority of data driven studies found that urban flows reduce during adverse weather conditions such as rainfall (Changnon, 1996, Al Hassan and Barker, 1999, Goodwin, 2002). Adverse weather is also thought to reduce road capacity, increasing the likelihood of congestion (Lam et al., 2013, Kamga and Yazici, 2014, Zhang et al., 2015). Some have found significant traffic disturbances due to extreme weather events such as snow storms (Nookala, 2006). However, forecasting studies have found that weather information contributed little to forecast accuracy and could even lead to over-fitting (Bajwa and Kuwahara, 2003, Zhou et al., 2014).

When analysing weather contexts for RoadCast, historical weather forecasts were not available. Instead, historical observations were used to understand whether RoadCast's forecasts could improve with accurate weather data. Many features based on these observations were developed, including rainfall amount, temperature, and sunset times. However, none of the features developed created a clear improvement in RoadCast's forecasts. Another issue is that when forecasting into the future, clearly RoadCast would need to use weather forecasts. These forecasts

may differ from observations, particularly when using a forecast horizon of a year. For these reasons, weather contexts would not be included in RoadCast in this study.

It should be noted that the weather may influence traffic conditions significantly more in other locations (Nookala, 2006), and so weather context could be considered when implementing RoadCast elsewhere. However, if included, the need for accurate weather forecasts may limit RoadCast's forecasting horizon, and so could only be used for certain ITS applications.

5.7.5 Other contexts

A number of other contexts were considered in this preliminary analysis that were not included in RoadCast. This included other events, such as Eastleigh FC football matches and cruises, which did not have a noticeable effect on Southampton's traffic.

By comparing information from local news reports to detector's traffic conditions, it appeared that planned roadworks caused visible disruption at some detectors. Unfortunately, reliable historical roadwork data was not available in Southampton, and so this context was not used. However, it was thought that a contextual feature for planned roadworks would be unlikely to improve forecasting accuracies. This is because roadworks occur in different locations, at different times, and cause different amounts of disruption each time. It is unlikely that enough representative training data could be collected to cover a sufficient number of scenarios of roadworks disruption at each detector. Without sufficient training data, RoadCast would not be able to 'learn' how a particular instance of planned roadworks in the future could be expected to affect traffic conditions.

New Year's Eve appeared to cause a spike in travel demand around midnight on the 31st December. However, it could be seen that the Christmas holiday feature itself accounted for this variation. RoadCast 'learnt' that this increase in flow occurred when the value of the Christmas feature was around -6 (i.e six days after Christmas Day).

Previous studies have incorporated traffic signal timing data into their forecasts to account for their disruption. This data was not available in this study, and so could not be included. This context would also greatly reduce RoadCast's forecast horizon.

5.7.6 Standard encoding methods

It was clear that the most effective methods to encode contextual features was consistent between certain contexts. It is likely important that the process between collecting contextual data and RoadCast's forecasts be as simple and automated as possible when implementing RoadCast in the real world. As such, the encoding methods were re-developed into a set of standard encoding methods which would describe how contexts could be encoded within RoadCast in general. These methods were designed to be transferable, so that when implemented in new locations,

the methods could be re-used to encode contextual data into features for use in RoadCast (for example if a different network had different local events that required contextual features). This was seen as important because if the same methods could be used to encode contexts in different locations, RoadCast could be implemented without the need for a preliminary analysis of contexts (as was undertaken in Southampton). Instead, the only necessary manual processes would be to initially choose what contexts to collect data for, and to collect their start times.

The table below describes the standard methods used to encode contexts in this research project. These methods would be used in later tests of the developed algorithms.

Feature type	**Standard encoding method**	**Features used in Southampton**
Time of day	Hour of day + (minutes/60)	Time of day
Day of week	Integer ranging from 0 to 6	Day of week
Modified day of week (used when a multiple day event (with reference) feature is included)	7 if during a multiple day event (with reference), integer ranging from 0 to 6 otherwise	Modified day of week
Single day events	The number of days + (hours/24) + (minutes/1440) to the nearest start time of an occurrence of the event (note times before start times are negative) if on the day of an event occurrence, 10 otherwise	Football matches, cricket matches, half marathon event, charity event
Multiple day events (without reference)	The number of days + (hours/24) + (minutes/1440) to the nearest start time of an occurrence of the event if during an event occurrence, 10 otherwise	Easter, other public holidays, boat show, music festival, school dates
Multiple day events (with reference)	The number of days + (hours/24) + (minutes/1440) to the nearest reference time of an occurrence of the event (note times before a reference time are negative) if during an event occurrence, 10 otherwise	Christmas

TABLE 5.2: Standardised methods of encoding each type of context.

Clearly, the methods developed found were not comprehensive in covering all possible contexts in road networks generally. For example, severe weather in other locations could be used to improve forecasts if encoded within RoadCast, but in the case studies used it was found to influence conditions negligibly, and so such features were not used. Despite this, the standard methods described can be seen as a starting point for which to encode particular contexts in new locations in an automated way.

5.7.7 Summary

This preliminary analysis was undertaken to understand how data from influential contexts could be encoded into a set of contextual features that would improve RoadCasts forecasts. This resulted in an understanding of how contexts disrupted traffic conditions in Southampton, and led to a set of standard encoding methods to encode different types of contexts. This set of methods would help ease the calibration process when implementing RoadCast in a new location.

5.8 RoadCast feature and hyper-parameter selection algorithm

Urban road networks exhibit a diverse range of traffic behaviours, so it is a challenge for urban forecasting algorithms to generalise well between different detectors. During preliminary tests, RoadCast was found to overfit to contexts that were known not to have any effect on a detector's traffic conditions. To avoid this issue, ideally RoadCast would only take input from contexts relevant to the detector and target variable in question. Similarly, the most effective parameters of the random forest appeared to differ between detectors and target variables. For example, detectors with more noise were found to be more accurate with earlier stopping criterions, because this allowed their forecasts to be an average of more messages, minimising overfitting on the noise variation.

An algorithm to automatically select features and hyper-parameters for each random forest was hence developed. The algorithm is herein referred to as the 'selection' algorithm. The selection algorithm aimed to enable the random forest at each detector and target variable to use only relevant contextual features and optimal parameters. That is, the algorithm allowed RoadCast to tailor its parameters and input features to the nature of the traffic conditions present at each particular detector. Important to this algorithm was that it required no manual input, meaning it would not add to the manual calibration requirements of the proposed IDA. This algorithm would differentiate RCID from IDAs reviewed in section 2.4, many of whom would either implement the same algorithm at every detector, or would require manual calibration to do so, such as by manually setting thresholds.

By using this selection algorithm, it was hoped that the process of implementing RoadCast (and hence RCID) at each detector and target variable would be made easier. The process is as follows. Firstly contextual and traffic training data would be collected, and contextual features would be encoded. Then, once the automatic selection algorithm had been run, RoadCast would be capable of using planned contexts to create forecasts into the future.

Algorithm 3 is a psuedocode of this algorithm.

A key part of the selection algorithm is a k-fold cross-validation score (herein referred to as 'score'). A cross-validation score is a method to gain an idea of the accuracy of an algorithm using the available training data. The method first splits the training dataset into a number, k, folds. Then for each fold, the fold is used as a test dataset, and the remaining data is used for training. After the training and testing is completed with each fold, an array (of length equal to the number of folds) or errors (in this case mean squared error) is obtained. The average of these errors is the cross-validation score. In this study, 10 fold cross validation was used.

The first procedure in the selection algorithm is to choose the contextual features that will be used. This aimed to reduce the use of features that could cause the model to overfit, while keeping features that improve the algorithm's accuracy. In this case, for the purposes of balancing

Algorithm 3 RoadCast selection algorithm

procedure CONTEXT INCLUSION(set of training messages Z^{tr}, set of contextual features A)
Shuffle the order of the messages
Set the benchmark score as the score on Z^{tr} with 'time of day' and 'day of week' features only
for each feature in A **do**
if the score improves when the feature is added **then**
Add the feature to the algorithm
end if
end for
Set the benchmark score as the score with the features currently in the algorithm
do
Remove the contextual feature which achieves the best score when removed
Set the benchmark score as the score with the features currently in the algorithm
while there exists a feature which improves the score when removed
if a multiple day event (with reference) feature is included **then**
Replace the 'day of week' feature with 'modified day of week'
end if
end procedure
procedure GRID SEARCH HYPER-PARAMETER SELECTION(set of training messages Z^{tr}, set of features included in the algorithm F)
for all $M \in [2, 5, 10, 25, 100, 200]$ **do**
for $S = 1$ to $|F|$ **do**
Find the score with parameters M and S
end for
end for
Return the parameters that achieved the best score, M^* and S^*
Retrain the algorithm on all available training data with parameters M^*, S^* and K
end procedure

computation time with accuracy, this procedure was completed with RoadCast using parameter values of $K = 10$, $M = 1$ and $S = 1$. Different values of these parameters could be used depending on the user's preference for computation time or accuracy.

The first step of the procedure is to shuffle the order of messages in the training dataset. This particularly aids in deciding whether rarely occurring contexts (such as Christmas) should be included in the algorithm. By shuffling messages, it is much more likely that each training and testing set during cross-validation includes instances of the rarely occurring context. This means that the cross-validation score will give a better indication of whether a particular context does improve RoadCast's forecasts.

The next step is to use a backwards recursive feature elimination process to select features (scikit-learn, 2017*c*). The idea of this process is to repeatedly train and test the algorithm with different features on subsets of the training data until the best performing combination of features is found. In backwards selection, the random forest is first trained using all the features, and the feature deemed least important is removed. This process continues until a stopping criterion is reached. In this case, the least important feature was found by removing each feature in turn and ranking the algorithm's accuracy. The stopping criterion was the iteration at which removing any feature would result in a reduction in accuracy. Also, in this case, instead of

initially training with all features, only the features which would improve accuracy when used with 'time of day' and 'day of week' features were considered. This was done to reduce the computation time of the algorithm.

The last step was to replace the 'day of week' context with 'modified day of week' when a multiple day (with reference) context is included. The reason for using this feature was described in section 5.7.3. After this, the procedure would return the contextual features that would be used for the random forest dedicated to the given detector and target variable.

The second procedure is to select the values of hyper-parameters in the random forest. By finding values of these parameters that perform well on the training data (using cross-validation), the aim of this procedure is to improve RoadCast's accuracy by tailoring the structure of the algorithm to each detector and target variable. As can be seen in algorithms 1 and 2, the random forest has three tuning parameters, the number of trees grown K, the size of the subset of features used to split on S, and the stopping criterion, which in this case was chosen to be the threshold number of messages required at each leaf, M. In this case, the values of M and S were chosen to be set at each detector and target variable separately, using the selection algorithm. This was done because each target variable and each detector was found to achieve better results with different values. High values of M (i.e. an early stopping criterion) were typically used when predicting values at detectors with high noise, such as detectors near traffic signals. By having a high value of M, predictions would be of averages over larger subsets, reducing variance. Low values were used when predicting less noisey values (i.e. many detectors predicting flow), and when many contexts were included. This would result in predictions of averages over smaller subsets, meaning the more fine patterns over many contexts could be 'learnt'.

The method to set the value of M and S was done using grid search, that is, finding the cross-validation score with each combination of M and S, and returning the values which achieved the best score. The last step of this procedure is to retrain the random forest on the entire training dataset, using the best scoring M and S values. In this case, this retraining was done using $K = 100$ in order to create a model with good accuracy. The choice of K is discussed further in the following paragraphs.

The number of trees used, K, was not altered for each detector and target variable. This was because generally the algorithm performed more accurately when using more trees on all detectors and target variables. Figure 5.3 demonstrates this. Here, the 10 fold cross-validation score of a random forest with M=1, S=1 was calculated on the Southampton training dataset, using the time of day and day of week features only, and the score was averaged over all of the detectors. As can be seen, the accuracy of the random forest tends to improve when more trees, K, are included in the forest.

The K parameter can be seen as a trade-off between accuracy and computational cost. Hence, during the selection algorithm, a low value of $K = 10$ was used (though high enough to achieve

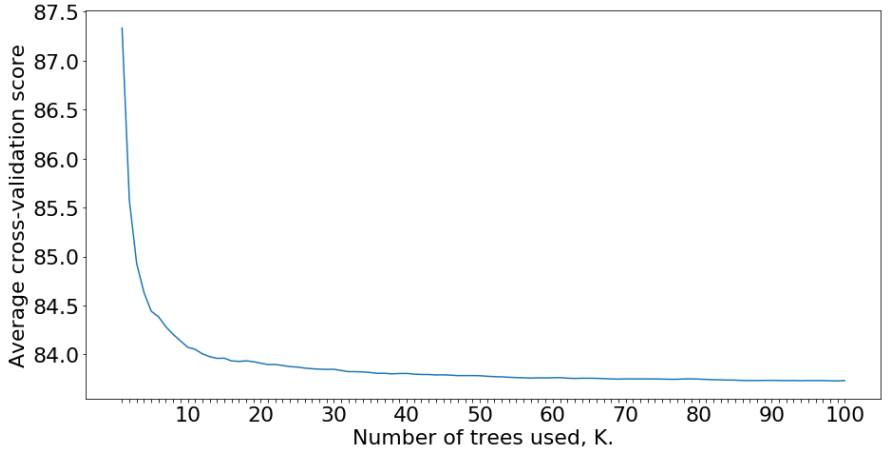


FIGURE 5.3: Cross-validation score of the random forest with various numbers of trees used.

representative results that would indicate which contexts and values of other parameters performed best). The choice of $K = 100$ for retraining the random forest at the end of the procedure was made because the accuracy improvements appear to tail off above this value, as can be seen in figure 5.3.

The selection algorithm used the context inclusion procedure first because contexts were found to affect accuracy more so than altering parameters M and S. The initial parameter values $M = 1$ and $S = 1$ were chosen as they were found to achieve the most stable and accurate results during the preliminary analysis, and $K = 10$ was found to be sufficiently accurate and stable whilst not requiring excessive computation time. Using these initial values of parameters, the context inclusion procedure was found to consistently include all contexts which were known to be relevant to the detector and target variable used (by observing the disruption caused by context in the traffic data), but not include irrelevant contexts. Once the context inclusion procedure was run, the parameter optimisation procedure would be run to fine tune the algorithm by altering values of M and S, which was found to lead to consistently more accurate forecasts.

On the initial Southampton training dataset, RoadCast took 1.5 minutes to train per detector (on the whole year of data). However, the automatic selection algorithm requires multiple iterations of training, so the total optimisation time was 8.1 minutes per detector. Predicting every message in the test data (363 days) took 0.25 seconds per detector. This was achieved on an Intel(R) Core(TM) i7-6700, 3.40Ghz, 16GB RAM. The selection algorithm took 8.1 minutes to complete per detector (or 178 detectors per day). This time was seen to be reasonable as the selection algorithm would not need to be run frequently. Once the selection algorithm had been run once, RoadCast could be simply retrained on updated training data using the given features and parameters (unless there had been a major road network or context change).

The trade-off considered when creating the selection algorithm was between computation time and finding the combination of features and parameters with best accuracy. Ideally, every combination of features and parameters would be considered, but this would lead to a very long computation time. The algorithm presented was seen to be effective as it would typically choose

the same contexts and parameters than would have been chosen by manual trial and error, and it would be completed in a reasonable time.

When implementing RoadCast in a location different than Southampton, some contextual features could be re-used (such as Christmas), but others would need to be changed in order to be relevant for the particular location (such as local sporting events). The new features could be developed using the standard methods described in this section. For example, if a concert was thought to influence traffic in a particular location, this could be encoded in the same way as other single day event contexts in this study.

When implementing RoadCast in a new location, the first step would be to identify possible relevant contextual features for the location, then encode them using the standard encoding methods described in table 5.2. Then once each context had been encoded, the automatic selection algorithm would decipher whether the context each relevant to each detector (if any). Hence, when incorporating contexts within RoadCast in a new location, the only necessary manual processes would be to initially choose what contexts to collect data for, and to collect their start and end times. The selection algorithm, together with the standard encoding methods, will allow RoadCast to better generalize to new locations, while requiring less calibration time, effort and expertise.

5.9 Originality

RoadCast is original in that it incorporates contextual data within a machine learning algorithm to forecast road traffic conditions up to a year ahead of time. Most traffic forecasting algorithms are based on recent observations of traffic conditions, whereas RoadCast is based on the patterns found in historical traffic datasets. This difference allows RoadCast to forecast at such a horizon, and makes it more suitable for the incident detection application, i.e. for forecasting 'expected traffic conditions'. RoadCast also differs from previous studies in that so many contexts were studied, and so many were found to improve its accuracy. Compared to previous studies, this is the most in depth analysis of how contexts can be incorporated into a traffic forecasting algorithm.

A key aim when designing RoadCast was to make it as automated as possible. A novel set of methods for encoding different types of features were created so that contexts in new locations could be easily incorporated with minimal manual effort. A novel selection algorithm was designed to automatically select relevant contexts and and the random forest's hyper-parameters. RoadCast could then use training data to automatically 'learn' the effect each context had on a detector's traffic conditions. These processes mean that if RoadCast is implemented in new locations, minimal manual calibration is required to achieve accurate forecasts.

5.10 Conclusions

This chapter described the methodology of the RoadCast traffic forecasting algorithm developed. The described methodology would be used when evaluating RoadCast, and RCID later.

The next step of this research project was to evaluate RoadCast in order to assess its suitability for the proposed IDA. If this traffic forecasting methodology could be shown to provide accurate forecasts, benefit from the incorporation of contextual data, and meet the forecasting requirements stated in section 5.2, it would be deemed suitable for development into the proposed IDA.

Chapter 6

Evaluating RoadCast

6.1 Introduction

This chapter provides an evaluation of RoadCast's ability to forecast traffic conditions. The aim of this evaluation is to assess RoadCast's performance and suitability for the proposed IDA. If found suitable, it would be used as the basis for the proposed IDA.

These tests would use the Southampton dataset described in section 5.3. The first year of data would be used for training RoadCast, and the second year for testing.

To make this evaluation, RoadCast will be tested under a number of scenarios with different forecast horizons and amounts of training data, and will be compared to a historical average, which is a commonly used predictor in ITS applications. Based on these tests, an implementation procedure will be recommended for RoadCast's potential use in real-world applications.

A number of interpretation methods also will be implemented. These methods will attempt to improve understanding of RoadCast's decision making process, and in particular its use of contexts in forming its forecasts. This analysis would contribute to the understanding of how traffic forecasting can use contexts to improve their accuracy. It was also explored whether these methods could be useful in real-world applications, such as an IDA having the ability to explain to the user what the cause of congestion is.

6.2 Contextual data

A wide range of contextual data was collected with the aim of developing contextual features that would improve the accuracy of RoadCast's forecasts. Contextual data was analysed, and standard methods of encoding contextual features were developed. This analysis was described in section 5.7. To evaluate RoadCast on the test data, the developed standard encoding methods

would be used to acquire the contextual features in RoadCast. A list of the contextual features used can be found in table 5.2.

A caveat of this study was that the contextual data was collected after the contexts took place (schedules of contexts from 16$^{\text{th}}$ March 2016 were not available). If RoadCast were to make forecasts a year into the future, it would need to use schedules of these contexts. Many of these schedules would not change, such as public holidays and New Year's Eve, but some may change, such as rescheduled football matches. If contexts were rescheduled, RoadCast could account for this by re-making its forecasts with updated contextual features, albeit at a shorter forecasting horizon.

6.3 Performance metric

In order to evaluate the accuracy of the presented algorithm, the Mean Squared Error (MSE) was used as the performance metric. MSE was chosen over mean absolute error because it gives a relatively higher weight to large errors. This was thought to be particularly important for the application of incident detection, because predictors would perform better if they could account for large traffic disturbances caused by contexts (which could cause false alerts). For a detector d, MSE is defined as:

$$MSE(d) = \frac{1}{N}\sum_{j=1}^{N}\left(\tilde{z}(d,t_j) - z(d,t_j)\right)^2 \tag{6.1}$$

where N is the number of messages in the test dataset, $\tilde{z}$ is the predicted value at time t_j, and z is the true value.

6.4 Historical average predictor

During the development of RoadCast, the historical average was found to be one of the most accurate and simple predictors. Many forms of historical average were compared using preliminary tests on the training data, including using the mean, median and mode of different combinations of messages' 'time of day', 'day of week' and 'month'. Figure 6.1 shows a comparison of the cross-validation score of different type of average within historical average predictors. A historical average based on different context values, as well as the time of day and day of the week, was also tested. The most accurate form of historical average found was to take the mean of subsets (of the training data) corresponding to each combination of 'time of day' and 'day of week'. That is, to forecast next Monday as the mean of all Mondays in the training data. Periods that had no messages in the training data were predicted as the same day, previous time period.

Figure 6.2 shows that RoadCast's forecasts appeared similar to the historical average when not using contexts (the lines overlap), i.e. when RoadCast only used 'time of day' and 'day of week' features. This is because RoadCast without context is effectively a form of historical average. Each tree in the forest would create subsets of the training data based on different combinations of 'time of day' and 'day of week'. However, as can be seen in figure 6.2, both predictors had

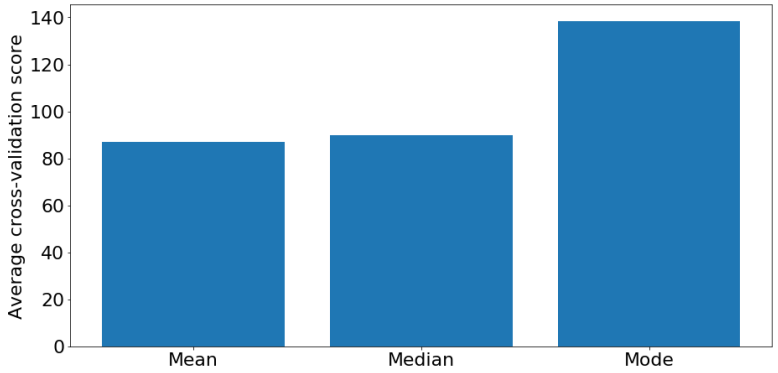


FIGURE 6.1: Cross-validation score of different types of average used within historical average predictors.

large errors on this day because New Year's day's traffic conditions were very different than an average Sunday. The final RoadCast algorithm (blue line) created a more accurate forecast by incorporating contextual data.

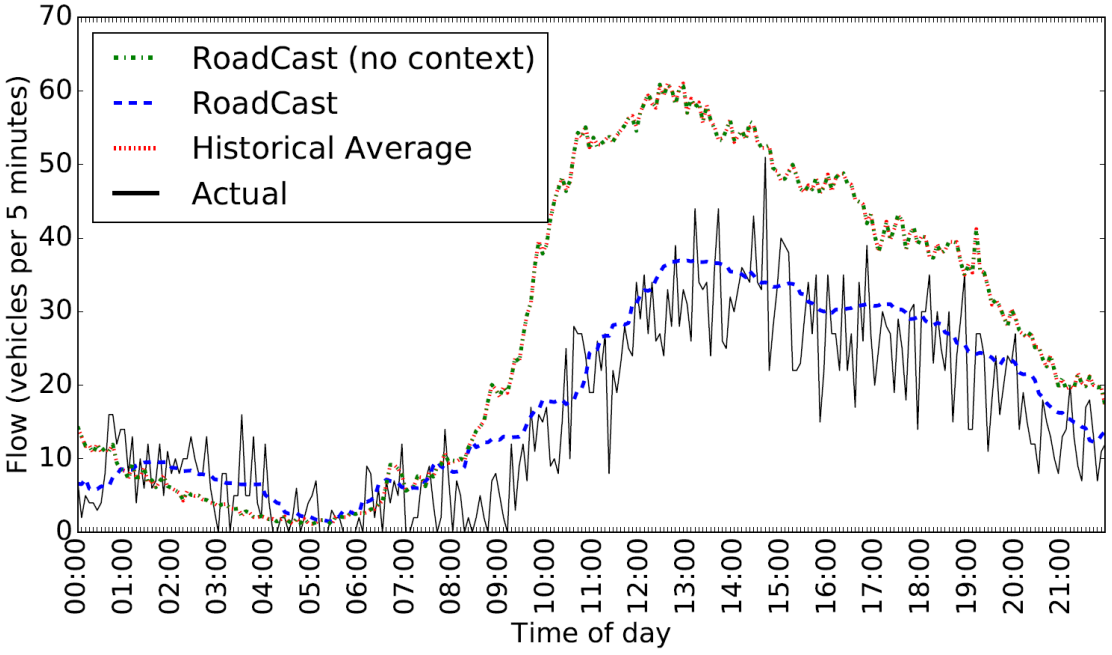


FIGURE 6.2: Flow forecast for Sunday, New Year's Day 2017, at detector B. RoadCast (without context) used 'time of day' and 'day of week' features only.

Because of the commonality of use throughout the literature, and its performance in preliminary tests, this form of historical average was chosen for comparison with RoadCast. It should be noted that there do exist many other traffic forecasting algorithms in the literature that were considered for comparison with RoadCast. Unfortunately, many of these algorithms were not suitable for the exact problem at hand, or could not be implemented from the details provided in the literature.

Comparisons with other types of machine learning and statistical algorithm were also considered. The decision to use a random forest in RoadCast was made after preliminary analysis on the dataset in Southampton. After the decision was made, a substantial amount of research was put into developing a random forest algorithm that was best suited to this problem. As such, a comparison of the final RoadCast algorithm with other out of the box machine learning and statistical algorithms would be unfair. As such, RoadCast would only be compared with a historical average predictor. If RoadCast could be found to be more accurate than the historical average predictor, it would be judged to be sufficiently accurate for the proposed IDA in order to answer the research hypothesis.

6.5 Initial offline test

Using the methodology described in section 5.6 and features described in table 5.2, RoadCast was applied to Southampton's historical traffic and context datasets (described in sections 5.3 and 6.2). This 'offline' test (i.e. test on historical data) was undertaken to assess RoadCast's suitability for the proposed incident detection algorithm. That is, to assess RoadCast's ability to use contextual data to improve its forecasts, while meeting the requirements stated in section 5.2.

6.5.1 Methodology

An overview of the methodology is as follows. Firstly, the selection algorithm was used on each detector and target variable combination to train RoadCast on the first year of study data (16^{th} March 2015 to 15^{th} March 2016). Then, RoadCast was tested on the second year of data (16^{th} March 2016 to 15^{th} March 2017). Hence, the forecast horizon varied between five minutes and one year.

During training, the predictors would have access to both the input features and the target variable, and so could use this data to discover patterns of the target variable with respect to the input features. The testing dataset would then be used to evaluate the performance of the algorithm. During testing, the algorithm has to predict the target variable using only the input features.

Of all the contexts found to influence Southampton's conditions, the least frequent occurred once per year (i.e. did not have values of 10 for a year continuously). As such, one year's worth of data was seen as sufficient and necessary for both training and testing the algorithm on each context. As only two years of data was available, longer training and testing periods were not available in this initial study. Hence, one year of data would be used for training, and one year for testing.

Clearly the horizon of forecasts required for incident detection would not need to be one year. However, the forecast required would have to be of expected conditions, i.e. it should not be able

to forecast the disruption caused by incidents. By developing an algorithm that could forecast at such a horizon, it would be clear that it was forecasting expected conditions, because incidents are not known of ahead of time, and rarely last longer than a few hours.

This test was devised such that the predictors would meet the forecasting algorithm requirements stated in section 5.2. That is, target variable suitable for incident detection was chosen, and a horizon of up to one year was set so that expected traffic conditions would be forecasted. RoadCast was also tested with only time of day and day of week features, so that the improvement RoadCast gained by incorporating contextual data could be assessed. These three factors would ensure that the results of this test could be used to assess RoadCast's suitability for the proposed IDA methodology.

Typically, when a traffic forecast of over an hour ahead is required, a form of historical average is used. Similarly, historical averages have also been used in IDAs which compare real-time observations to forecasts of expected conditions (Chrobok et al., 2000, Syrjarinne, 2016). As such, a historical average predictor was used as a benchmark for which to compare RoadCast. If RoadCast could produce more accurate forecasts than a historical average, particularly during contexts, RoadCast would be seen to have met its aim, i.e. it would be suitable for the proposed IDA methodology.

The following sections evaluate the performance of RoadCast on the test data, and make comparisons to the historical average predictor.

6.5.2 Results

Over all detectors, RoadCast's flow forecasts had an average MSE of 80.7 vehicles squared, compared to the historical average's 84.4, a 4.4% improvement. For average speed, RoadCast had an average MSE of 16.34 miles per hour (mph) squared, compared to the historical average's 17.02, a 4.0% improvement. RoadCast was more accurate than the historical average on 93% and 98% of detectors when forecasting flow and average speed respectively. RoadCast (with context)'s accuracy improved on RoadCast (without context) from 84.5 MSE to 81.6 MSE for flow (a 3.4% improvement), and 16.56 MSE to 16.54 MSE for average speed (a 0.1% improvement).

To assess the significance of the results achieved, paired t-tests were conducted on the predictors' squared errors, using the Python library SciPy (SciPy, 2018). A significance level of 5% was chosen. There is a statistically significant difference between the accuracy of RoadCast and the historical average when forecasting both average speed and flow, with p-values of 0.000 in both cases. There is also a statistically significant difference between RoadCast with and without contexts when forecasting flow and average speed, with p-values of 0.000 and $6.3 * 10^{-86}$ respectively.

The smaller difference in accuracy between RoadCast with and without context when forecasting average speed is thought to be because the contexts influenced flows more directly than average

speeds. Every context was seen to cause flow variation by altering travel demand. However, this would only lead to average speed variation when congestion was caused by flows exceeding capacity. On inspection, very few detectors appeared to have contexts that caused congestion, and those that did only caused congestion for a very short period of time. Variation in average speeds may also have come from affected road conditions, such as rainfall limiting road capacity, but this variation was found to be negligible in Southampton.

Figure 6.3 shows the improvement of RoadCast over the historical average at 110 of the detectors used. One detector was 8% more accurate than the historical average over flow, but 23% less accurate over average speed, and so was omitted from the figure. This anomaly could be attributed to RoadCast training on unrepresentative data, e.g. temporarily faulty detectors, roadworks etc. When unrepresentative traffic conditions occurred during rarely occurring contexts, RoadCast could incorrectly 'learn' to predict such conditions in the same contexts in the test data. In this case, roadworks during Easter 2015 caused severe congestion on the omitted detector. This resulted in RoadCast predicting very low average speeds during Easter 2016, creating large errors that were not produced by the historical average. It is thought that this issue would generally affect RoadCast more so than the historical average, because the error induced for the historical average would be spread throughout the year of testing data, whereas the error induced for RoadCast would be concentrated at Easter, which would contribute to a higher MSE overall.

This phenomenon could also be seen at other detectors, but with a lesser detrimental effect because the conditions during the context deviated less from average conditions, for example during an accident that disrupted traffic nearby a detector. Another likely reason for this phenomenon leading to less accurate forecasts is because fewer representative messages can be averaged to form each prediction (representative messages may be left out), meaning variation in the representative data is less accounted for. To limit this phenomenon, it may be possible to remove unrepresentative training data, but this would be a challenging task to achieve in a practical way. Comparing training data to incident reports manually to identify disrupted periods would be very labour intensive. Alternatively, the time period of incident logs that are within a certain distance to a detector could be automatically removed from the traffic dataset, but this method may be unreliable as representative data could easily be removed. Automatically identifying unrepresentative data by analysing traffic data and comparing to contexts may be possible by using an off-line version of the proposed IDA on the training data. If RCID were to be used, it would require a cross-validation approach, which would increase RCID's calibration time by a factor approximately equal to the number of folds minus one. After investigating methods to remove unrepresentative data, no method was found to be suitable enough to provide a sufficient benefit to RoadCast, hence no method was used. It should be noted that this problem of removing unrepresentative training data is faced by many road traffic forecasting algorithms, and a general solution would benefit the field. As such, this problem would be a useful area for future research.

Without context, RoadCast forecasted more accurately than the historical average in terms of flow and average speed. Accuracy improvements came from averaging over similar time of day and day of week periods (e.g. Sunday 23:35-23:55), which reduced variance in the predictions.

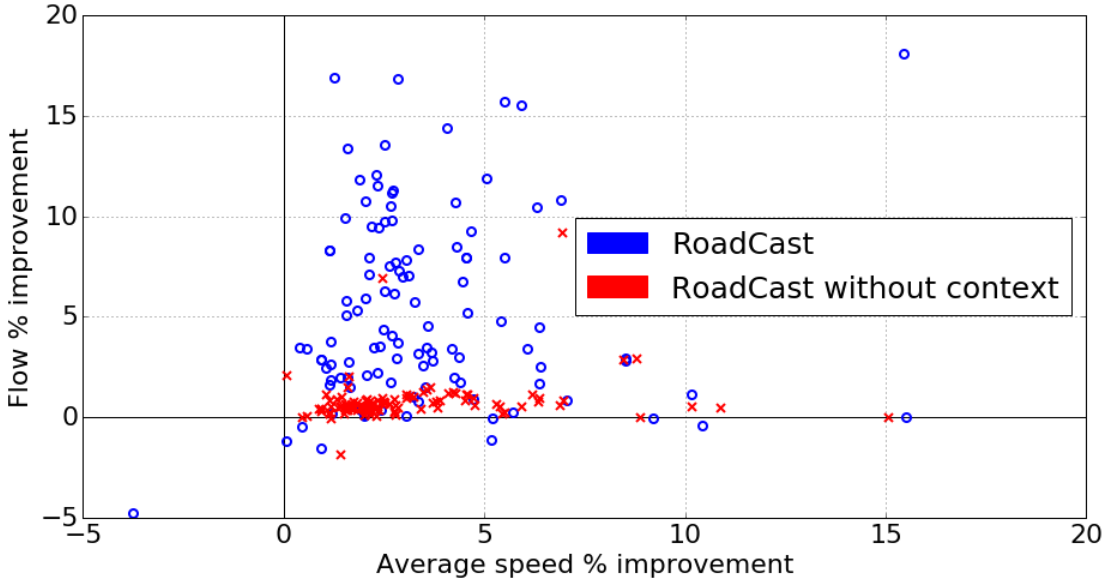
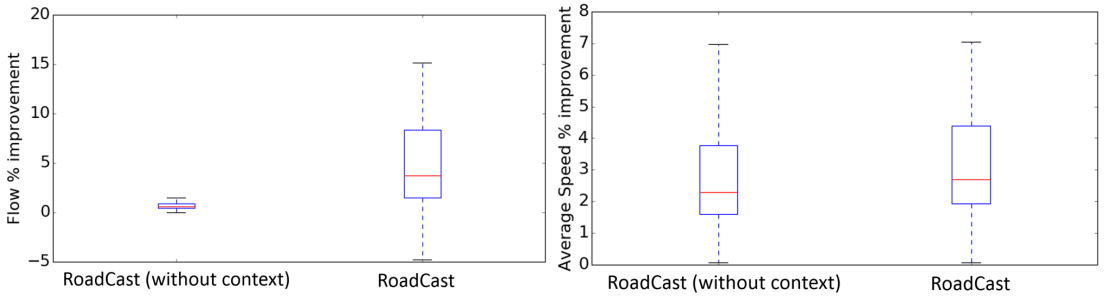


FIGURE 6.3: RoadCast's percentage improvement over the historical average in MSE at each detector.

Further accuracy improvements came when contexts were incorporated as RoadCast could account for the contexts' variation by 'learning' from previous occurrences in the training data.

As can be seen in figure 6.4, the distribution of the percentage improvement in flow was much wider when RoadCast used context. The interquartile range for flow was 0.2 with contexts, and 6.8 without contexts. This was because the contexts had different effects on different detectors. Some detectors were not affected by contexts, but others were affected by many contexts that caused visibly large amounts of variation.

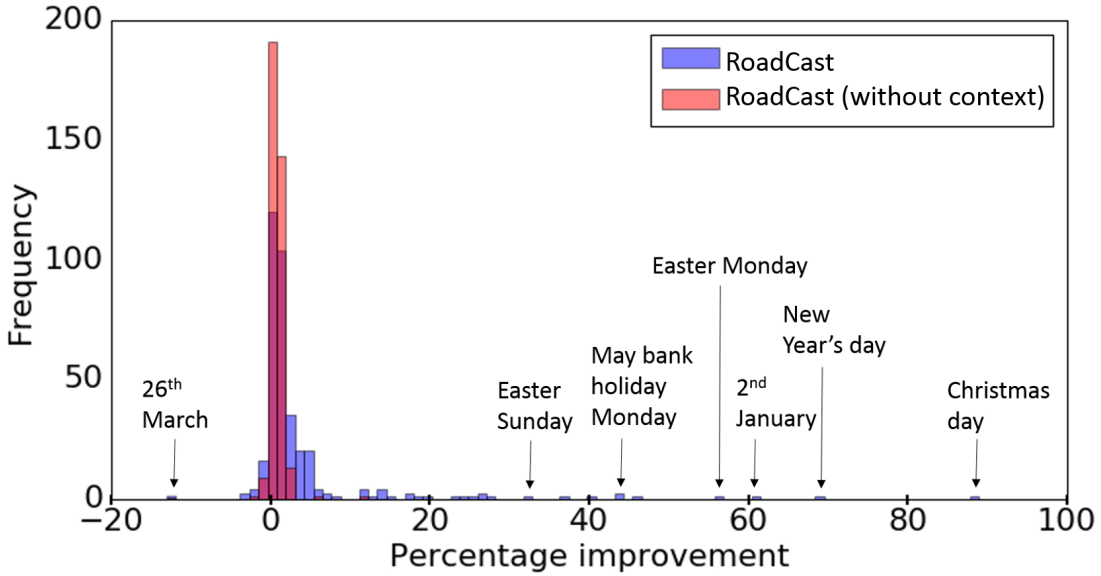


(A) Flow

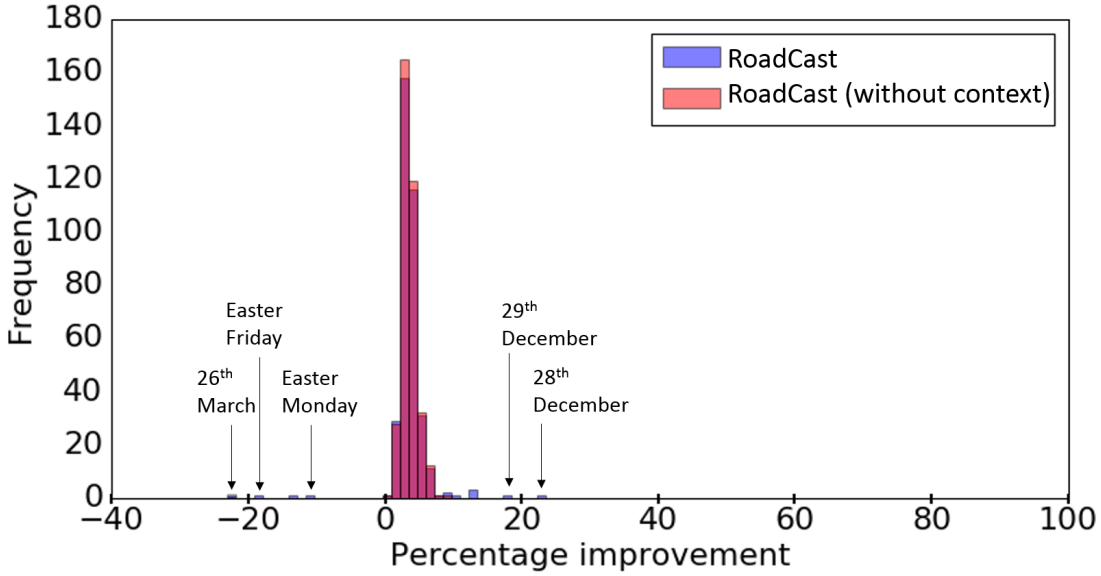
(B) Average speed

FIGURE 6.4: Box plot of RoadCast's MSE percentage improvement over the historical average.

Figure 6.5 shows the percentage improvement (in terms of MSE) of RoadCast's forecasts over the historical average over all detectors, broken down by each day. RoadCast's improvement with and without context are compared. The purple areas are where the predictors' bars overlap. Figure 6.5a shows that for flow, RoadCast made a small improvement on most days, which came from a reduction in variance (as discussed earlier), and a large improvement on few days where contexts caused variation.



(A) Flow



(B) Average speed

FIGURE 6.5: Histogram of RoadCast's percentage improvement over the historical average each day, averaged over all detectors, in terms of MSE.

When predicting flow, RoadCast (with context) was more accurate than the historical average on 89.1% of the days. Without context, the largest improvement was 12% and largest loss was -1.8%, but with context the largest improvement was 88% and the largest loss was -12%. The days with the largest improvements were on holidays. When using context, RoadCast had a larger spread of percentage improvements and losses, but reduced its error overall.

When predicting average speed, RoadCast (with context) was more accurate than the historical average on 98.9% of the days. Without context, the largest improvement was 9.2% and least improvement was 0.9%, but with context the largest improvement was 23.9% and the least improvement was -21.6%. Again, the days with the largest improvements (and losses) were on

holidays.

As can be seen in figure 6.5b, RoadCast forecasted average speed more accurately than flow on more days than for average speed, but to a lesser magnitude on average. During Easter, RoadCast forecasted average speed less accurately than the historical average, and flow less accurately on 26th March. On investigation, this appeared to be caused by roadworks disrupting traffic at one detector in the south west of the city, during the Easter holiday in 2015. This detector was the detector omitted from figure 6.3, as described at the start of this section. RoadCast 'learnt' from the Easter context that flows on Good Friday differed from typical Fridays, and hence forecasted 2016's Easter values similarly to 2015's Easter. Because there was only one Easter in the training data, and roadwork disruption happened to occur during this context, larger errors occurred in RoadCast than the historical average on this Good Friday.

This section has shown that RoadCast is able to improve its forecasts by 'learning' the impact of contextual features on traffic conditions. It is cannot be concluded that RoadCast accounted for all variation from contexts in its forecasts because it is difficult to quantify the errors caused by failing to account for contexts, and errors from other types of variation, such as noise. Noise are unpredictable variations created during the collection and pre-processing of data. For example, a source of noise in this case would be the method of estimating the average speed of vehicles (Cherrett et al., 2001). Despite this, RoadCast has shown a clear improvement by incorporating contexts, and was consistently more accurate than the benchmark historical average predictor.

6.5.3 Feature discussion

In this section, each contextual feature included in RoadCast is discussed. By comparing RoadCast's forecasts to the historical average predictor, the improvements made by incorporating contextual data are demonstrated.

6.5.3.1 Time

Unsurprisingly, the most important features for forecasting flow and average speed were found to be 'time of day' and 'day of week'. It could be seen that each detector had distinct flow and average speed patterns, characterised by the time of day and the day of the week. Detectors on arterial routes into and out of the city centre typically had peak flows on weekday mornings and evenings respectively. Detectors by the shopping district also exhibited morning and evening peaks, but the highest flows occurred during the middle of the day, particularly on weekends.

6.5.3.2 Holidays

Despite holiday features occurring rarely (only one Christmas and Easter in the training data), it was found that the traffic characteristics could be 'learnt' well enough to create large improvements. In fact, the largest improvements in RoadCast came from incorporating holiday features, because traffic on holidays was so different than on working days. Typically, holidays did not

have AM or PM peaks, and had less overall flow and congestion than average weekdays (see figure 6.6). Typically, RoadCast's forecast of holidays appeared to be an average of messages at similar times at the same point of the holiday in the training data, for example, the same time on Christmas Eve, or the Monday after Easter. Interestingly, forecasts of public holiday Fridays appeared similar to typical Saturdays, and forecasts of public holiday Mondays appeared similar to typical Sundays. The improvements made highlight the importance of forecasting algorithm's ability to understand the patterns of commuter travel behaviour.

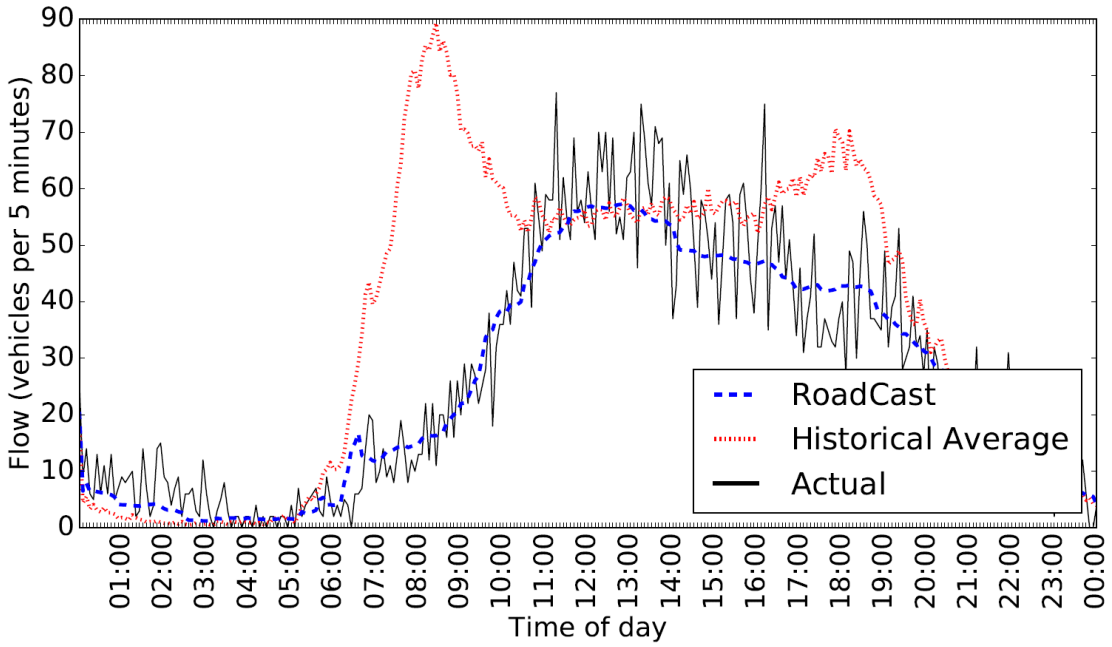
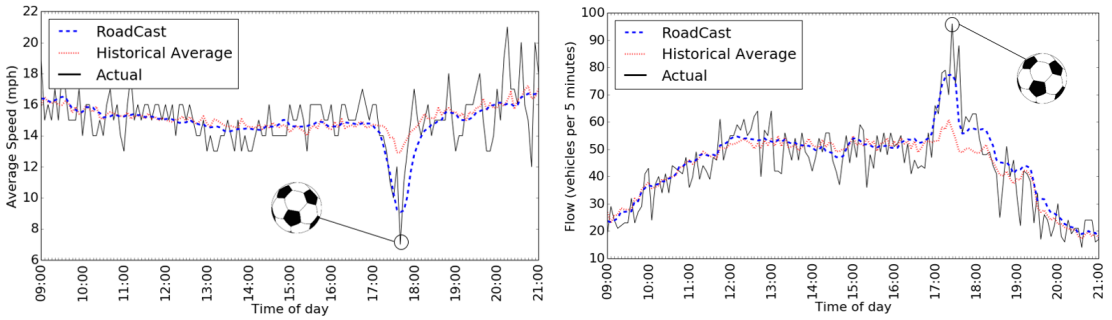


FIGURE 6.6: Flow forecast for public holiday Monday, 2nd May 2016, at detector B.

6.5.3.3 Events

Incorporating event features created smaller improvements in RoadCast's overall accuracy than holidays, but such improvements could be seen clearly in the forecast and traffic datasets. These improvements were seen to be important because many ITS applications require a forecast of the expected disruption caused by events (e.g. route guidance, traffic control).

Football matches were found to be the most predictable event type because of their significant, consistent effect on traffic conditions. As can be seen in figure 6.7, 2.5 hours after kick-off, detector C's average speed dropped to seven miles per hour, and flow peaked at around 95 vehicles (per 5 minutes). This was because detector C was on a main arterial route used by football fans leaving the city after matches. RoadCast predicted this well by 'learning' from consistently similar football match disruptions in the training data. Because matches are held on some but not all Saturdays at 15:00, the historical average predicted a small increase in flow, and drop in average speed. This predictor cannot account for football context, and so forecasted traffic worse on both match and non-match days.



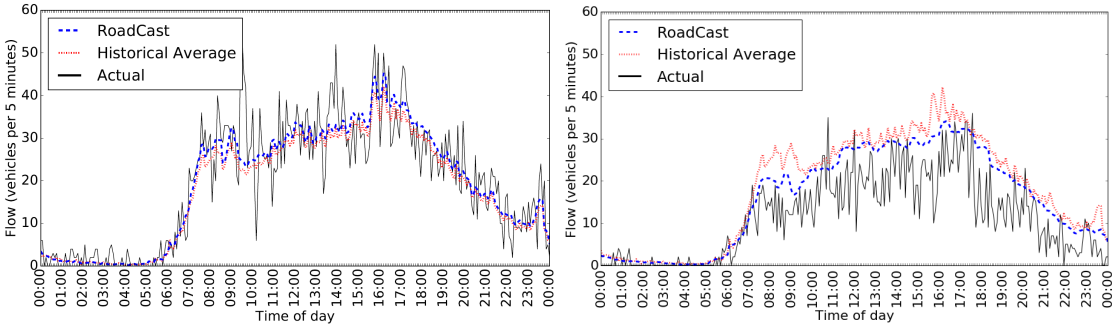
(A) Average speed forecast (B) Flow forecast

FIGURE 6.7: Traffic forecast for Saturday, 9th April 2016, at detector C. Premier League football match against Newcastle kicked off at 15:00 at St Mary's Stadium.

Many other event contexts had similar effects on traffic conditions to football matches. That is, sudden increases in flow, which would occasionally lead to congestion. These other events were each found to improve forecasts when included, but were more rare and typically caused less disruption than football matches. The charity event and music festival contexts were not included in any detectors by the selection algorithm, indicating that their effect on traffic conditions at the study's detectors was minor.

6.5.3.4 Education

Detectors nearby schools achieved more accurate flow forecasts with the addition of the 'school' feature. Figure 6.8 shows how lower flows were typically forecasted (and observed) during school holidays, particularly around school opening and closing times (the historical average's forecast remained the same). However, the context did not fully account for the context's variation at some detectors, as can be seen in figure 6.8b for detector A. The reduction in travel demand during school holiday days was found to vary from day to day, but the cause of this variation was not clear. The most likely explanation was that higher demand was found on days with unseasonably warm weather. This explanation was not investigated further because it was decided that RoadCast would not include weather contexts, as was stated in section 5.7.4.



(A) School term (B) School holiday

FIGURE 6.8: Traffic forecast for a typical Friday during school term (18th March 2016) and school holiday (19th August 2016), at detector A.

6.6 Sensitivity to the quantity of training data

In the offline test of RoadCast (see section 6.5.2), RoadCast was first tested with a year of training data so that all contexts could be learnt from in training, and evaluated in testing. However, if RoadCast required one year of data to train, its use in the real-world may be limited. If accurate traffic forecasts were needed within a month of data being collected for a particular location, it is unclear whether RoadCast would be suitable. As such, this section evaluates RoadCast's forecast accuracy when using different amounts of training data.

6.6.1 Test methodology

The test devised would assess RoadCast's ability to forecast traffic conditions when using different amounts of training data. The test would always use the second year of data for testing (16^{th} March 2016 onwards), but various length periods for training. Training periods would be of the last week, two weeks, month, two months, four months, eight months and one year before the testing period start date.

For short training periods, the time period covered is clearly important. If the predictors were trained only on a week during the Christmas holiday, inaccurate predictions of the following year could be expected. The weeks leading up to 16^{th} March 2016 could be described as relatively typical because no public holiday occurred since Christmas, and the most recent football match was on the 5^{th} March. Because only two years of data was available, with a midpoint at 16^{th} March 2016, only this date was suitable to be the start of the testing period.

A year of testing data was used so that RoadCast's ability to forecast every context could be assessed. The same year was used each time so that the results of each test would be directly comparable. The training period was taken as the period before the test start date (rather than 16^{th} March 2015 onwards) as this would be most similar to a real-world implementation of RoadCast. That is, if a certain amount of training data had been collected to date at a particular location, an idea of the accuracy of RoadCast's forecasts would be gained.

6.6.2 Flow - overall

Figure 6.9 shows each predictor's mean squared error, averaged over all detectors, when forecasting flow while using different amounts of training data.

As expected, when each predictor had more training data, forecasts became more accurate, and so the mean squared error reduced. When using only a week of training data, RoadCast had a very similar accuracy with and without contextual data. This was because no event or holiday contexts occurred in the week between 9^{th} and 16^{th} March 2016, and so the selection algorithm rarely included contexts. However, the historical average predictor was less accurate by 44 MSE. As illustrated in figure 6.10, this was because it would simply use the message with the same time

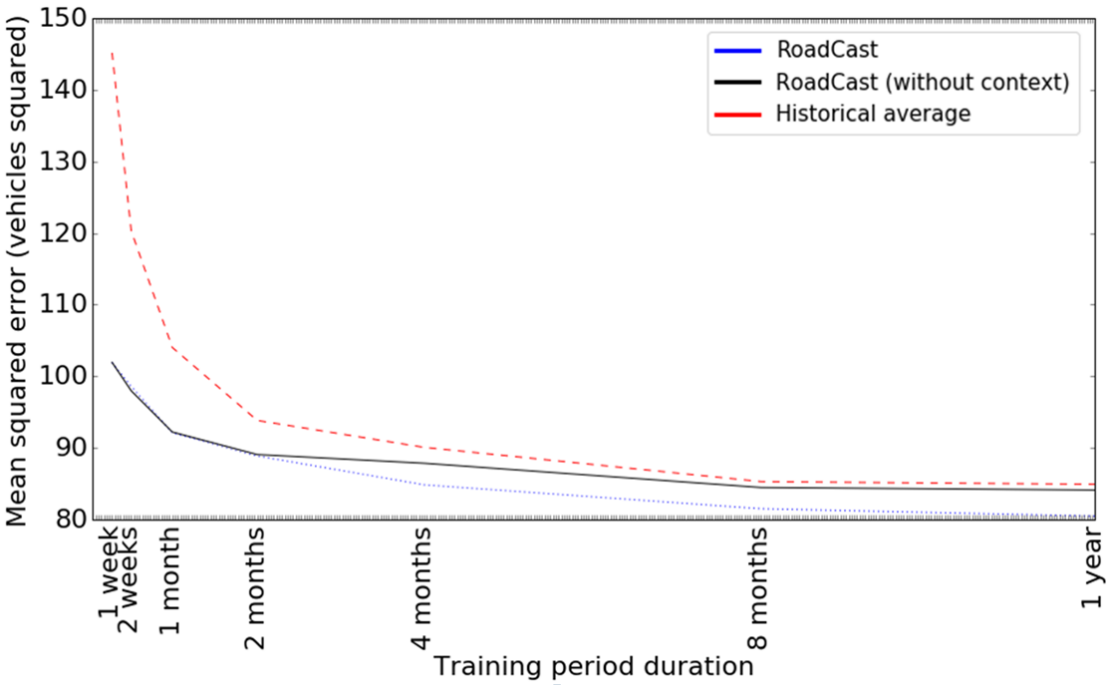


FIGURE 6.9: Different predictor's average mean squared error over all detectors when forecasting flow, using different amounts of training data.

and day of the week as in the training data, whereas RoadCast would forecast with less variance by averaging over messages of similar times and days of the week, minimising the error from the noise in the data. This resulted in a far lower mean squared error because each prediction's absolute error would be relatively low, whereas the historical average would occasionally have a large absolute error, which would increase mean squared error by a large amount (this can be seen in figure 6.10).

When more training data was used, forecast errors decreased, and the historical average's error became closer to RoadCast without context. With one year of data, the difference between the predictors was only 3.6 MSE. The convergence happened because the historical average's forecast averaged over more values, and so reduced the variance from noise. RoadCast without context remained more accurate because it could average over similar time periods and days of the week, but to a lesser extent because of the larger amounts of data available.

From around two months of training data onwards, RoadCast showed improvement by incorporating contexts. At two months the improvement was 0.2 MSE, which came mainly from the use of the football context, because it was the only event or holiday type context that had occurred in the training and testing period (as found in chapter 5, events and holidays had most influenceon forecast accuracies). An improvement of 3 MSE was found at four months because the training period included Christmas (which had very different flows to other days), meaning forecasts of Christmas 2017 were much more accurate. Other day's forecasts also improved slightly, as they would now be an average of messages that were not during Christmas, meaning the forecast wouldn't be skewed by Christmas's typically low flows. At eight months the improvement was still 3 MSE. Here, public holiday, cricket matches and boat show contexts were also included

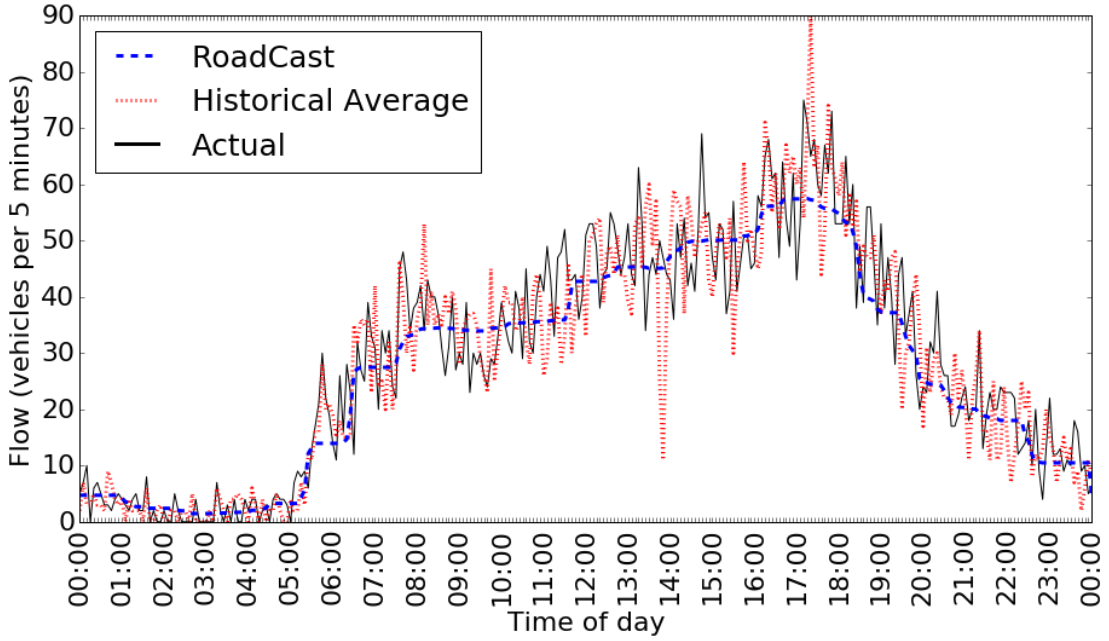


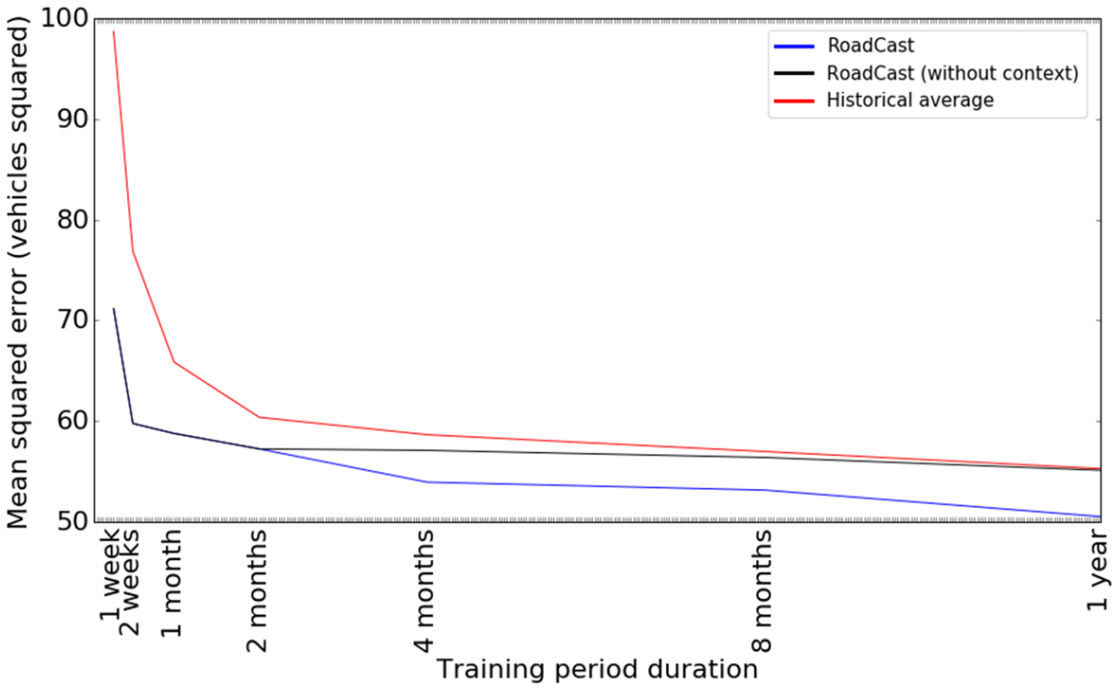
FIGURE 6.10: Forecast of detector C's flow on the 17th March 2016, after using one week of training data.

in the training data. An improvement of 4 MSE was found at one year, due to Easter, charity event, half marathon and music festival contexts also being included. With more training data, RoadCast continued to improve by incorporating more contexts.

If a longer time period of training data were to be used, it is unclear how forecast accuracies would change. Accuracy improvements may come from reducing variance further by averaging over more training data, but accuracies may worsen if this data is unrepresentative. For example, long term changes, such as population increase, could alter traffic patterns, and so cause all predictors to become less accurate. Accuracies of RoadCast (without context) and the historical average would be expected to converge, because with more data, RoadCast (without context) would be more likely to create leaf nodes that simply hold messages of one particular time of day and day of week value. RoadCast (with context) would be expected to continue to improve relative to the other predictors, by learning from more occurrences of rarely occurring contexts, such as Christmas.

6.6.3 Flow - single detector

In this section, the same analysis is undertaken on a single detector (rather than an average of all detectors), in order to gain perspective on how incorporating contexts improved forecasts. Figure 6.11 shows that detector C followed similar trends to the average of all detectors. However, an understanding of the impact made by contexts can be gained from observing which contexts were included by the selection algorithm with each amount of training data, as shown in figure 6.11b.



(A) Different predictor's mean squared error when forecasting flow at detector C, using different amounts of training data.

Training duration (start date)	1 week (09-03-16)	2 weeks (02-03-16)	1 month (16-02-16)	2 months (16-01-16)	4 months (16-11-15)	8 months (16-07-15)	1 year (16-03-15)
Contexts					Football	Football	Football
					Christmas	Christmas	Christmas
							Easter

(B) Contexts included after the RoadCast selection algorithm when using different amounts of training data at detector C.

FIGURE 6.11: Predictor's contexts used and mean squared errors when forecasting flow at detector C, using different amounts of training data.

Using up to two months of training data, the selection algorithm did not include any contexts, and so RoadCast produced an almost identical mean squared error to RoadCast without context (negligible variation due to different random numbers being used while retraining). With four and eight months of training data, RoadCast included the football and Christmas features, and hence improved its forecasts relative to RoadCast without context by 3.4 and 3.9 MSE respectively. With one year, the selection algorithm also included the Easter context, and so improved forecasts further relative to RoadCast without context, to a total difference of 8 MSE.

6.6.4 Average speed - overall

When forecasting average speed, the predictors followed a similar trend to forecasting flow but with some key differences. Firstly, in contrast to the historical average, RoadCast's error started low and improved gradually. This large difference of 12.9 MSE at one week of training data was caused by RoadCast's ability to reduce variance by averaging over similar times and days. This

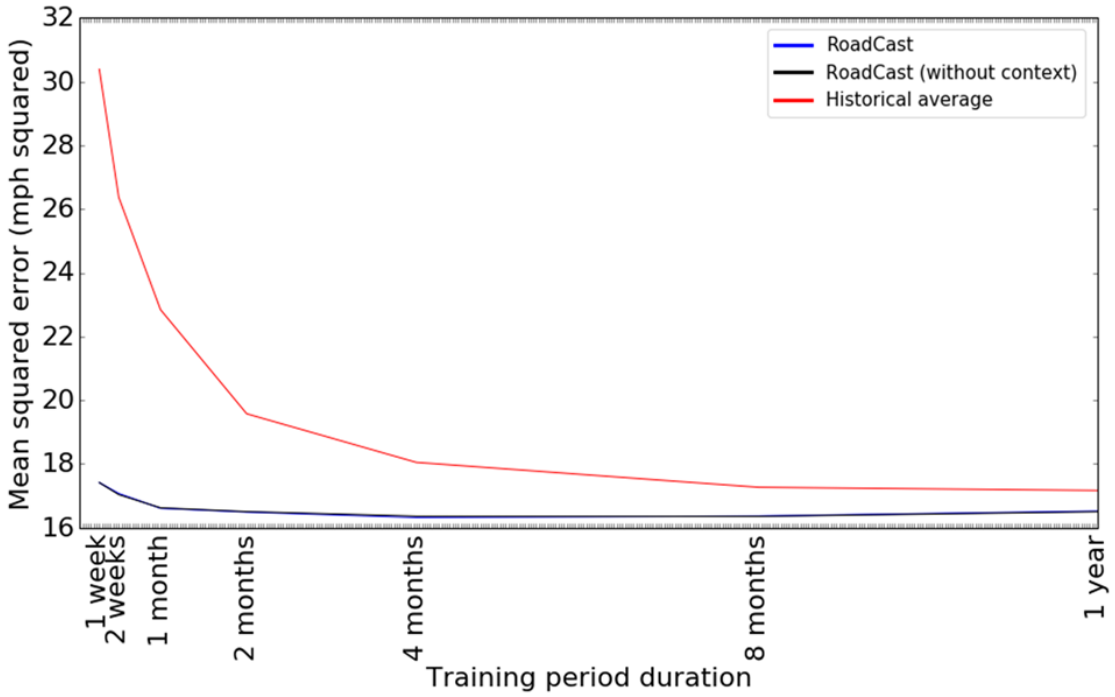


FIGURE 6.12: Different predictor's average mean squared error over all detectors when forecasting average speed, using different amounts of training data.

is particularly important when forecasting average speed, as the values remain fairly constant most of the time (during free-flow), but noise in the data causes a lot of variance. In the same way as when forecasting flow, the historical average is able to reduce its error with more training data, because it can reduce the variance by averaging over more data.

It can also be seen that when forecasting average speed, incorporating contexts did not reduce the overall mean squared error significantly. As could be seen in chapter 5, RoadCast was able to forecast the congestion caused by football matches, and correctly forecast that there would not be congestion during rush hour on public holidays, unlike the historical average. However, these abilities did not create a large improvement in the overall mean squared error as the contexts only disrupt the average speed rarely, and only at some detectors. Instead, the overall error comes mostly from noise during free flow conditions. Indeed, it is thought that a greater fraction of the average speed forecasting error is made up of noise than the flow forecasting error.

6.6.5 Forecasts of contexts

Overall it could be seen that RoadCast became more accurate with more training data, in part by incorporating contexts. But how much training data is required to forecast each context? This section investigates how RoadCast's forecasts of particular contexts varied with the amount of training data used.

Figure 6.13 shows RoadCast's forecast of the day of a football match when using different amounts of training data. The selection algorithm did not include the football feature when

using one week of training data, but did with every other amount. At 2pm, the actual value of flow was 103. The flow value of RoadCast's forecast can be seen changing from 61 to 108 vehicles as the amount of training data changed from one week to one year. With one week, RoadCast forecasted values similar to the previous Saturday, which did not account for the football match disruption because no matches occurred in the training data. But with more training data, and hence football matches, RoadCast's forecasts became more accurate.

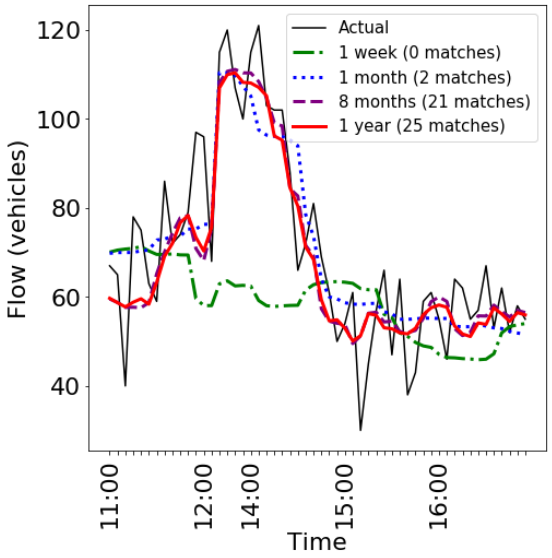


FIGURE 6.13: RoadCast's forecast for Saturday, 9th April 2016, at detector C, when using different amounts of training data. A Premier League football match against Newcastle kicked off at 15:00 at St Mary's Stadium.

Figure 6.14 shows how RoadCast's forecast of New Year's Day changed when using different amounts of training data. As expected, the selection algorithm did not include the Christmas or New Year's Eve features when they did not occur in the training data (one week, one month and two months), but did include them when they were in the training period (four months, eight months and one year). As can be seen in figure 6.14, RoadCast was visibly more accurate throughout the day when it could learn from the Christmas context during training. The MSE of the day's flow values for RoadCast with one week, four months and one year was 284.7, 37.2 and 31.2 respectively.

It is apparent from the figures above that the improvements gained by incorporating contexts come from 'learning' from many previous occurrences in the training data. That is, the accuracy of forecasting particular contexts appears most dependent on the number of representative occurrences of the context in the training data, rather than the shear amount of training data used.

6.6.6 Discussion

RoadCast was found to be more accurate with more training data, and consistently more accurate than the historical average. Improvements gained by incorporating contexts could be seen from two months onwards (because a number of football matches occurred in this period). Ideally, RoadCast would use one year of training data so that it could 'learn' from every context

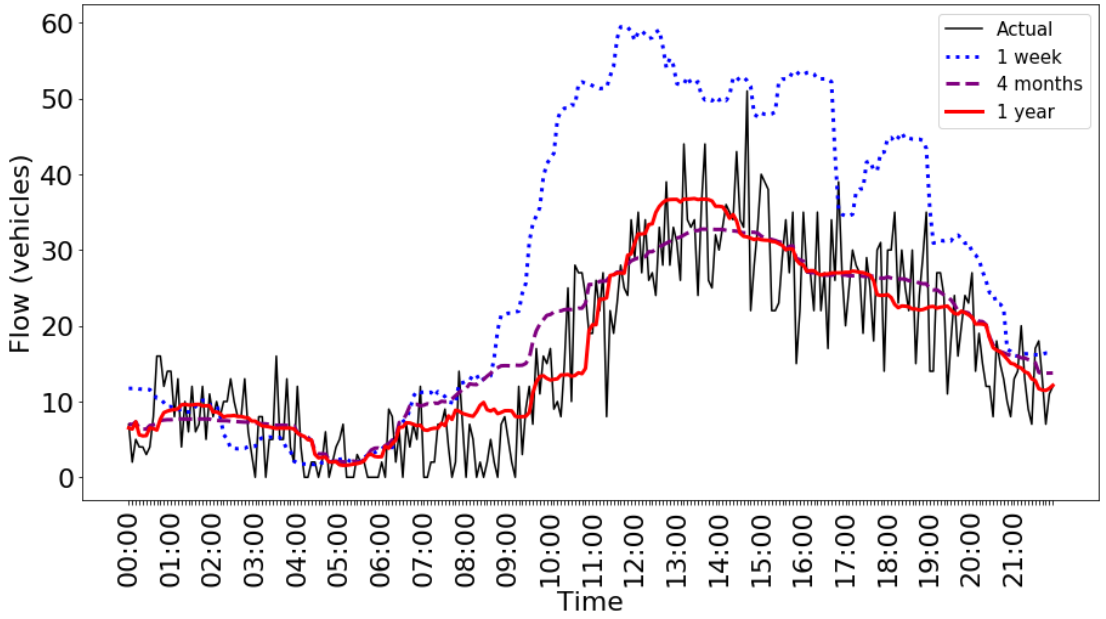


FIGURE 6.14: Flow forecast for Sunday, New Year's Day 2017, at detector B, using different amounts of training data. RoadCast (without context) used 'time of day' and 'day of week' features only.

that was included in the Southampton dataset. But if this was not possible, RoadCast would still forecast more accurately than the historical average predictor by using averaging to produce forecasts with reduced variance.

Similarly, RoadCast's forecasts of contexts were seen to improve with more training data. But rather than the amount of data, the number of occurrences of contexts in the training data appeared to influence RoadCast's accuracy most. The number of occurrences required for an accurate forecast depends on many factors, such as the context and detector being used, and the level of accuracy required. But using football forecasts at detector C as an example, figure 6.13 found that an improvement of 37 vehicles absolute error could be made when using only two context occurrences in training.

Practically, this result can be used to provide recommendations as to how much training data RoadCast needs to be effective in practice. Clearly, this decision will depend on the application in which the traffic forecasting algorithm is used in. Figures 6.9 and 6.12 shows that RoadCast should be preferred to a historical average predictor when any amount of training data is used, even when less than a week of training data is available. For applications for which accuracy is imperative, and there is less necessity for the algorithm to be used when little training data is available, it is recommended that RoadCast is implemented when at least one month of training data is available.

6.7 Sensitivity to the forecast horizon

This test aimed to discover whether RoadCast's forecast accuracy would degrade when using a greater forecasting horizon. If it did significantly, regular re-training of the algorithm would be advised so that forecasts would be based on representative data.

It was clear that RoadCast's (and the historical average's) forecast would not differ greatly when forecasting with different horizons over small time periods, e.g. 5, 10, 15 minutes. This is because both algorithms are not based on recent observations, but on the patterns found in large periods of historical data. However, when forecasting many months into the future, long-term variations in traffic conditions (such as population growth or building construction) could cause lower forecasting accuracies. As such, RoadCast was tested over a variety of long-term forecasting horizons. This would help to understand whether RoadCast's accuracy degrades at larger horizons due to long-term trends in traffic conditions.

6.7.1 Test methodology

The test devised to evaluate RoadCast's forecasting horizon was to repeatedly test on the last month's data (from 15th February 2017), using a year's worth of training data from different periods before the test set. That is, a year's data that ended one day, one week, one month, two months, four months and eleven months before the testing period. It should be noted that five detectors which had no messages during the one week testing period, due to missing or erroneous data, were not included in the results.

6.7.2 Flow

It can be seen in figure 6.15 that as the forecast horizon increased, each predictor became less accurate, but RoadCast remained the most accurate throughout. RoadCast's MSE went from 68.9 to 72.3 as the forecast horizon increased from one day to 11 months. Each detector followed a similar trend in terms of the change of the absolute value of the error and the rate of change of error as the forecast horizon increased. The observed increases in errors as the forecast horizon increased are most likely caused by long term changes in traffic conditions, such as population increase or travel mode change.

Although errors were largest at the greatest forecast horizon, figure 6.16 shows how the forecasts achieved still follow the actual travel conditions closely, and can still incorporate contexts to achieve more accurate forecasts. This suggests that the long term variation in Southampton's traffic conditions is minor, and that RoadCast was resilient to this variation. However, it is expected that if the road network changed significantly, e.g. a new road was built near to the detector, forecasts would become inaccurate and so retraining would be required. Similarly, significant changes to contexts (such as football team changing stadium) would result in forecasts of such contexts becoming inaccurate.

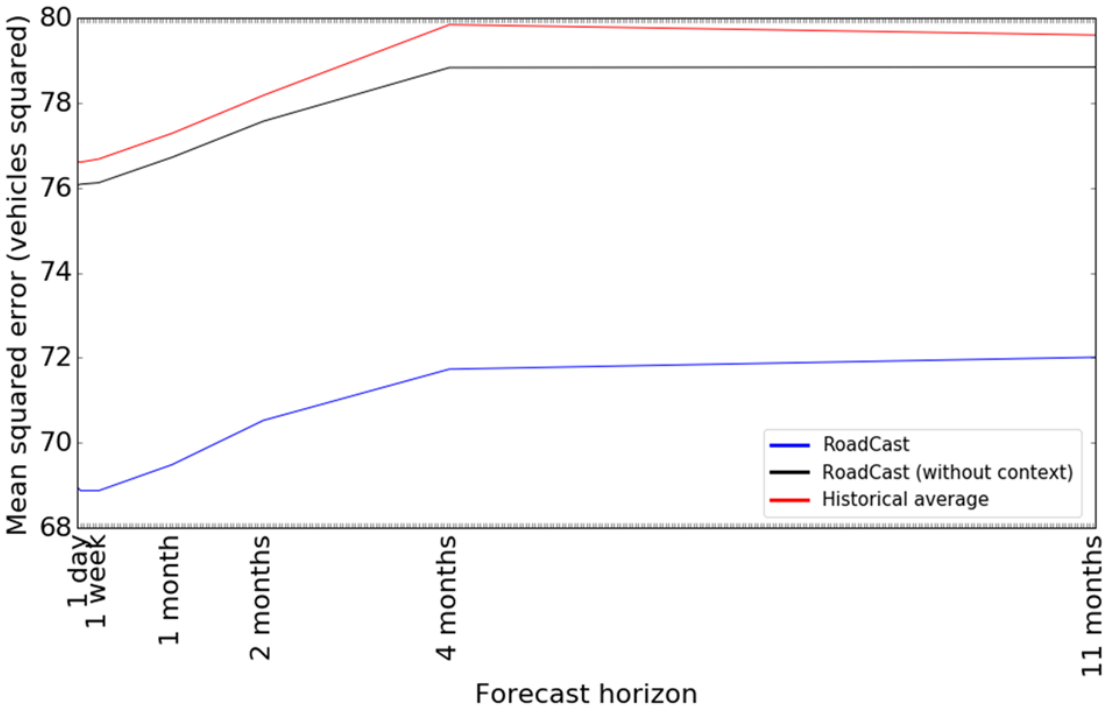


FIGURE 6.15: Different predictor's average mean squared error over all detectors when forecasting flow at different horizons.

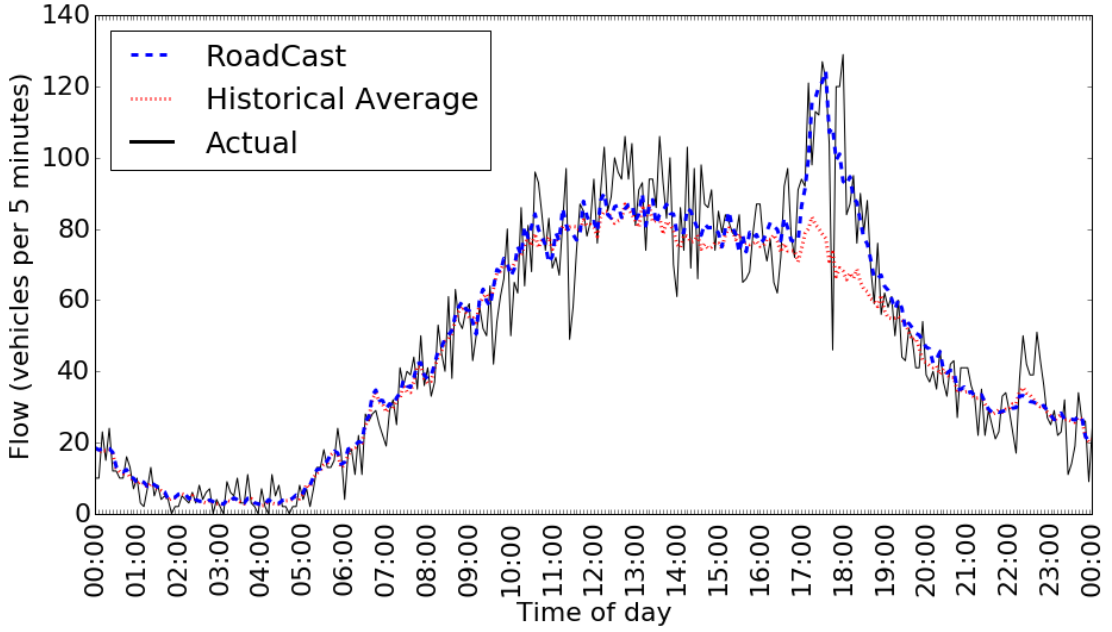


FIGURE 6.16: Flow forecast at a one year horizon, for Saturday, 4th February 2017, at detector A. Premier League football match against West Ham kicked off at 15:00 at St Mary's Stadium.

6.7.3 Average speed

As can be seen in figure 6.17, each of the predictor's average speed errors also increased, and at a similar rate of increase, at larger horizons for each predictor. RoadCast's MSE went from 16.6 to 18.3 as the forecast horizon increased from one day to 11 months.

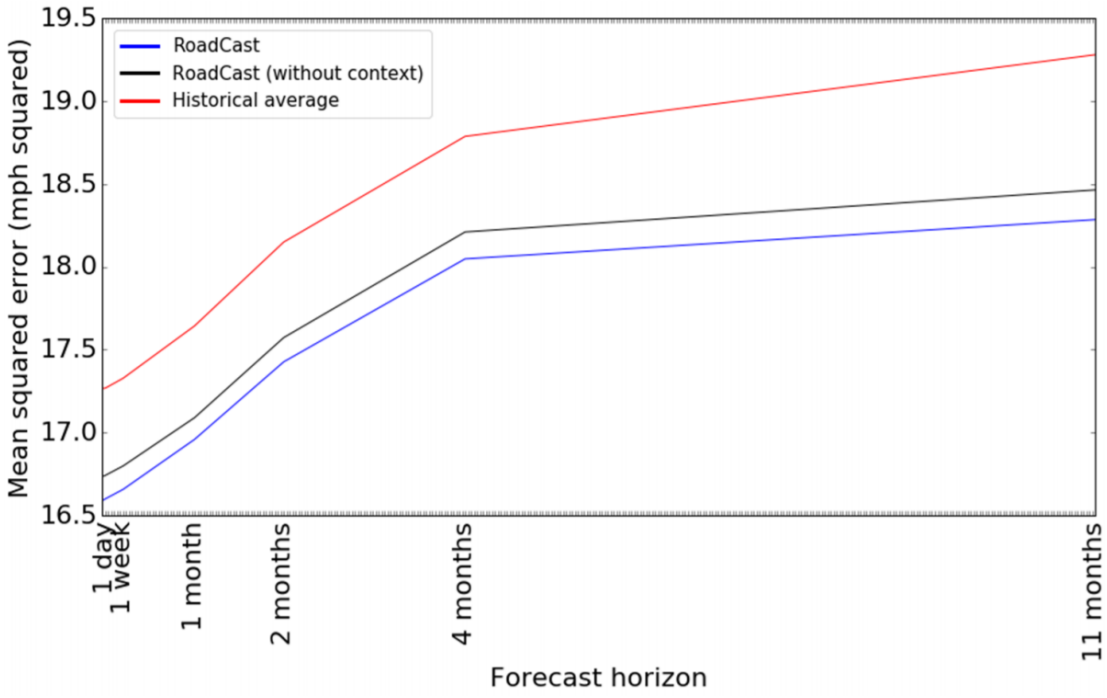


FIGURE 6.17: Different predictor's average mean squared error over all detectors when forecasting average speed at different horizons.

6.7.4 Discussion

This section tested RoadCast's ability to forecast specific traffic conditions, long in to the future. It was found that RoadCast's, and the historical average's forecasts degraded at similar rates with increasing horizons, both for flow and average speed. Using the findings of this section, recommendations on the forecast horizon of RoadCast can be made.

The tests found that although forecasts became less accurate at larger horizons, RoadCast remained more accurate than the historical average, and was more accurate when considering contexts. As the forecasting horizon became larger, the increase in error was small compared to the difference in accuracy between RoadCast and the historical average. As such, to achieve the most accurate forecasts possible, it is recommended that RoadCast be retrained as often as possible, on the latest year of data. However, it is noted that that less frequent retraining may be more practical for implementations in large networks or on computer systems with slow processing speeds or low memory capacities. Based on figures 6.15 and 6.17, it is recommended that acceptable accuracy levels can be achieved in such cases if retraining took place monthly. If more than two year's of data were available for this study, further recommendations could have been made based on the accuracies at multiple year forecast horizons, and training on multiple years of historical data.

It is clear that the loss in accuracy found would vary in different networks and forecasting different target variables. For example, areas with large net migration, or economic progress could have much larger losses in accuracy at larger forecasting horizons. As such, further research to repeat this test in different locations would validate the findings stated. However, the Southampton

network is diverse, it covers detectors on arterials, streets, and by signalised junctions, serving many types of travel demand. As such, the findings for this network were seen as a valid basis for recommendation as to how RoadCast could be implemented in new networks.

6.8 Interpretability

For many applications it may be important to understand how a traffic forecasting algorithm forms its forecasts, i.e. its decision making process. For example, in an incident detection application, an operator may want an explanation of why an incident alert was not raised even though a drop of average speeds occurred. In this case, the alert may not have been raised because the traffic forecasting algorithm knew that a drop in average speed would have been caused by travellers queuing to go to a nearby football match. With an understanding of the algorithm's decision making process, and in this case its use of contextual features, this intelligence could be communicated to the operator. This understanding could also help to explain why RoadCast's accuracy improved when contexts were incorporated (found in section 5 of this chapter).

RoadCast uses a random forest, which is a complex machine learning algorithm. Because of machine learning algorithms' complexity, many are seen as 'black boxes', in that their process of going from input to output cannot be easily understood. The complexity of random forests comes from the large number of trees created, and their size. However, unlike some other machine learning algorithms, there do exist methods that attempt to understand random forests' decision making process, and use of input features. This section implements these methods on the Southampton dataset in order to gain an understanding of how RoadCast's random forest forms its forecasts.

Some researchers have implemented feature 'importance' methods. There are a number of different methods that can be implemented to determine a feature's importance, but in a broad sense each aim to determine which features are most predictive in determining a random forest's predictions (Leo Breiman, 2017). Laña et al. (2016) used a feature importance method in a random forest algorithm that aimed to forecast pollution. Cloud type, wind speed, month and hour of the day were found to be the most important features, while precipitation was found to be surprisingly comparatively unimportant. Hou et al. (2015) used this method in a traffic flow forecasting algorithm designed for use in the presence of roadworks. The most important roadworks features were found to be the roadwork area's speed limit and the length of the roadwork area. In this study, a feature importance method is implemented in order to understand which features were most predictive in determining RoadCast's forecasts.

The structure of this analysis can be seen as two parts. The first implements methods which provide a measure of 'importance' for each input feature. These methods aim to understand how predictive each input feature is across a trained random forest. They do this by looking at either the structure of a trained random forest, or forecasts made on a validation set. The second part implements methods which take an individual prediction (of a message in a test set), and attempt to understand the decision making process of going from input to output. Such

methods are useful in understanding how an algorithm creates a particular prediction, but can fail to appreciate the bigger picture of the general patterns and strategies developed.

6.8.1 Feature importance

Feature importance measures attempt to estimate how predictive each input feature is in determining an algorithm's output. The more a feature is used in the key decisions made by the algorithm, the higher its relative importance.

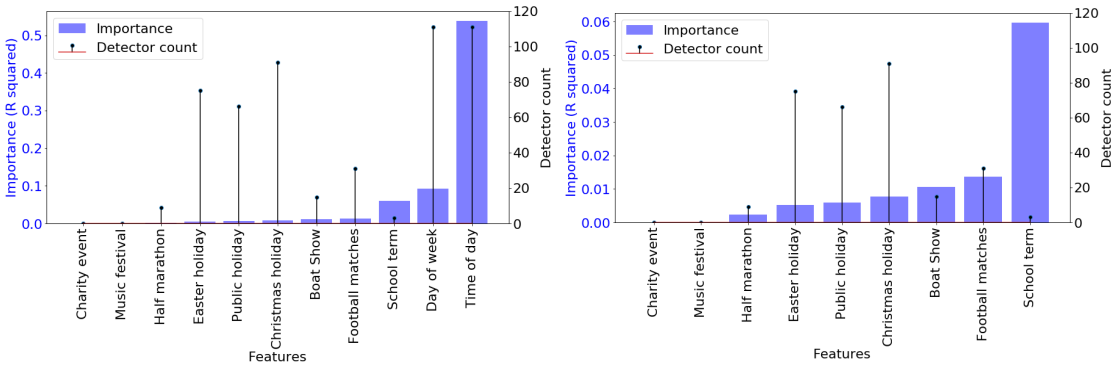
The area of research dedicated to developing feature importance methods is currently evolving quickly, and so there are a number of ways to estimate feature importance that have been presented in recent years. The most commonly used metrics for random forests are currently gini importance and permutation importance. However, each of these metrics have been found to exhibit bias in certain situations; gini importance has been found to be 'not reliable in situations where potential predictor variables vary in their scale of measurement or their number of categories', and permutation importance has been found to 'over-estimates the importance of correlated predictor variables' (input features) (Ghosh and Smith, 2014, Strobl et al., 2007). Drop-column feature importance is known to be a more accurate, but more computationally expensive method (Parr et al., 2018). For the purposes of getting the most accurate results, the drop-column importance method would be implemented in this research project.

6.8.1.1 Drop-column importance

The drop-column feature importance method is effectively a brute-force approach. The approach is to train a random forest a number of times, each time with all but one of the features included (a different feature would be left out each time). For each of these iterations, the accuracy of the model on a validation set is recorded, and compared to a baseline accuracy from when all features were included. The importance of a feature to a random forest is the difference between the accuracy achieved when it was removed and the accuracy achieved when all features were included.

In the case of RoadCast, a number of decisions were made in order to implement the drop-column feature importance method. Firstly, each random forest would go through the selection algorithm before this method was implemented. As such, only the importance of features included by the selection algorithm at each particular detector were calculated (values at other detectors would be 0). In this case, the validation set was chosen to be the out-of-bag samples from the random forest, i.e. the messages that were not randomly selected for inclusion in the bootstrap sampling process during the training procedure. In the Python framework used by RoadCast in this research project (Scikit-learn), the accuracy of the out-of-bag samples is only recorded as an R-squared value. As such, the unit of the importance of each feature would be the difference in the R-squared value. To calculate the importance of each feature across all random forests (one random forest for each detector), the average importance difference was calculated. That is, each feature's overall importance was calculated as the average importance at every random forest for which the feature was included.

Figure 6.18 shows the results of these tests. In blue is the importance of each feature, and in black is the number of detectors at which the feature was included. Figure 6.18a shows a graph of the percentage improvement of all features. Given that the time of day and day of week feature so prominently, figure 6.18a shows only the contextual features.



(A) All features.

(B) Contextual features only.

FIGURE 6.18: The accuracy improvement made by each feature in RoadCast.

Firstly, figure 6.18 shows that the time of day and day of week features were most important in forming RoadCast's forecasts. Without either of these features, forecasts would have been far less accurate.

Of the contextual features, the school feature had the greatest importance at the detectors in which it was included. Only three detectors included the school feature (because they had the closest proximity to a local school), but at such detectors the variation from travellers to and from school was clear (as was demonstrated in section 6.5.3.4). A factor that may have contributed to this feature's high importance value is that the variation occurred so frequently (most of the day on every week day).

The football feature had the second highest importance, likely due to the clear visible variation caused by football matches, and the frequency at which they occur. Christmas had the largest importance of the holiday features, likely due to it being the longest holiday period, and conditions differed from the norm the most on these days. The rest of the features had a small difference, likely due to them typically occurring less frequently, and causing less disruption to traffic conditions. The charity event and music festival contexts were not included at any detectors using the selection algorithm. As such, the importance of these features could not be ascertained. However, given that these features were not included by the selection algorithm at all, it can be concluded that these features were not important to RoadCast's forecasts.

6.8.2 Decision-making process interpretation

In this section, methods to interpret how RoadCast forms its predictions are implemented. These methods aim to gain understanding into RoadCast's decision-making process from input to output. The first method analyses the splits made by a particular decision tree in RoadCast,

and the second looks at the difference made by these splits (i.e. the difference between the target variable values in each node before and after a split is made).

6.8.2.1 Data

The methods in this section analyse one message at a time, rather than the previous method which aggregated measure over entire random forests. As such, this analysis would only be undertaken on a small subset of the total number of messages that were forecasted by RoadCast. Also, rather than using out of bag samples, this analysis would be undertaken on the test dataset, so that multiple predictions across a particular day could be analysed. Because of the complexity involved in analysing each message's prediction, a single day's forecast at one detector was focused on. Figure 6.19 shows the particular day chosen. The day was Saturday, 9th April 2016 at detector C, where a Premier League football match against Newcastle kicked off at 15:00 at St Mary's Stadium. In particular, the influence made by the football context was analysed. In each test, the same datasets as in the offline test section were used, i.e. one year of training and testing data.

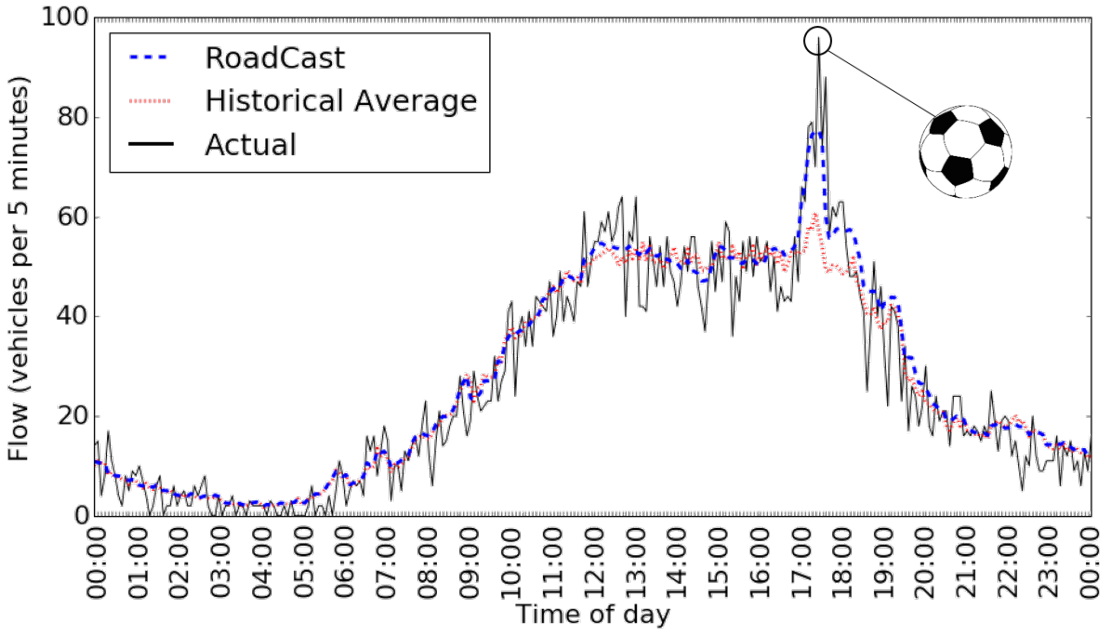


FIGURE 6.19: Traffic forecast for Saturday, 9th April 2016, at detector C. Premier League football match against Newcastle kicked off at 15:00 at St Mary's stadium.

The following subsections describe the results of a number of approaches taken to understand how RoadCast used contexts to form its forecast for the day described.

6.8.2.2 RoadCast's splits

In this section, the splits used to form RoadCast's forecasts were studied. By observing the splits made by RoadCast, the type of messages being averaged to form RoadCast's forecast is revealed. This gives insight into how RoadCast used contexts to form its forecasts. Firstly, the

method employed is described, then two message's splits are analysed.

Figure 6.20 shows an example of a tree in RoadCast's forest. It shows the first tree in the forest when forecasting flow at detector A, but to visualise the figure, a maximum depth of three splits was set (in reality the tree was much larger). In each tree in RoadCast, the predicted message goes from the root (top) of the tree, to the predicted leaf (bottom) via a number of splits of nodes. The path taken down the tree is called the decision path. Here, the top row represents the split made at each non-leaf node, i.e. the context chosen and the value on which the split was made. The 'messages' field on the second row represents the number of messages present in the given node. Finally, the mean represents the mean of the flow values of the messages in the given node.

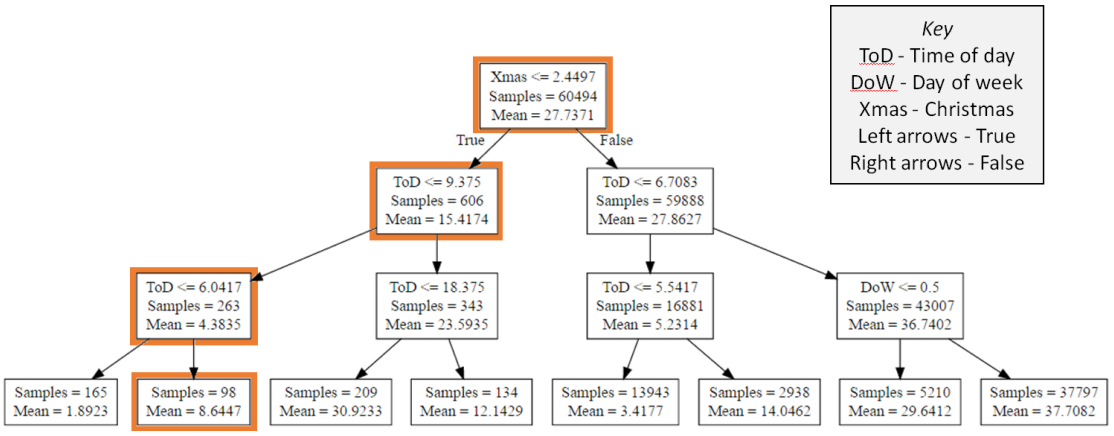


FIGURE 6.20: Example of a RoadCast decision tree for detector A, when predicting flow.

For a message at 8am on Christmas day ('Time of day'=8, 'Xmas'=0), the decision path highlighted in figure 6.20 would be taken. The tree's forecast is then simply the mean of the training messages in the predicted leaf, in this case 8.6. One can see that the splits made for this message were:

- Xmas<=2.45 (i.e. during the Christmas holiday)
- Time of day<=9.38 (i.e. before 9:22)
- Time of day>6.04 (i.e. after 6:03)

Because of the vast number of splits created by RoadCast, forecasts for two particular messages, in the first tree of the random forest were analysed. Again, both messages were from the forecast of Saturday 9th April 2016, at detector C, where a Premier League football match against Newcastle kicked off at 15:00 at St Mary's stadium (shown in figure 6.19). The first message was at 2pm, which was before the match started, where there appeared to be little difference in flow caused by the football match. The second message was at 5.20pm, the peak flow of the day, two hours 20 minutes after the match had started. The following paragraphs show the splits made for the 2pm and 5.20pm messages, in the first tree in RoadCast. It should be noted that the redundant splits aren't shown, for example 'Day of week>4' and 'Day of week>5' are replaced

with just 'Day of week>5'.

For the 2pm message, the splits in the first tree were:

- 13.9<Time of day<=14.24 (i.e. between 13:54 and 14:14)
- 5.5<Day of week (i.e. Saturday)
- Football<=0.05 (i.e. at most 1 hour 18 minutes after a match)

It can be seen that the decision tree's forecast for this message is the average over a small time period (13:55, 14:00, 14:05 and 14:10), but a large range of values of the football context. This indicates that for messages at most 1 hour 18 minutes after a match, the exact time until kickoff was not seen as an important indicator of flow, but the time and day of the week were. If the time until kickoff was more important at this point, RoadCast would use the football context to split further, separating messages from different kick-off times (e.g. 3pm and 5.30pm). However, by making the football split it did, RoadCast separated out messages that were over 1 hour 18 minutes after matches started, 'learning' that disruption occurred after this time (i.e. around when the matches end, which is around 1.5 hours after kickoff).

For the 5.20pm message, the splits in the first tree were:

- 15.9<Time of day<=18.5 (i.e. between 15:54 and 18:30)
- 5.5<Day of week (i.e. Saturday)
- 0.082<Football<=0.144 (between 2 and 2.5 hours after a match)

This message used an average over a wider range of time values, but a more narrow range of football context values. In fact, the time of day context was so wide that the predicted leaf included training messages that were on match days with different kick-off times, including 12.30pm and 5.30pm matches. These messages had similar values of the football context because they were a similar duration from the start of a football match, but were at very different times of day because the matches started at different times. So in essence, the forecast at this point was an average over all training messages on Saturdays that were at a similar proximity to a football match kick-off. This suggests that at this point, the time of day was less important to the decision tree than the time until a football match.

6.8.2.3 Feature contributions

Palczewska et al. (2014) presented a feature contribution method, which would be used in this study to understand the influence made by each context when splitting the training data. This method would provide further insight into how each context affected RoadCast's forecasts. The same tree and forecasted messages (2pm and 5.20pm) as the previous section were re-used.

As can be seen in figure 6.20, as one moves down the nodes of a decision path, the number of training messages in each node become fewer, and their means change in response to how the messages are split. These changes in mean from one node to another can be seen as the 'contribution' made by a context. Take for example the highlighted decision path in figure 6.20, the first split 'Xmas<=2.45' (i.e. at most 2.45 days after the start of Christmas day) reduced the mean of messages in the decision path from 27.7 to 15.4 vehicles. This can be seen as a contribution by the Christmas feature of -13.3 vehicles, reflecting that lower flows occurred during Christmas in the training period. The next split, 'Time of day<=9.375' (i.e. the time was no later than 9.23am), reduced the mean of flow values in the decision path from 15.4 to 4.4 vehicles, because early morning flows were lower than average in the training period. This can be seen as a contribution by the time of day feature of -11 vehicles.

Because the tree's prediction is simply the mean of the messages in the predicted leaf, the prediction of the tree can be seen as the mean of all the training messages +/- the contributions made by each of the splits. In the highlighted decision path, the prediction for messages in the predicted leaf is:

$$\begin{aligned} 8.6 =& 27.7 \text{ (training set mean)} \\ &- 12.3 \text{ (reduction from the split 'Xmas<=2.45')} \\ &- 11.0 \text{ (reduction from the split 'Time of day'<=9.38')} \\ &+ 4.3 \text{ (increase from the split 'Time of day'>6.04')} \end{aligned} \tag{6.2}$$

As such, the aggregated contributions made to forecast a value in this leaf would be:

- Christmas: -12.3
- Time of day: -6.7

Again, this indicates that RoadCast used the time of day and Christmas context to forecast a flow value less than the training set mean. This was because in training, lower than average flows were observed during the Christmas holiday, and of the messages during the Christmas holiday, values in the morning were lower.

To apply this method to the whole random forest, the mean over all trees of the aggregated context contributions can be taken. This metric represents the influence each context had in forming RoadCast's forecast for one particular message.

This method is not flawless however, principally because the order in which splits take place can influence the contributions given. In RoadCast, there is an element of randomness in choosing the feature used at each split, and a split made by a context early in the decision path may result in a different contribution than if the split had been made later. Take for example the highlighted decision path in figure 6.20, the first split makes a contribution of -12.3 vehicles for

the Christmas feature. However, if by chance the Christmas feature was used as the last split on the decision path, the contribution would likely be smaller because the values in the penultimate node already had a low mean due to the split made by the time of day feature. As such, this method can be seen only as an indication of the influence that contexts have on RoadCast's forecasts.

This method was employed to further understand the match day forecast of figure 6.19, specifically for the messages at 2pm and 5.20pm. Figure 6.21 shows RoadCast's forecast on one axis, and the contribution made by the football context on the other. It also shows details of each contribution made for the forecast of the messages at 2pm and 5.20pm.

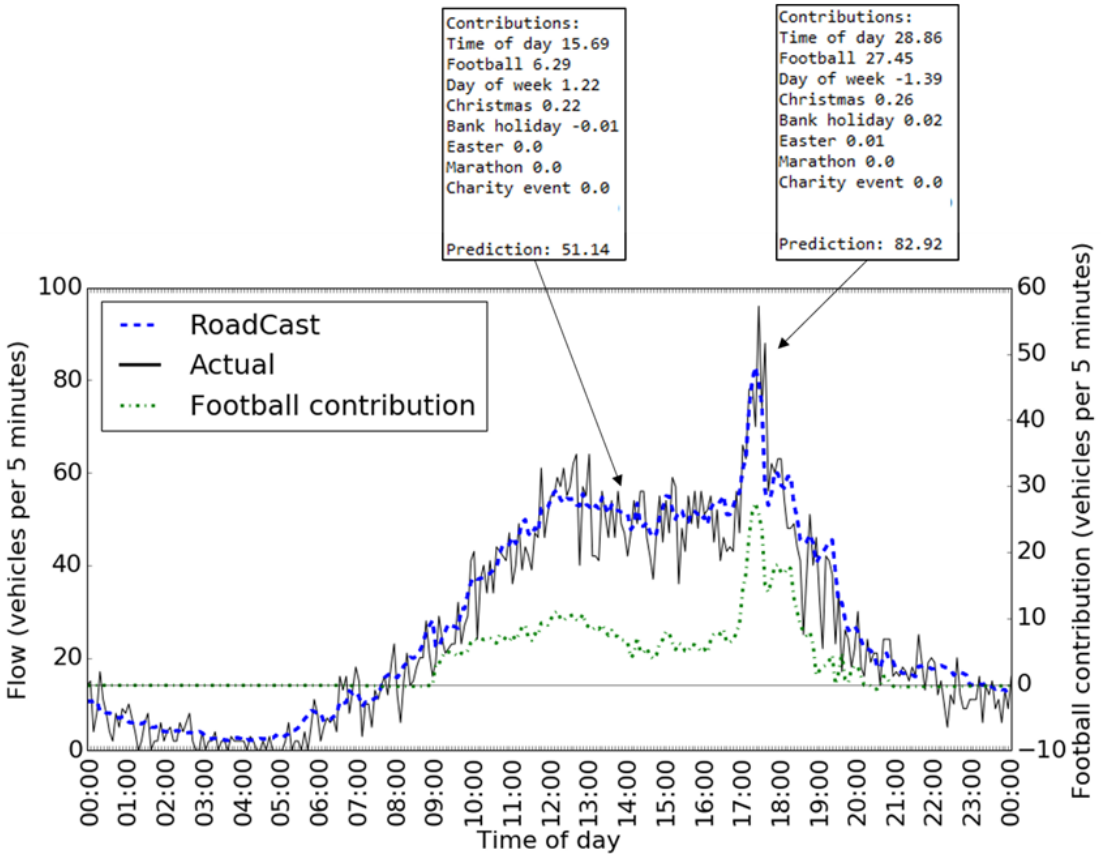


FIGURE 6.21: Contributions of RoadCast's forecast for Saturday, 9th April 2016, at detector C. Premier League football match against Newcastle kicked off at 15:00 at St Mary's Stadium.

As expected, the largest contribution came at the time of greatest flow, where football could be seen to visibly disrupt the traffic conditions. At this point, the contribution of the football feature was 27 (vehicles per five minutes). Despite there being only a small difference made by the football context in the hours leading up to the match, the contribution was consistently positive, at around five vehicles. This may be due to RoadCast splitting on the football context early, and so having a positive contribution because training messages close to football matches were higher than the average of all messages.

The contributions of each context for the messages at 2pm and 5.20pm give insight into the influence each context had on RoadCast's forecast. Firstly, the time of day made a positive contribution in both cases, but more so at 5.20pm. This was because flows were typically higher at this time, due to week day rush hours and football matches on Saturdays at 3pm (11 of 51 Saturdays had a match at 3pm). The day of week context made a positive difference at 2pm, but negative at 5.20pm. This was because at 2pm, the detector's flows were typically higher on Saturdays (perhaps due to travel demand being made up of shoppers and holidays makers, rather than workers commuting). At 5.20pm the contribution was negative because average Saturdays had lower flows than weekday rush hours. The only other contexts to make contributions were holidays, which were positive because holidays typically had lower flows than non-holiday periods.

6.8.3 Discussion

One advantage of the random forest over other complex algorithms is that there exist methods to interpret its decision making process. This section implements a number of these methods to help understand RoadCast's forecasts, in particular its use of contexts.

Firstly, the features that made most difference to RoadCast's forecasts were discovered. The time of day and day of week features were found to be most important to RoadCast. It appeared that these features were used as the basis of RoadCast's forecasts, and that the contextual features relied on such features to make inferences. The most important contextual features were public holidays (Christmas, Easter and other bank holidays), and football matches.

Methods to interpret RoadCast's decision making process were investigated for a day in which a football match caused disruption to a detector's traffic data. For messages closer to the football match's disruption, RoadCast's forecast was based more so on the football feature, and less so on the time of day and day of the week. This shows how RoadCast 'learnt' the times at which the football feature was an important indicator of traffic conditions.

A number of ITS applications could benefit from the interpretation methods' insight into the decision making process of machine learning forecasting algorithms. In particular, the contribution method could form the basis of a tool to explain RoadCast's forecasts to users. For example, profiles of RoadCast's forecasts could be presented to users with sections highlighted for particularly high contribution values of a particular contextual feature. These highlighted sections would indicate to users that the context was disrupting traffic conditions at the time. This could also be applied to an incident detection algorithm to give insight into its alert messages, such as 'no incident present, but conditions disrupted by local football match'.

6.9 Implementation procedure

As was previously stated, minimal calibration time, effort and expertise is required for many applications of a traffic forecasting algorithm. As such, RoadCast was designed in such a way that implementation would be simple, quick and as automated as possible. This has meant developing standardised feature encoding methods and an automatic selection algorithm, and evaluating RoadCast in a number of different scenarios. Using the results of these tests, this section describes how best RoadCast can be implemented in a new location. This implementation could be for the proposed incident detection application, or any of the other traffic forecasting applications highlighted in section 6.11.

The first step is to identify contexts which may affect road traffic conditions in the network being implemented, and a data source from which their schedules can be collected. This list does not have to be exhaustive, as any relevant context added should only improve the forecasts made. Also, not every context identified has to necessarily affect conditions, because the selection algorithm will later rule out contexts that aren't relevant to each detector and target variable. This step requires some local knowledge and intuition, but this would be expected in the possible applications of RoadCast, operators in TMCs. For this study, the author's local knowledge was relied upon to identify relevant contexts in Southampton.

Next, one year's historical data requires collection, both for the traffic variable being forecasted, and the identified contexts. One year was found to result in the best forecast accuracy, because all contexts could be accounted for. Single day event contexts require a start time, and multiple day contexts require a start date and end date. For this study, the data was collected manually by visiting relevant websites, but for implementation in real-world applications, an automatic web scraper would be recommended.

The next step is to encode each context as features to input into RoadCast. To do this, the standardised methods of encoding features, defined in table 5.2, should be followed.

At this point, the selection algorithm can be run. This takes the training data as input, and returns features and random forest parameters that will be used at each detector and target variable. It will also retrain RoadCast on such features and parameters on the entire training dataset.

Finally, RoadCast can make forecasts of future traffic conditions. RoadCast was found to be most accurate when forecasting at the shortest forecast horizons tested, up to one day. As such, re-training is recommended as frequently as possible. In this study, RoadCast took 8.1 minutes to fully train and test per detector and target variable (30 hours for flow and average speed on all 111 detectors), which was achieved on an Intel(R) Core(TM) i7-6700, 3.40Ghz, 16GB RAM. As such, re-training every day may not be practical in practice, particularly on networks with many detectors. RoadCast's flow MSE increased by 3.6% percent as the horizon increased from one day to 11 months. Hence the re-training frequency is a trade-off between accuracy

and computation cost and time, which needs to be considered on a case by case basis during implementation.

6.10 Limitations

Evaluating RoadCast on the test set resulted in some promising results, and suggested RoadCast could be used as a basis for an IDA. However, a number limitations of the algorithm were also identified. This section describes the limitations found.

Perhaps the largest limitation of RoadCast is the amount of data required to train. Of all the contexts studied, the least frequent occurred at least once per year. This means that for all contexts to be accounted for, one year's worth of training data would be required. This limitation is not exclusive to RoadCast, instead it would affect any algorithm that incorporated such infrequent contexts. In chapter 6, the performance of RoadCast under different training periods was studied, and it was found that RoadCast improved with more data primarily because it had more contextual data to 'learn' from.

When implementing RoadCast in a new location, a certain amount of knowledge and time is required to identify relevant contexts. For this case study in Southampton, contexts were identified using local knowledge and observation of traffic data, and relevant data sources were searched for manually using internet search engines. It is envisaged that a TMC operator would undertake this task when implementing RoadCast in a new network. Future research would be required to automate this process, i.e. creating an algorithm capable of automatically searching for and comparing internet sources and traffic data in order to identify relevant contexts that affect the given road network, and then automatically web scraping and storing this data.

These two limitations can be seen as a trade-off from the benefit gained from incorporating contextual data. It was clear that contexts occurred irregularly but caused variation in traffic patterns. So to account for this variation, it was seen as necessary to incorporate the schedules of contexts. The benefit of doing this has been demonstrated in section 6.5.2. But this benefit comes at the cost of requiring sufficient training data, and time to identify suitable contexts.

If there was a major change in traffic conditions at a detector during the testing period, RoadCast was prone to making inaccurate predictions after the change. This change in traffic conditions could have many causes, such as a change in topology, travel demand, or detector malfunction. There were at least five detectors which were found to have such a change during the test period in Southampton. In Bristol there were 21. For the offline test in Southampton, these detectors were left within the results, but in the online test in Bristol such detectors were identified by the operators and removed in order to improve the operators' user experience of the web app.

If RoadCast were to be re-trained regularly, inaccurate forecasts could occur at detectors with major changes until the training period was entirely after the change. To limit this drawback,

manual intervention could be used, or an algorithm to automatically identify such a change, and re-train RoadCast accordingly, could be developed. It should be noted that two approaches to address this issue were investigated, but neither improved the accuracy of the algorithm. The first attempted to find the point in the training data at which the change occurred. Methods developed to find this point were unreliable, often confusing the change point with other causes of variation such as contexts. The accuracy of forecasts often worsened because of the algorithm having to train on less data. The second method was to include a time of year feature (encoded as an integer representing the day of the year), such that the algorithm would split the data so as to use data from outside the period where the major change was affecting traffic conditions. It should be noted that this would only be beneficial if the representative period constituted the majority of the training data. This would limit the time for which inaccurate predictions could be made for. However, this method resulted in lower accuracies because RoadCast would on occasion use this feature to split the data too far, and so produce forecasts overfit on too few samples.

In some cases (see section 6.5.2), RoadCast could predict inaccurately when inferring from incidents in the training data, which was particularly apparent when these incidents occurred during rarely occurring contexts. For example, if roadworks occurred during Easter in the training period, RoadCast could forecast conditions similar to these roadworks for Easter during testing. This is an issue for RoadCast because it aims to forecast expected conditions, but is in part trained on conditions during incidents. However, with sufficient training data, this issue is thought to be limited because of the rarity of such incidents. To rectify this, a method to identify incidents in training data could be developed. Such a method was not developed in this study because the impact of this issue was small enough that RoadCast's forecasts still improved when using contexts, and were deemed suitably accurate for the proposed IDA.

RoadCast's forecast accuracy was limited by variations in the traffic data that were not accounted for. Some causes of variation may have been missed, and others may not have been suitable for incorporation. For example, disruption during particularly busy shopping days could be identified (and verified by Southampton City Council tweets), but not predicted beforehand. Another example is the noise in the data, which could have come naturally from the unpredictable arrival time of individual vehicles, or from distortion in the detectors collecting and transmitting the data. Some variation in the traffic data appeared to be inherently unpredictable. For example, one detector experienced a peak in flow in the evening rush hour every week day. On some occasions this increase in travel demand would result in a drop in average speed due to congestion (and hence a drop in flow), but on other occasions no such drop in average speed or flow would occur. There were no contextual factors found that could explain when each scenario would occur, and so RoadCast was often visibly inaccurate at these times. Possible explanations for this phenomenon include alterations in the nearby signalised junction's control strategy, or variations in individual's behaviour, such as a vehicle turning suddenly causing a 'phantom jam' (Kerner and Konhäuser, 1993).

The standard encoding methods created in this study allow RoadCast to be implemented in other networks with minimal manual calibration time and effort, while allowing contexts to be incorporated effectively. However, because this method has not been tested extensively across

many different road networks, and hence many different local contexts, it cannot be concluded that the method is suitable for every local context that could exist. As such, a potential limitation of RoadCast is the suitability of these encoding methods to untested contexts, such as local concerts or parades.

RoadCast was designed to be as transferable as possible so that it could be implemented in a number of real-world ITS applications. However, because this test only evaluated data from Southampton, conclusions cannot be made on RoadCast's transferability to other scenarios, such as different detector types, levels of aggregation and road types.

6.11 Conclusions

This case study evaluated RoadCast, a novel algorithm that forecasts expected traffic conditions. The algorithm was tested by forecasting loop detector flow and average speed up to a year ahead of time. It was compared to a historical average predictor, and the effect of various contextual features were evaluated.

RoadCast was found to be more accurate than the historical average in forecasting both flow and average speed, by 4.4% and 4.0% respectively. Significant reductions in error were achieved by incorporating contextual data within a machine learning algorithm.

Although RoadCast was designed for use within an incident detection algorithm, it is thought that it could be developed for many other applications within ITS and beyond. Possible applications include:

- Route guidance systems, for planning journeys multiple hours or days ahead of time.
- Varying strategy of public transport and planned roadworks in response to forecasted traffic conditions. A project by Purple WiFi ltd. and Cisco Systems Inc. is using real-time traffic data to alter public transport schedules (Purple and Cisco, 2017). The idea being that when major events occur, they will be detected and responded to with demand-responsive public transport services. This could be improved by using context based forecasts of traffic conditions to pre-empt the demand for public transport services.
- Varying congestion charges and tolling based on forecasted congestion levels. For example, Stockholm's congestion charging scheme was introduced based on different times of day and day of the week (Eliasson et al., 2009). This approach could be improved by using charges based on the amount of travel demand forecasted. This could be particularly effective in mitigating the sudden increase in travel demand caused by major events, encouraging alternate methods of travel that disrupt road network conditions less.
- Varying logistic companies' schedules and routes based on congestion forecasts.

This chapter has shown that RoadCast meets the requirements of section 5.2. It was able to forecast traffic variables suitable for incident detection (5-minute values of flow and average

speed), improved by using contexts, and could be assumed to be forecasting 'expected' traffic conditions, as the forecasting horizon was up to a year. As such, this study indicated that RoadCast would be suitable for the proposed IDA. Next, RoadCast would next be developed into an incident detection algorithm, RCID, in order to answer the question of whether IDAs can be improved with the incorporation of contextual data.

Chapter 7

RoadCast Incident Detection methodology and offline test

7.1 Introduction

This chapter describes the development and offline test of an IDA based on the approach outlined in chapter 4. This IDA is named RoadCast Incident Detection (RCID), because it is based on RoadCast's forecasts.

The offline test will involve the same dataset as was used to evaluate RoadCast, i.e. historical loop detector data from Southampton. Performance will be determined using incident messages from Southampton's TMC, and by making comparisons to other state of the art IDAs, RAID and McMaster. The aim of this evaluation is to prove or disprove the hypothesis of this research project, i.e. to understand whether an IDA can use contextual data to better understand traffic conditions that can be expected to occur, and hence differentiate incidents from contexts. In doing so, this evaluation will also address objective three of this research project, i.e. to evaluate the developed IDA in order to determine whether the issue of differentiating incidents from contexts has been addressed, and whether it is an improvement on the state of the art (see section 1.8).

7.2 Methodology

The methodology of RCID is based on the approach described in chapter 4, and can be described as two key steps. Firstly, to create a forecast of a target traffic variable (such as flow) that represents what could be expected if no incident were to occur. Then, to compare this forecast to real-time values of the target variable, and to raise an alert when a sufficient difference is observed.

The following sections describe the development of the RCID methodology. Initially, the proposed methodology is designed to be simple in order to determine whether the approach taken did

indeed improve the differentiation of incidents and contexts. Later in the thesis, the methodology will be iterated on in an effort to develop the most effective RCID methodology possible.

7.2.1 Prediction intervals

The RoadCast algorithm evaluated in chapter 6 produced a prediction of a single value, representing the algorithm's 'best guess' of the target variable. However, as was described in section 6.10, some variation was unaccounted for, and so at times uncertainty laid within RoadCast's forecast. Clearly the uncertainty of RoadCasts forecast can vary based on the particular message being forecast. For example, disruption in football matches in Southampton appeared to have more variation between occurrences than public holidays, resulting in more uncertainty in future forecasts. This uncertainty was observed by looking at the spread of values in each leaf of each decision tree in RoadCast. Leaves holding a wide spread of values would lead to more uncertain forecasts represented by wider prediction intervals.

Although a single value prediction output would be most suitable for many ITS applications (such as dynamic congestion charging), when developing RCID it became clear that an understanding of the uncertainty in the traffic forecast would benefit the incident detection application. With such an understanding, RCID could be made less sensitive to raising alerts when forecasts were uncertain (and vice versa), such as when a rarely occurring context was due to occur, or when an unaccounted cause of variation would disrupt traffic conditions.

Based on this observation, it could be seen that a more suitable approach for RoadCast would be to produce a range of expected values, i.e. a prediction interval, rather than a single value forecast. A benefit of the random forest algorithm is that there exist methods to produce such prediction intervals. As such, RoadCast would be modified to produce prediction intervals, which would be used as input to the incident detection methodology of RCID.

A prediction interval is an estimate of an interval for which future observations (of the target variable) will fall into with a given probability. A number of methods to produce prediction intervals were considered, but the only method found to consistently produce accurate intervals (i.e. the percentage of future observations that fell inside the interval was consistently closest to the given prediction interval width), was a method first proposed in (Meinshausen, 2006).

As explained in section 5.6, a random forests forecast is the mean of each trees forecast, where each trees forecast is the mean of the target variable values in the trees predicted leaf. Instead of using this to produce a 'best guess' prediction, prediction intervals were created by taking the appropriate percentiles of all the target variable values of the messages in every trees predicted leaf (Meinshausen, 2006). For example, a 95% interval is the range from the 2.5^{th} and 97.5^{th} percentiles of the values in every predicted leaf in the random forest. This means that real-time traffic variable values should fall within the prediction interval approximately 95% of the time. Based on the literature review undertaken (described in section 2), it is thought that this is the first time that this random forest prediction interval methodology have been used as part of an

IDA.

The use of prediction intervals is predicated on the assumption that the distribution of values that it is used over comes from a normal distribution. This was visually checked by analysing histograms of the trees' values, which appeared to show that normality held. A more rigorous test of the method's validation was undertaken by making a preliminary test on the training data. Using cross-validation, the method created 95% prediction intervals on the training data. Next, the percentage of actual flow values that fell within the produced prediction intervals was determined. It could be seen from this test that the prediction intervals produced were on average, accurate estimates of the underlying distribution of traffic flow values. As such, this method was deemed suitable for use in RCID.

7.2.2 Incident detection logic

In a preliminary test on the training data, RCID would simply raise an alert when real-time values of the target variable fell outside of the prediction interval. With this method, variation from noise in the traffic data would result in many unnecessary false alerts being raised, and at times incidents going undetected. As such, a persistence test of three messages was introduced. This aimed to ensure that alerts would only be raised when the underlying trend of the target variable had truly deviated from what the forecasting algorithm had expected. This persistence test would improve RCIDs false alert and detection rate, but at the cost of worsening its average time to detect.

The following flow chart in figure 7.1 shows the steps required for RCID to raise an alert.

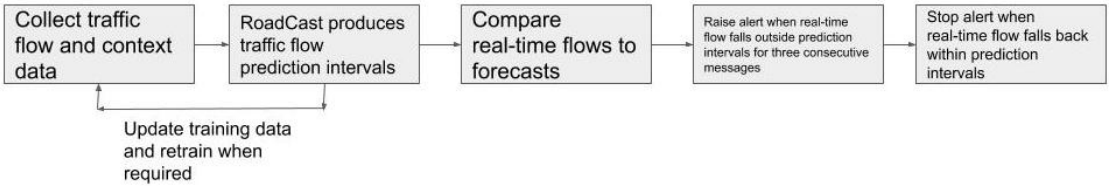


FIGURE 7.1: Flow chart showing RCID's incident detection method.

A persistence test for turning off alerts was also considered. In preliminary tests, RCID appeared to occasionally raise multiple alerts for what appeared to be the same incident. This occurred when the trend of messages was clearly outside a prediction interval, but occasional messages distorted by noise would cause the alert to end. The introduction of an off persistence test aimed to rectify this. In preliminary tests, the off persistence test resulted in a tendency to raise alerts less frequently but for longer periods of time, covering the entire period that an incident disrupted traffic for. However, it also resulted in an increase in false alert rate due to the increase in alert duration. Also, at times when an incident had clearly stopped disrupting traffic, and RoadCast's forecasts followed the actual traffic conditions, the alert could continue due to the persistence test using messages that were distorted by noise. As such, it was decided that an off persistence test would not be used. However, it is thought that in practice, the right choice as to whether to use an off persistence test would depend on the preferences of the TMC. For

example, for TMCs that aim to automate the task of incident detection and management, an off persistence test may be preferred because each alert is more likely to cover the entire period of time that an incident disrupts traffic for, which may be necessary for automated incident management such as altering traffic signal strategies.

It is noted that there are a number of other methods that could have been used to detect incidents based on RoadCast's forecasts. Such as by setting a fixed threshold of the difference in the traffic variable between RoadCast's forecasts and real-time traffic data, or using a different method to estimate RoadCast's forecast uncertainty. However, the described method was found to be sufficiently simple and effective in preliminary tests to be able to test the hypothesis of this research project. Attempts to improve on this initial methodology are then experimented with in subsequent sections. If RCID is found to be able to differentiate incidents from contexts, and improve on the state of the art, then the research hypothesis of this project will be concluded as true.

7.3 Initial offline test

RCID will first be evaluated in an offline test. The aim of this test is to understand RCID's ability to detect incidents, in particular in comparison to the state of the art, and whether it can improve by using contexts.

This test involves training and testing RCID on historical data, and evaluating its performance by comparing its alerts to incident logs. The following sections describe the details and results of this test.

7.3.1 Data

7.3.1.1 Location

A number of locations were considered for the offline test of RCID, but many were ruled out because of inadequate incident data. The most reliable sources of incident data were found to be from Councils' TMC logs, and Police and emergency service records. The UK Government provide Police data of road accidents that involve personal injury across Great Britain since 1979 (UK Government, 2018), but this data only covers a subset of all types of incidents. Other types of incidents, such as disruption from roadworks, breakdowns and illegal parking were not covered. It appeared that the only data source that provided information on all types of incidents were TMC's incident logs.

In the interview with Siemens' Head of Consultancy Services, it was found that three of Siemens' client's TMCs record incident logs (see section 3.3). Each of these incident data sources were not available for reasons including; privacy of the personal data recorded in the logs, insufficient location data of the incidents, and a lack of incidents recorded. However, Southampton City

Council and Hampshire County Council disseminate their TMC's incident logs to the public via Twitter (Hampshire County Council, 2018, Southampton City Council, 2018). When studying this data, along with the already acquired loop detector data, it appeared that this data source gave sufficient coverage of the network and was of sufficient quality to evaluate RCID. As such, Southampton was chosen as the case study on which to evaluate RCID offline.

7.3.1.2 Traffic and contextual data

As described above, Southampton was chosen as the location for the offline test. Southampton City Council's TMC is responsible for an area surrounding the loop detectors used in the test of RoadCast (described in chapter 6), so the incident tweets covered the required area. As such, the traffic data used to evaluate RoadCast was found to also be suitable for the offline test of RCID.

A description of the loop detectors used, their location, and the pre-processing used can be found in section 5.3. A description of the contextual data used can be found in section 6.2. As with the evaluation of RoadCast, the first year of traffic data would be used for training of RoadCast, and RCID would be implemented and tested on the second year of data. As with the offline test of RoadCast, this decision was made so that all contexts used could be 'learnt' from during training, and evaluated during testing.

7.3.1.3 Incident data

As described previously, incident data was collected from a Twitter feed provided by Southampton City Council and Balfour Beatty (Southampton City Council, 2018). The tweets covered the testing period of 14th December 2016 to 16th March 2017. By comparing this dataset with the available loop detector data and cross-referencing with other online sources, including the STATS19 UK Government crash dataset (UK Government, 2018), this Twitter feed was judged to have sufficient coverage and reporting quality to evaluate RCID.

It should be noted that limitations exist with using this Twitter dataset as a ground truth. It may be the case that some incidents detected by operators in Southampton's TMC are not disseminated via Twitter. Other incidents may have caused disruption, but were not detected by the TMC at all. There may also be inaccuracies in the reported time, location or severity of incidents in the tweets. Another limitation is that IDAs time to detect cannot be evaluated, because the time that a tweet is posted may have a variable delay from when the incident first occurred.

Not all of the tweets on the feed were suitable for the evaluation of RCID. Firstly, many described disruption from contexts rather than incidents. As such, only tweets with a description of an incident were considered. RCID could not be reasonably expected to detect incidents that did not cause any disruption to a detectors traffic conditions. As such, a number of further

pre-processing steps were undertaken to determine cases of incidents disrupting traffic conditions.

Firstly, the search space was narrowed by making the assumption that a detector's traffic could only be affected by an incident that was at most 5 kilometres (km) away by vehicle. As such, a filter to only consider incidents that were within 5km driving distance of detectors was implemented. The driving distance was found using Google Maps Distance Matrix API (Google Maps, 2017). Unfortunately, there is a limit to the number of calls that can be made using this API. As such, the direct distance was first calculated to reduce the search space. To do this, the Python package 'geopy' was used to calculate the Vicenty distance, i.e. the distance based on the assumption that the earth is an oblate spheroid (Geopy contributors, 2017). Each incident was paired with a detector if it was within 5km Vicenty distance. The driving distance between each pair (i.e. each detector and incident that were within 5km of eachother) was then calculated using the Google Maps Distance Matrix API (Google Maps, 2017). Only pairs that were within 5km driving distance were kept. The driving distance was used because this represents the distance in which queues could be, which may be far longer than the direct distance.

Next, to ascertain which incidents disrupted traffic conditions, and at what times, each tweet of an incident was investigated by manually observing nearby loop detectors traffic data and comparing with historical average values. Only tweets of incidents that were identified as disrupting at least one detectors traffic conditions (of any of its variables) were considered. The start time and end time of each incident's disruption was recorded at each detector.

Finally, a column of data was added to the traffic dataset of each detector. This column was given a value of one when at least one disruptive incident was occurring, and zero otherwise. This resulted in a traffic dataset that indicated when incidents within 5km driving distance were causing disruption at each detector. After completing this process, 113 cases of incidents disrupting a detectors traffic conditions were identified, from 37 seperate incidents.

7.3.2 Performance metrics

The most commonly used performance measures of IDAs are detection rate (DR), false alert rate (FAR) and average time to detect. These three metrics were chosen as the metrics to evaluate RCID. Unfortunately, the exact time of incidents was not stated in the incident tweets. Because there would be a variable delay between incidents occurring, operators detecting them, and tweets being posted, the time-stamp of tweets would also be unsuitable for evaluating RCIDs average time to detect. As such, only DR and FAR were used as performance metrics. These metrics would help understand RCID's accuracy and reliability in detecting incidents, which are important considerations for operators in TMCs (Guin, 2004). As was stated in section 2.3.3, the definition of FAR and DR vary throughout the literature. The definitions chosen for this project were the most commonly used definitions, which after reflection, were judged to be most suitable. The definitions were deemed to best reflect the performance required by operators (based on the TMC reviews in sections 3.2 and 3.3), whilst being sufficiently feasible to be possible in

the following tests.

Each IDA would be judged to have correctly detected an incident if an alert was raised while an incident was disrupting the detectors traffic conditions (this period was ascertained by comparing the detectors traffic data with a historical average). DR was defined as the number of correctly detected incidents, divided by the number of incidents in the Twitter dataset.

FAR was defined as the number of messages where an alert was raised incorrectly, divided by the total number of messages where an incident was not occurring. Another metric, number of false alerts per detector per day, was also used to give a more clear understanding of the number of false alerts that TMC operators could expect when implemented. It should be noted that an incident alert could span multiple consecutive messages.

7.3.3 IDAs for comparison

A number of factors were considered when choosing which algorithms to compare with RCID. These included:

- Implementation feasibility, i.e. whether clear and detailed guidance on how to implement the algorithm are available.
- Implementation requirements, for example if certain traffic variables are required that are not available from Southampton's loop detectors.
- Stated performance achieved in the literature.
- Commonality of use in the literature and in practice.

Unfortunately, many state of the art IDAs were unsuitable for comparing to RCID on the available data. Below are the most common reasons for why state of the art IDAs were not suitable.

- The IDA requires too large an amount of incident data for training. This is often the case for IDAs designed to 'learn' what conditions can be expected when incidents occur. Such IDAs are often unable to train on real-world data because of the infrequency of incidents occurring, and so instead require a transport simulation of the network in which the IDA is being implemented. It was considered infeasible to implement IDAs that require such training on a simulator.
- The IDA requires adjacent or upstream/downstream detectors. Many IDAs compare nearby detectors' data to detect incidents that occur in-between them. The dataset RCID was implemented on was of SCOOT loop detectors in Southampton, and so were typically placed on the entry and exits of signalised junctions. As such, there were too few adjacent or upstream/downstream pairs of detectors to implement IDAs that were based on this approach.
- The IDA only works on a different data source or target variable e.g. probe vehicle data, social media data, journey time data etc.

- The literature describing the IDA does not give sufficient detail for the IDA to be replicable.
- The IDA has not been evaluated or has not been compared to other state of the art IDAs in the literature.

Although these reasons limited the choice of feasible IDAs to compare RCID with, two were found to be suitable. The following subsections describe these IDAs.

7.3.3.1 McMaster

The McMaster IDA is a comparative IDA that was reviewed in section 2.4.1.3. The first presented version of the IDA raised alerts if traffic conditions were below a threshold average speed, or in a certain area of a flow/occupancy graph (Persaud et al., 1990). The IDA was designed for motorways, and for 30 second data. However, based on the reasons above, the McMaster IDA was judged to be the most feasible and suitable to implement for comparison to RCID. It was hoped that because the methodology was simple and not specialised for motorways in particular, the IDA would transfer well to urban networks.

A drawback of implementing the McMaster algorithm is the manual calibration needed to define the areas of the flow/occupancy graph that represent incident conditions. To define these areas manually on all 111 detectors would be impractical, and so a method to automatically calibrate the algorithm was sought after. Weil et al. (1998) developed such a method using historical data, which would be used in this study. With this method, the McMaster methodology becomes the following; a message would be considered to be representing an incident if one of the two following inequalities held:

$$SP < \mu - \beta\sigma \tag{7.1}$$

where SP is the value of average speed of the message being tested, and μ is the mean of average speed values in the training data, σ is the standard deviation of average speed values in the training data, and β is a constant that can be used to alter the sensitivity of the IDA.

$$FL \leq \mu_o - \alpha\sigma_o \tag{7.2}$$

where FL is the value of flow of the message being tested, μ_o is the mean of the flow values of messages in the training data that have equal occupancy to the message being tested, σ_o is the standard deviation of the flow values of messages in the training data that have equal occupancy to the message being tested, and α is a constant that can be used to alter the sensitivity of the IDA.

If real-time messages satisfied one of these two inequalities for three consecutive messages (the same persistence test as used by RCID), an alert would be raised. Once an alert has been raised, the alert would be turned off once both of these inequalities were not satisfied for three consecutive messages.

Because of its performance and commonality of use in the literature, and feasibility of being replicated, the McMaster IDA was chosen for comparison with RCID.

7.3.3.2 RAID

RAID is a comparative IDA that was reviewed in section 2.4.1.6. It uses loop detector values of average loop-occupancy time per vehicle (ALOTPV) and average time-gap between vehicles (ATGBV) to detect incidents. ALOTPV is the average time period that each vehicle spends occupying the road space above a loop detector, and ATGBV is the average time period in-between each vehicle occupying a detector. Each variable was calculated directly from the occupied' and non-occupied states of the loop detectors, which were sampled every 0.25 seconds. In the literature, RAID was evaluated on 30 second aggregates of ALOTPV and ATGBV. However, because only 5 minute aggregates were available in this study, RAID would use 5 minute aggregates of ALOTPV and ATGBV.

The presented IDA would judge a message as being representative of an incident if it was above the 85^{th} percentile of the training data ALOTPV values, and below the 15^{th} percentile of ATGBV values in the given peak or off-peak period. However, it was noted that these percentiles could be changed to alter the sensitivity of the IDA. Peak periods were defined as being 07:00-09:30 and 16:00-19:00. If the values broke these thresholds for three consecutive messages during the off-peak period, or four consecutive messages during the peak period, an incident alert would be raised. This alert would then stop when either of the thresholds was not met.

Because RAID had reported positive performance in an online test as well as in theory, and because of its feasibility of being replicated, the IDA was chosen for comparison with RCID.

7.3.4 Implementation details

The RCID, RAID and McMaster IDAs described above were trained on the first year of the Southampton dataset, and tested offline on the second year. The performance of the IDAs were then evaluated using the performance metrics described in section 7.3.2.

As was stated previously, there exists a relationship between the false alert rate and detection rate of an IDA (see section 2.3.3). If an IDA's parameters are tuned to make it more sensitive to raising alerts, its detection rate will increase, but so will its false alert rate. The required performance of detection and false alert rate can vary between different TMCs depending on the use of the IDA. For example, if a TMC had operators present at all times, a comparatively more sensitive IDA may be more suitable, so that more incidents would be detected, and operators could discard the extra false alerts by manually checking each alert.

In the literature, both RAID and McMaster were said to have certain parameters which could be changed in order to alter the IDA's sensitivity to suit the TMC's needs. As such, each of the

IDAs were tested with many different parameters settings, which would help to understand their performance with respect to this trade off.

For McMaster, the values of α and β could be altered. From initial tests on the training data using a grid search over α and β, it appeared that the best performance was achieved using an α value of 1.75. As such, for this test, α was set to 1.75, and β would be used to alter the IDA's sensitivity.

For RAID, it was recommended that to change the sensitivity, the percentile threshold of ALOTPV and ATGBV values should be altered. To increase the sensitivity of RAID, the ALOTPV percentile should be decreased, and the ATGBV percentile should be increased by the same amount.

For RCID, the prediction interval percentile will be altered. With a greater percentile, the IDA will be less sensitive to raising alerts, and so will have a lower detection rate but lower false alert rate.

7.3.5 Results

Figure 7.2 shows the performance of RCID, RAID and McMaster. By plotting the detection rate and false alert rate of the IDAs on two axis of the same graph, the trade-off of false alert rate and detection rate can be seen. When each IDA's parameters were altered to increase sensitivity, detection rates would increase, but false alert rates would also increase. The area highlighted in grey are the limits of 'acceptable performance' deemed by TMC operators in a survey on incident detection, which were for "DR to be at least 88.3% and of the FAR to be at most 1.8%" (Ritchie and Abdulhai, 1997). It should be noted that this survey used the same definitions of FAR and DR as were being used in this test.

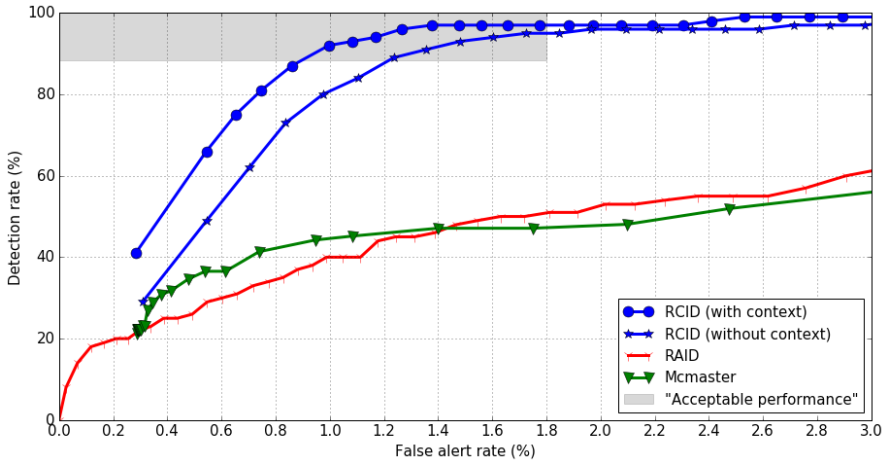


FIGURE 7.2: IDAs' false alert rates and detection rates. The grey area represents bounds for which a survey of TMC operators deemed 'acceptable performance' (Ritchie and Abdulhai, 1997).

The performance of RAID and McMaster was largely similar at various degrees of sensitivity. McMaster had slightly higher detection rates at low false alert rates, but lower detection rates at higher false alert rates. However, RCID had a better false alert rate and detection rate than McMaster and RAID at every point on the curve. RCID was the only IDA that achieved performance that was within the 'acceptable performance' boundaries of the TMC survey. The performance of RCID with contexts was consistently better than RCID without contexts. At every prediction interval setting, both the detection rate and false alert rate improved when contexts were incorporated. Taking the 90% prediction interval as an example, the detection rate improved from 94.4% to 96.7%, and the false alert rate improved from 1.75% to 1.50%.

To assess the significance of the results achieved, paired t-tests were conducted on RCID's and RAID's false alert rates and detection rates independently, using the Python library SciPy (SciPy, 2018). Specifically, the version of RCID with a 90% prediction interval, and RAID with a 60^{th} percentile ALOTPV threshold (and hence 40^{th} percentile ALOTPV threshold) were used. This setting of RAID achieved a 64% DR and 2.94% FAR. Again, a 5% significance level was used.

There is a statistically significant difference between RCID with and without contexts at a 90% prediction interval in terms of false alert rate (p-value of 0.000), but not for detection rate (p-value of 0.32). This value is likely high because of the low sample size of 113 incident cases. RCID (with context) was then compared to RAID. In this case, there was sufficient difference to claim statistical significance in both DR and FAR cases, with p-values of $1.6 * 10^{-7}$ and 0.000 respectively.

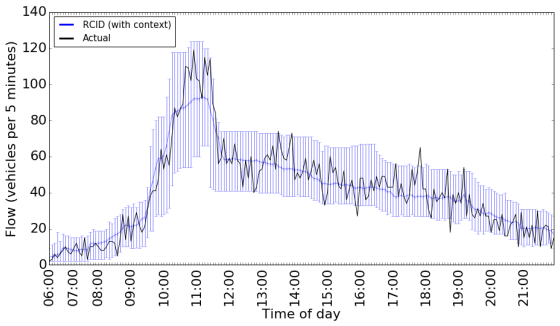
The 1.50% FAR of RCID achieved with a 90% prediction interval was within the 'acceptable' bounds of the TMC operators survey. This FAR translated to an average of 75 false alerts per day across the 111 detectors. Clearly, the number of false alerts produced by RCID would correlate with the number of detectors it was implemented on in a TMC's network. As such, depending on the number of operators in a TMC, and the size of the network being managed, it is unclear whether this rate of false alerts would be suitable in practice. An online test of the IDA would be beneficial in that it could provide context of the frequency of alerts that could reasonably be monitored in practice.

7.3.6 Analysis

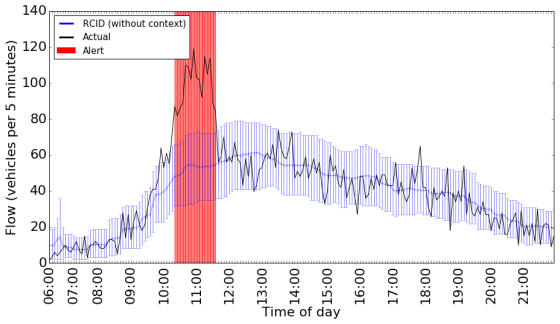
In this section, graphs of the IDA's alerts are presented in order to explain the differences in performance found in the previous section.

Figure 7.3 shows how RCID used the football context to learn what disruption could be expected, resulting in it not raising a false alert when travellers approached the stadium. RCID (without context) raised a false alert in this case because it did not accurately forecast the contexts disruption. RAID did not raise an alert because ALOTPV values did not remain above its threshold for three consecutive messages, i.e. the football match did not sufficient cause congestion for a sufficient period of time for an alert to be raised. McMaster also did not raise a false alert due

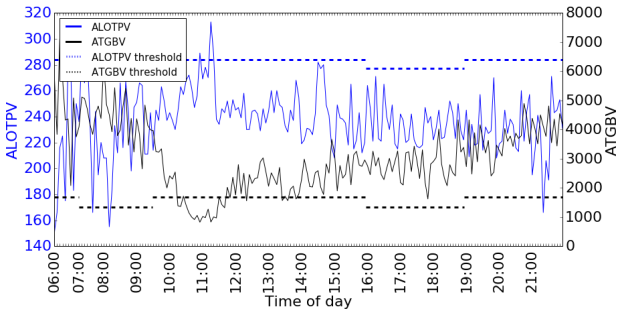
to the football match due to the flow methodology seeking values that are lower than expected (the match caused higher than typical flow values).



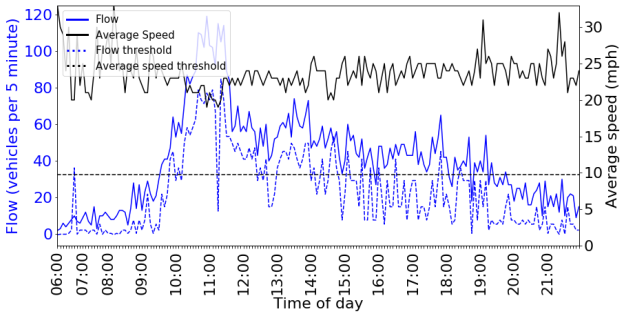
(A) RCID



(B) RCID (without context)



(C) RAID



(D) McMaster

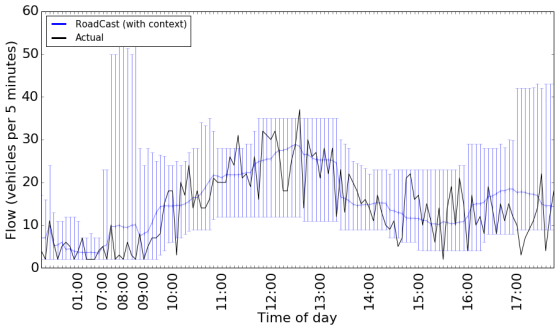
FIGURE 7.3: IDAs' alerts on the day of a Premier League Football match against Leicester F.C., which kicked off at 12:00 at St Marys Stadium. No incident occurred. Sunday 22nd January 2017, at detector B. RCID used a 90% prediction interval. RAID used a threshold of the 85th percentile of ALOTPV values, and the 15th percentile of ATGBV values. McMaster used an α value of 1.75 and β value of 5.0. It should be noted that the forecasts and prediction intervals of RCID are indicated by the blue lines and error bars respectively, and the red highlighted areas are times when the IDA raised an alert.

Figure 7.4 again shows how RCID used contexts to avoid raising false alerts. RCID (without context) raised false alerts throughout Christmas day because it could not differentiate between traffic on Christmas day and traffic on an average Sunday. RAID did not raise an alert because no congestion occurred on this day, and so ALOTPV and ATGBV were sufficiently typical. McMaster also did not raise an alert because average speed values were typical (no congestion was caused), and flow values at the given occupancy level were low because of the lack of vehicles on Christmas day, meaning that the flow threshold was sufficiently low. It is thought that the wide prediction intervals produced by RoadCast (without context) were due to the lack of data during the Christmas period (as some days in the holiday had missing data at this detector). This led to the algorithm including a range of values that included messages on other days in the Christmas holiday. To avoid this issue, either a smaller prediction interval percentage could be used, or more representative data could be used in training (i.e. more years of data or a dataset with fewer missing messages).

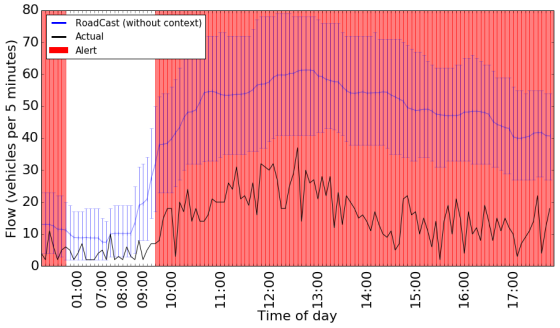
At the typical time of day and day of the week of a contexts disruption, RCID (without context) would often create prediction intervals wide enough to cover the contexts disruption, both when the context occurred and when it didnt. This occurred more often and to a greater extent when higher percentage prediction intervals were used. Figure 7.6 shows a wide prediction interval caused by the disruption from occasional weekday evening football matches. This may have caused the incident to go undetected if it occurred an hour later. Figure 7.6 also shows RAID failing to detect an incident because ALOTPV and ATGBV values were not disrupted sufficiently, i.e. the incident did not cause sufficient congestion for an alert to be raised. McMaster also failed to detect the incident due to the average speed being unaffected, and flow values being higher than typical (the threshold only detects lower than expected flow values). It should be noted that incidents with less disruption are typically of lower priority to TMCs because they typically have lesser consequences. However, such incidents are more common than incidents with major disruption. As such, even though less severe incidents are more difficult to detect, it is still important for an IDA to be able to detect them effectively.

Figure 7.5 shows an example of how RAID could raise false alerts at times when a contexts caused disruption. In this case, a football match caused the disruption to ALOTPV values. RCID (with context) did not raise an alert due to its understanding of the disruption expected from the football match.

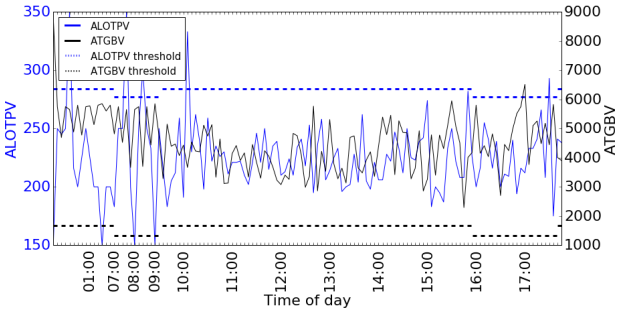
RCID (without context) often raised false alerts when contexts caused disruption (as can be seen in figure 7.3). However, in some cases it would not raise false alerts for contexts, particularly for contexts that occurred frequently at a particular time or day of the week, such as football matches at 3pm on Saturdays. As can be seen in figure 7.7, at times the prediction interval was wide enough to cover the contexts disruption, because the messages in the predicted leaves were from both times when a match was occurring and when it wasnt. With a 95% prediction interval, one could assume that if a context caused disruption at a particular time and day of the week on less than 2.5% of occasions in the training period, RCID (without context) would be susceptible to raising false alerts on these occasions in the testing period. However, such wide prediction intervals made RCID more susceptible to failing to detect incidents at times when the



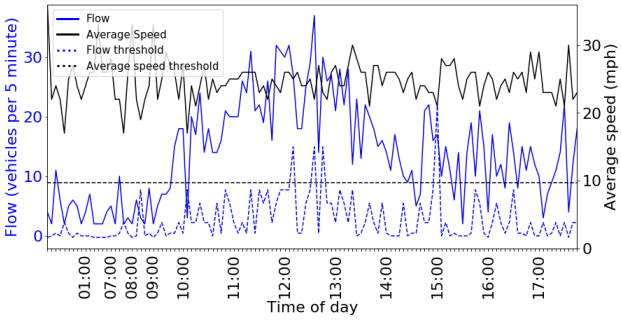
(A) RCID



(B) RCID (without context)



(C) RAID

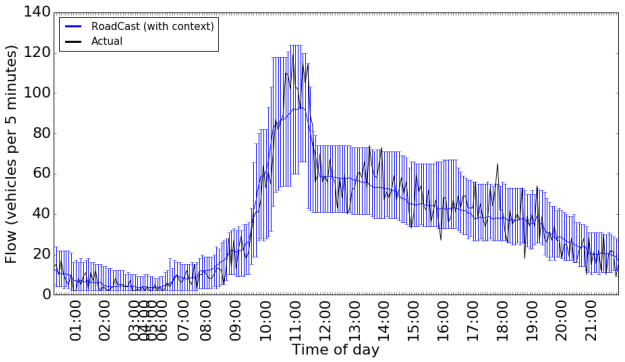


(D) McMaster

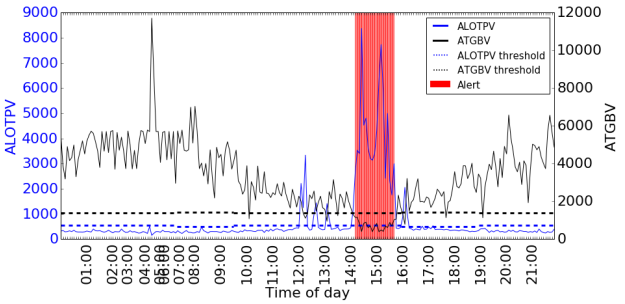
FIGURE 7.4: IDAs' alerts on Sunday, Christmas Day 2016 (25^{th} December), at detector B. RCID used a 90% prediction interval. RAID used a threshold of the 85^{th} percentile of ALOTPV values, and the 15^{th} percentile of ATGBV values. McMaster used an α value of 1.75 and β value of 5.0.

particular context does not occur.

Unfortunately, no incidents occurred at times when contexts typically disrupted traffic conditions



(A) RCID

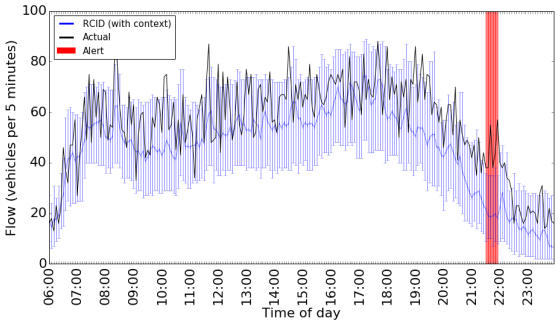


(B) RAID

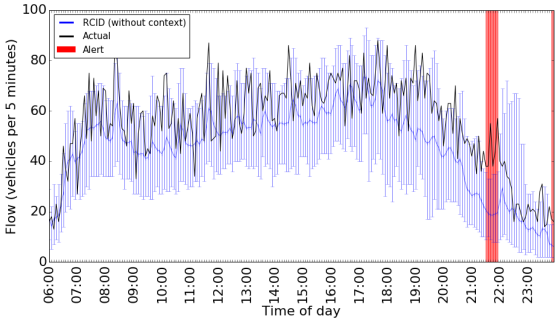
FIGURE 7.5: IDAs' alerts on Sunday, 21st December 2017, at detector A. A Southampton FC football match took place at midday against Leicester FC. RCID used a 90% prediction interval. RAID used a threshold of the 85th percentile of ALOTPV values, and the 15th percentile of ATGBV values.

(figure 7.6 shows the closest occurrence), and so the effectiveness of RCID (with and without context) to detect incidents during context caused disruption could not be seen. However RCID could be seen to produce more naive and uncertain prediction intervals when contexts were not included in general. This could be seen particularly at times when contexts typically caused disruption (i.e. the same time of day and day of week), as can be seen in figure 7.6. This resulted in RCID (without context) having a lower detection rate at each prediction interval setting. Because of the visible increase in uncertainty and naivety in RCID's prediction intervals when not using contexts, it is thought that RCID is more effective at detecting incidents when it uses contexts. It is thought that by repeating this test on a dataset where incidents occurred at times of contexts' disruption, the difference in RCID's ability (with and without context) to detect incidents during contexts' disruption would be seen.

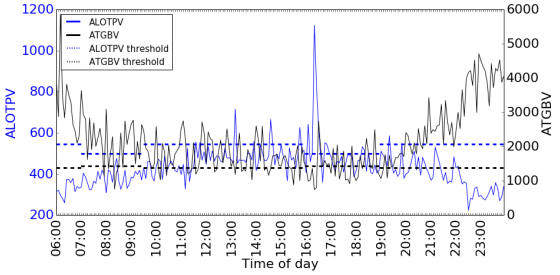
In section 7.3.5, RCID was found to produce fewer false alerts when incorporating contexts at each prediction interval setting. From the analysis, it can be seen that this improvement was due to RCID (with context) producing more accurate and more certain prediction intervals at times when contexts typically disrupted traffic conditions. Figure 7.3 shows an example of this. Such intervals were produced because of RoadCast's ability to use historical data to 'learn' traffic conditions that could be expected in the future. The intervals allowed RCID to better differentiate incidents from contexts, and hence produce fewer false alerts.



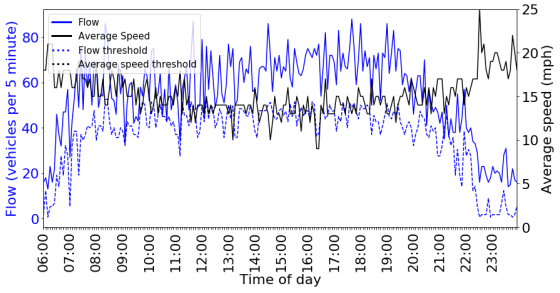
(A) RCID



(B) RCID (without context)



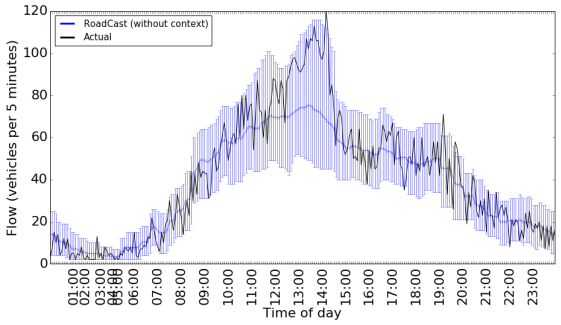
(C) RAID



(D) McMaster

FIGURE 7.6: IDAs' alerts at a time where an emergency roadworks incident caused one lane of a nearby roundabout to be closed, causing disruption between 6pm and 11pm. Thursday 15th December, at detector A. RCID used a 90% prediction interval. RAID used a threshold of the 85th percentile of ALOTPV values, and the 15th percentile of ATGBV values. McMaster used an α value of 1.75 and β value of 5.0.

Figure 7.8 shows a histogram of the number of alerts that were generated each day by RCID (with context) with a 90% prediction interval. The two days with the highest number of false alerts were 23rd December (during the Christmas holiday context), and 28th March (Easter Monday, during the Easter holiday context). This figure shows that despite RCID improving its accuracy by incorporating contextual data to forecast contexts' disruption more accurately, such days were



(A) RCID

FIGURE 7.7: RCID (without context) alerts on Saturday 15th December, at detector B. RCID used a 90% prediction interval.

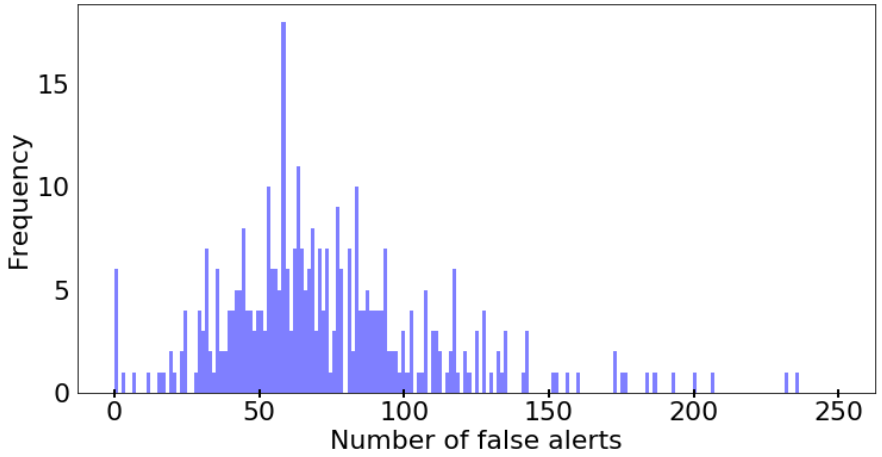


FIGURE 7.8: Histogram of RCID (with context)'s false alerts each day. A 90% prediction interval was used.

still forecasted less accurately than typical days that did not have contexts causing disruption. The first likely reason for this phenomena is that traffic conditions are typically more variable on days with contexts' disruption. For example, traffic conditions during Christmas may vary depending on the day of the week that the holidays starts or ends, and could be more dependent on the weather conditions. This factor makes forecasting such days inherently less predictable. Then second reason is that for many contexts, RoadCast had a far smaller sample of data to train on than for days without contexts' disruption. For example, only one Easter Monday occurred in the training data, but many typical Mondays occurred. This factor may have contributed to RCID producing a less representative prediction interval on such days.

In general, RAID was found to be somewhat effective at detecting congestion, as ATGBV and ALOTPV appeared to be good indicators of queues occurring at loop detectors. However, its performance as an IDA was limited in two ways; it would raise false alerts during context caused congestion, and it would fail to detect incidents that didnt cause congestion. The former limitation was due to the IDA having no means of being able to differentiate incidents from contexts, and so would raise an incident alert if congestion occurred, regardless of the cause. The latter was due to the IDA only using traffic variables that indicated congestion, and so was unable to detect incidents that disrupted other traffic variables, such as flow.

RAID was also limited by the pre-defined 'peak' and 'off-peak' times, which often did not meet their objective of accounting for the time of day variance in ALOTPV and ATGBV values. See for example the peak ALOPTV threshold in figure 7.6, which is lower than the off-peak threshold). It is thought that these pre-set 'peak' times were ineffective because different detectors had different peak times, and peak times differed on different days of the week.

McMaster was also found to be somewhat effective at detecting incidents that caused congestion, but also had many drawbacks resulting in a comparatively worse performance than RCID. Because McMaster's thresholds were based on average speed and flow at a given occupancy level, many incidents that only disrupted flow values went undetected. Another drawback of the IDA is that it only raised alerts for real-time values of average speed and flow below a threshold, and so incidents that cause untypically high values of flow went undetected. Finally, the IDA was limited in that it used a fixed threshold value for average speed, and a flow threshold value that was based only on the occupancy value. This meant that the IDA was unable to account for contextual, seasonal, weekly or daily variation in traffic conditions.

7.3.7 Summary

This section presented the results of an offline test of RCID. RCID was compared to other state of the art IDAs, RAID and McMaster, in order to understand its benefits and limitations. RCID was found to outperform RAID and McMaster in terms of detection rate and false alert rate, at all levels of sensitivity. RCID was also found to have better performance when using contextual data, both in terms of detection rate and false alert rate, due to RoadCast's more accurate predictions.

This test addressed objective two and three of this research project (see section 1.8). Firstly, RCID was implemented on a real-world traffic dataset from Southampton, U.K. Once trained on the first year of data, RCID was ran in an offline manner on the second year of data, and its alerts were compared to an incident dataset from Southampton's TMC in order to determine its performance. Other state of the art IDAs, RAID and McMaster, were also implemented and tested on the same dataset in order to gain evaluate RCID's performance. Objective two of this research project was met in this test, because an IDA that addressed a limitation of state of the art IDAs was developed and implemented.

The test results revealed that RCID was able to differentiate contexts from incidents more effectively by incorporating contextual data. RCID achieved a greater performance than the other state of the art IDAs implemented. This test satisfied objective three of this research, i.e. to evaluate RCID and make comparisons with the state of the art in order to determine whether RCID addressed the limitation identified with the state of the art, and improved on the performance of state of the art IDAs. This result also demonstrated this research project's hypothesis to be true. That is, RCID demonstrated that contextual data can be incorporated within an IDA in order to better understand traffic conditions that can be expected to occur, and hence better differentiate incidents from contexts.

Although this test was sufficient to prove this research's hypothesis, and provided perspective by comparing its performance to the state of the art, there were some areas of RCID that could not be evaluated fully. For example, it was unclear whether RCID's false alert rate would be suitable for operators to be able to manage in practice. Other factors could also not be evaluated, such as usability and calibration requirements (which were described in section 2.3.3). Given that the goal of RCID is to aid TMC's detect incident more effectively, these dimensions of RCID's effectiveness could only be concluded to be satisfactory by TMCs themselves. An online test of RCID in a TMC is required to assess RCID's suitability for use in practice.

The analysis section showed that there is still some room for improvement in RCID's performance. The presented IDA methodology was simple and sufficient to demonstrate RCID's improvement on the state of the art, and benefit of using contextual data. However, as there is still room for improvement, a number of other incident detection methodologies were developed for RCID to try to improve performance further. These methods are presented and evaluated in section C. None of these methods demonstrated an improvement on the RCID methodology presented in this section.

7.4 Limitations

In this section, limitations of RCID that were found during the initial offline test are described and explained.

A limitation of each of the IDAs in the offline test may have been the average time to detect. Unfortunately, this couldnt be evaluated quantitatively because the exact time of incidents occurrence was unknown. However, based on the persistence tests used, it could be expected to be at least 15 minutes for every tested IDA, which is higher than the reported value of many other IDAs presented in the literature. Then again, operators reported in the online test that RCID was often faster than their existing methods, so perhaps it is still an improvement over what it used in practice. This issue stemmed from the IDAs using messages over long time periods (i.e. 5 minute messages). If 30 second messages were used instead of 5 minute messages, the IDAs could have used a persistence test over a shorter time period, and hence may have been able to detect incidents more quickly. However, the trade-off of this difference is likely to be an increased amount of noise in the messages, due to the fewer number of vehicles that would be found in each message.

In general, RCIDs false alert rate was limited by inaccurate predictions from RoadCast, caused by variations in the traffic data that were not accounted for. Some causes of variation may have been missed, such as contextual events that were not identified. Other unaccounted causes of variation may not have been suitable for incorporation, such as noise or disruption during particularly busy shopping days, which could be identified (and verified by Southampton City Council tweets), but not predicted beforehand. The detection rate was most limited by failing to detect incidents that caused minor amounts of disruption. In these cases, the prediction intervals were too wide because of RoadCasts forecasting uncertainty, which stemmed from the unaccounted

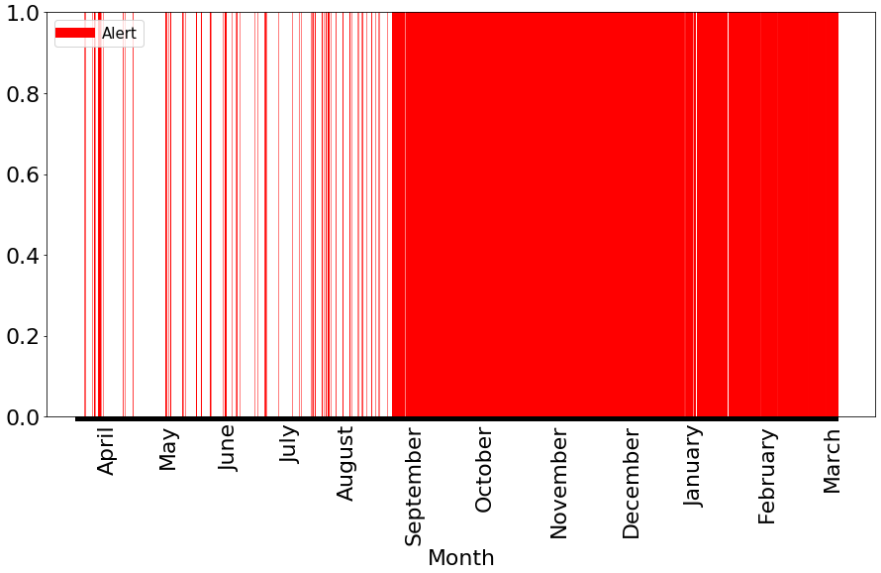


FIGURE 7.9: RCID (with context) with a 90% prediction interval, across all 365 days of the test dataset of a detector that had an 11% false alert rate.

causes of variation in the data. The causes for these limitations in false alert rate and detection rate are inspected on a lower level in the following paragraphs.

One cause of false alerts in RCID comes naturally from the choice of prediction interval. Even if the prediction intervals produced by RCID perfectly represented the distribution of flow values that could be expected, a number of false alerts would still be raised by chance. Taking a 90% prediction interval as an example, then with perfectly representative prediction intervals, actual flow values would fall outside this interval 10% of the time. With the persistence test of three consecutive messages, this means that any sequence of three consecutive messages has a 0.1% chance of raising an alert. As such, a 0.1% false alert rate can be seen as a lower bound for RCID at a 90% prediction interval. Of course RCID could use a prediction interval of 100% in order to avoid raising false alerts, but with this interval RCID's detection rate is limited because the interval covered any unrepresentative training data, and was unable to detect incidents that produced flow values that were within the range of all values in the training data with similar contextual values (i.e. in the same leaf of each tree in the random forest).

As discussed in section 6.10, RoadCast could make inaccurate predictions if there was a major change in traffic conditions at a detector during the testing or training period. This change could be caused by a change in topology, travel demand, or a detector malfunctioning. Such a change would result in accurate predictions due to the random forest model being trained on data that was unrepresentative of the of traffic conditions that could be expected in the testing period. These inaccurate predictions would lead to false alerts being produced by RCID due to the difference between real-time conditions and RoadCast's forecasts. There were at least five found examples of this in the test of RoadCast on the Southampton dataset, and 21 in Bristol. Figure 7.9 shows an example of a detector with a step change during the test dataset.

This drawback translated to RCID having a high false alert rate at the same detectors. It is difficult to tell how many detectors are affected by this phenomenon, and by how much. However, this is an issue for which all IDAs that are based on training on historical data could be expected to suffer from. However, if this issue was identified by an operator when used in a TMC, this issue could easily be rectified by retraining the IDA on data from the time period after the step change only, or removing the detector entirely. It may also be possible to identify the step change and remove the unrepresentative data automatically using an algorithm. However, detecting these values automatically can be difficult due to the diversity of profiles produced by different detectors, and the many causes of variation in traffic data. This limitation was mitigated in the online test by manually removing such detectors, but this may not be practical in real-world implementations of RCID.

Due to the difficulty in verifying whether a detector has a step change (finding the cause of the change), and the subjectiveness in deciding whether a detector's change is clear enough to warrant removal or retraining, it was decided that all detectors that appeared to have a step change would not be removed from the offline test results and analysis. However, this decision may mean that each of the IDAs could perform better in practice if TMC operators removed or retrained such detectors, or a pre-processing algorithm was implemented to achieve this automatically. In the online test, it was agreed that such detectors would be removed from the test when identified by the operators before and during the test, who had the prior knowledge of the network to identify such cases. This was agreed in order to ensure that the operators managing the TMC would be using the most effective possible IDA to help them with their responsibility to detect incidents.

7.5 Conclusions

This chapter described the development and evaluation of RCID, an IDA that aimed to address the hypothesis of this research project. That is, to address the problem of IDAs creating unnecessary false alerts by failing to differentiate incidents from contexts. Such false alerts distract operators, and has led to many IDAs being disabled or simply ignored.

RCID is a novel random forest incident detection algorithm which aims to use contextual data to tackle this problem, and hence improve on the performance of state of the art IDAs. RCID was evaluated offline, on loop detector flow data and TMC incident logs from Southampton, U.K. Comparisons were made with and without context, and to state of the art IDAs McMaster and RAID.

In the offline test, RCID was found to outperform RAID and McMaster in terms of detection rate and false alert rate. RCID was also found to reduce its false alert rate when incorporating contextual data. This improvement came from RCIDs ability to differentiate incidents from contexts by learning how contexts could be expected to disrupt traffic conditions. This result suggested that if RCID were to be implemented in a TMC, operators would be distracted by far fewer false alerts from contexts than is currently the case with state of the art algorithms. This would enable operators to detect incidents more effectively, and hence respond more effectively

in order to reduces the disruption caused.

A number of alterations to the original version of RCID were made with the aim of improving the IDA's performance. These methods included alterations to the incident detection methodology of RCID, and a number of spatial strategies, aiming to take advantage from the patterns in which incidents and other causes of variation in traffic propagate through the network. Each of these alterations did not result in clear improvements, and so the simplest version was deemed most suitable for implementation.

From the findings of this evaluation, conclusions can be drawn with regards to the hypothesis of this research project. It has clearly been shown that with the appropriate methodology, IDAs can have the ability to differentiate disruption from contexts and incidents by incorporating contextual data to better understand traffic conditions that can be expected to occur. The incorporation of such contexts improves performance, but also introduces downsides such as the time and effort required to collect and incorporate contextual data. The costs and benefits of incorporating contexts have been covered extensively in the previous sections. Based on this analysis, it is suggested that the benefits do out weigh the positives on the whole. Although time, expertise and effort was required to collect, organise and incorporate contextual data into RCID, the benefit of 2% detection rate and 0.25% false alert rate (when using a 90% prediction interval) is thought to be out weigh theses costs. Looking forward, it is thought that the benefits will likely only become greater, and the costs more minor. That is, with development and refinement of the methodology, the performance can likely improve further, and the effort required to incorporate contexts will likely lower due to automation. As such, the use of contextual data to be incorporated within IDAs in order to better understand traffic conditions that can be expected to occur is recommended.

Although many insights and conclusions could be drawn from the results of the offline test undertaken in this chapter, a number of aspects of RCID could not be fully evaluated. Aspects such as the usability of the IDA, calibration requirements and transferability to other locations could not be fully evaluated. As such, it was decided that an online test would be required to fully evaluate the performance of RCID that could be expected in practice.

Chapter 8

RoadCast Incident Detection online test

8.1 Introduction

As was described in section 7.3.7, the offline test of RCID was found to be insufficient in evaluating all aspects of the IDA. For example, factors such as the usability and calibration requirements of RCID could not be evaluated fully. As such, an online test of RCID is seen as necessary. The aim of this test is to understand the performance and usability of RCID that could be expected in a TMC in practice. This test also aims to help understand the best role that RCID could play in order to benefit an operator the most.

This test will involve implementing RCID in an online manner, i.e. to detect incidents from a real-time feed of traffic data, and to be used by operators in a TMC. TMC operators will be presented with RCID's alerts in real-time, be able to use the alerts to detect and respond to incidents, and provide feedback on the alerts for the purposes of this research study. Operators will also be interviewed before and after the test in order to understand whether RCID is suitable and useful for use in TMCs in practice.

8.2 Test details

Bristol City Council Traffic Signals Operations Centre agreed to host this test, and gave permission to obtain and use the training and real-time traffic data required to implement RCID. Bristol's TMC was seen as a suitable place to test RCID given the size of the city, the 591 loop detectors available, and the diverse contextual factors that disrupt traffic across the city. The TMC has a team of five staff that actively monitor the network between the hours of 7am-7pm Monday to Friday and 9am-5pm on Saturday. One or two operators actively monitor the network at any one time. Part of each of their responsibilities is to detect and respond to incidents as they occur. All five TMC operators would be involved in the online test of RCID. The test would

run for the month of April, during the hours in which the operators would actively monitor the network.

Unfortunately in this test, the commonly used performance metrics DR, FAR and MTTD could not be used to evaluate RCID. This was because no ground truth of the times and locations of incidents was available. As such, feedback from each alert raised, and post-test operator interviews would be relied upon to evaluate RCID in this test.

A key consideration of this test was the interface in which RCID's alerts would be presented to the TMC's operators. This interface would not only be required to display RCID's alerts in real-time, but must also be able to collect feedback from operators. Given the previous literature reporting that many IDA's have distracted from operators other responsibilities (see section 3.2.5), an interface which reducesd the effort required to give and receive feedback was sought after. A factor in this decision was that at the time, Bristol's TMC operators used a traffic management web application developed by Siemens to monitor the network. As such, they were already familiar with the functionality of a web application that displayed and received traffic information. Based on this, it was decided that the most suitable interface would be a web application.

It was decided that the initial version of RCID, evaluated in section 7.2, would be the methodology to be implemented in the online test. This was chosen due to its simplicity to implement, and performance in comparison to the alternative methodologies tested in appendix C.

It was decided that the implementation procedure of RCID in the online test would be as close as possible to the offline test, in order to be able to compare the results of both tests, and make conclusions from the findings of each. The only differences in the online test were that a single prediction interval would be decided on before the test and used throughout, and detectors that were reported by operators as unrepresentative would be removed. Unrepresentative detectors were those that were a distraction to the operator due to the number of false alerts being produced, either due to the detector being faulty or having a step-change in the training data. In order to do this, it was agreed that a single operator would use and give feedback on the application before the test. Based on the feedback, 46 detectors were removed, which are described in the following section. This feedback was also used to decide upon a prediction interval to be used in the online test. 95% was agreed upon. It should be noted that with this prediction interval, RCID (with context) achieved a 92% DR and 1.0% FAR in the offline test, which was within the acceptable bounds of Ritchie and Abdulhai (1997)'s TMC operator survey. Other than these changes, RCID was implemented in the same was as was conducted in the offline test.

8.3 Data

As described in section 8.2, Bristol City Council's TMC agreed to host the online test of RCID, and allowed the required traffic data to be used. The following sections describe the data sources

used in the online test of RCID.

8.3.1 Traffic data

8.3.1.1 Location

Similarly to Southampton, the problem of congestion comes at a significant cost to Bristol, again predominantly in the form of losses in productivity and fuel costs. Bristol drivers spent an estimated average of 27 hours in delays in 2016 (three more than Southampton), resulting in a cost to the city of £154 million, or £845 per driver (Cookson and Pishue, 2016). Bristol was ranked as the 10th most congested in the U.K., eight places worse (more congested) than Southampton.

591 single inductive loop detectors around Bristol were used to collect traffic data for the study. Figure 8.1 shows the location of the detectors used.

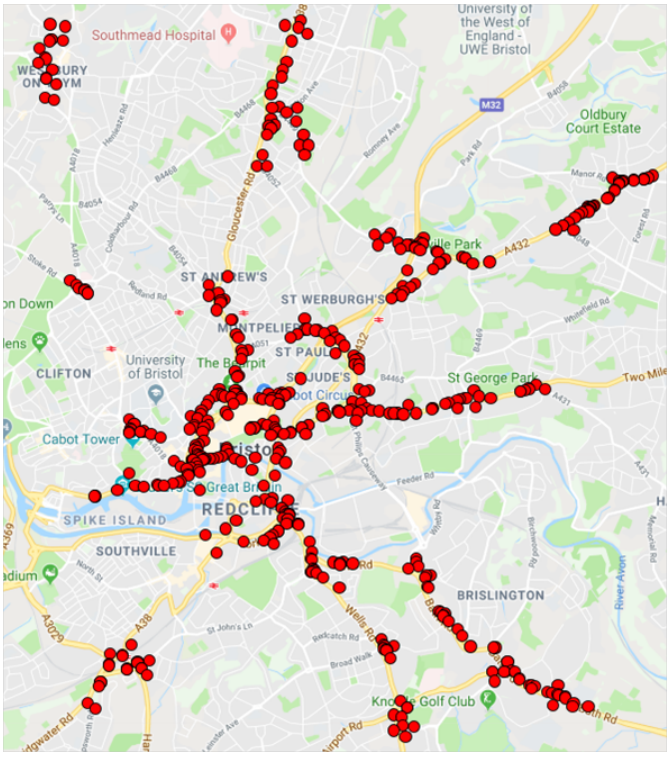


FIGURE 8.1: Locations of the detectors used in this online test. This image was created with Google Maps.

The detectors used in this study were located on a range of urban road types, including urban arterials, streets and junctions. As stated in section 2.1, urban networks are more complex than other types of networks such as motorways, and so are thought to be more challenging to forecast in. Similarly to Southampton, these detectors were installed primarily for the purpose of improving traffic signal management strategies, and so were typically installed on the stop-lines, approaches and exits of signalised intersections. As such, there may be a bias in this urban traffic dataset towards detectors in close proximity to signalised intersections.

8.3.1.2 Data description

375 days worth of loop detector data was collected from 1st October 2017 to 10th October 2018. All of this data would be used to train RoadCast in an offline manner.

RCID would be tested online by detecting incidents from a real-time traffic data source. This data source was provided by Siemens Mobility Limited in the form of a web application. Siemens' online database collects traffic data messages from the detector's UTC system in real-time. The most recent traffic data messages from each detector are taken from this online database and displayed in the web application. This web application was used as the real-time traffic data source for the online test of RCID. Data from this web application was collected in real-time using HTTP requests.

Both the training data and live feed provided five minute messages of flow that were processed by the same Urban Traffic Control (UTC) system. These messages were calculated from the detector's 1s and 0s in the same way as in the Southampton dataset, as described in section 5.3.2. As was the case with the offline test, RCID would use flow as the only target variable.

8.3.1.3 Pre-processing

Pre-processing was deliberately conducted in the same way as the offline test (see section 5.3.3). All messages with a flow value of zero were removed, along with the previous and next message (because a detector could start/end returning unrepresentative data at any point during a five minute period). After this, detectors which had fewer than 50 training messages or 50 testing messages were disregarded. The reasons for these choices were described in section 5.3.3. Of the 591 detectors present in the network, 456 remained after the above pre-processing steps.

As stated in section 8.2, an operator used the web app before the test started in order to identify a number of unrepresentative detectors. Some of these unrepresentative detectors were faulty, but did not produce values of zero flow continuously. As such, these detectors produced a sufficient number of messages with non-zero values of flow to not be removed based on the pre-processing steps above. However, the data produced could be identified as false by the operator comparing the traffic profile to CCTV imagery over the detector. Based on this feedback, 25 detectors were removed on the basis of being faulty.

Other detectors appeared to have a step change, an issue described in detail in section 6.10. These detectors were identified using feedback from the operator before the test. They could also be observed by observing traffic profiles in the training data. These detectors produced many false alerts, and so were a distraction to operators. As such, 21 detectors that appeared to have a step change were excluded. After these detectors were removed, 410 remained, which would be used in the online test.

8.3.2 Contextual data

The collection of contextual data for the online test of RCID in Bristol was conducted in a similar manner than for the offline test in Southampton. However, Bristol's TMC provided a spreadsheet of events that were known to cause disruption in Bristol. This spreadsheet was used as the starting point to identify contextual factors local to Bristol. The spreadsheet identified the cause, magnitude, and location of events that disrupted traffic in Bristol. Using this spreadsheet along with searches of the internet, data for contextual factors were identified. Data for these contexts were collected manually and with web scrapers.

The standard encoding methods described in section 5.7.6 were used to encode Bristol's contextual data into features used by RoadCast. Table 8.1 describes the name and type of each feature used in the online test of RCID in Bristol.

Feature type	**Standard encoding method**	**Features used in Bristol**
Time of day	Time of day	Hour of day + (minutes/60)
Day of week	Integer ranging from 0 to 6	Day of week
Modified day of week (used when a multiple day event (with reference) feature is included)	7 if during a multiple day event (with reference), integer ranging from 0 to 6 otherwise	Modified day of week
Single day events	The number of days + (hours/24) + (minutes/1440) to the nearest start time of an occurrence of the event (note times before matches are negative) if on the day of an event occurrence, 10 otherwise	Bristol City football matches, Bristol Rovers football matches, Bristol Bears rugby matches, Downs music festival, 10km and half marathon running races, charity cycling event, St Paul's carnival and a political protest.
Multiple day events (without reference)	The number of days + (hours/24) + (minutes/1440) to the nearest start time of an occurrence of the event (note times before matches are negative) if during an event occurrence, 10 otherwise	Easter, other public holidays, Harbour festival, Balloon fiesta.
Multiple day events (with reference)	The number of days + (hours/24) + (minutes/1440) to the nearest reference time of an occurrence of the event (note times before a reference time are negative) if during an event occurrence, 10 otherwise	Christmas.

TABLE 8.1: Features used in the online test of RCID in Bristol.

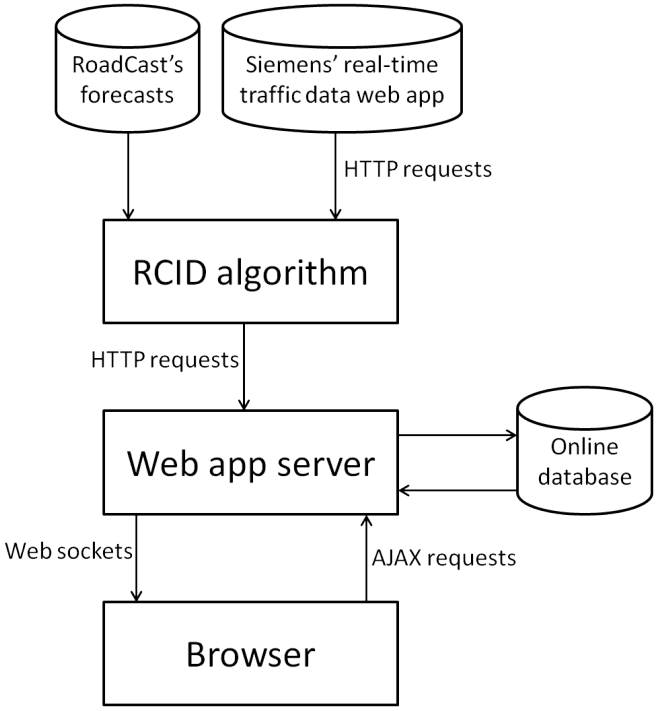


FIGURE 8.2: Architecture diagram of the web application developed for the online test of RCID.

8.4 Web application design

The design of the web application would be key to ensuring that operators could effectively be informed by and provide feedback on RCID's alerts. This section describes the design decisions behind the web application. Figure 8.2 shows the architecture of the web application.

First, RoadCast was trained on the historical data in an offline manner on a University computer, producing and storing prediction intervals for the time period covering the test. The next part of the application ran in real-time. The University computer would receive real-time traffic data from the Siemens' web app, and RCID would compare this data to RoadCast's forecasts in order to detect incidents. When an incident was detected, data related to the incident was sent in JSON format from the University computer to the web application server using a HTTP request. Data fields included the ID of the loop detector, the detector's coordinates, a location description, the start time of the alert, and end time of the alert (which would have the value 'Ongoing' when the alert was first raised). Data was then sent from the backend to the server twice for each alert, once to signify the start of an alert, and once to signify the end of the alert, where the 'end time' field would be updated from 'Ongoing' to the time at which RCID stopped raising the alert. The alert data was then stored online in a JSON file.

Web sockets were used to transfer alerts from the JSON file to the browser. Web sockets allowed alerts to be displayed on the browser asynchronously, i.e. as soon as they were received on the server, without the need for users to refresh the page. AJAX requests were then used to transfer feedback data from the browser to the JSON file.

Figure 8.3 shows the web application interface, as was viewed by the TMC operators.

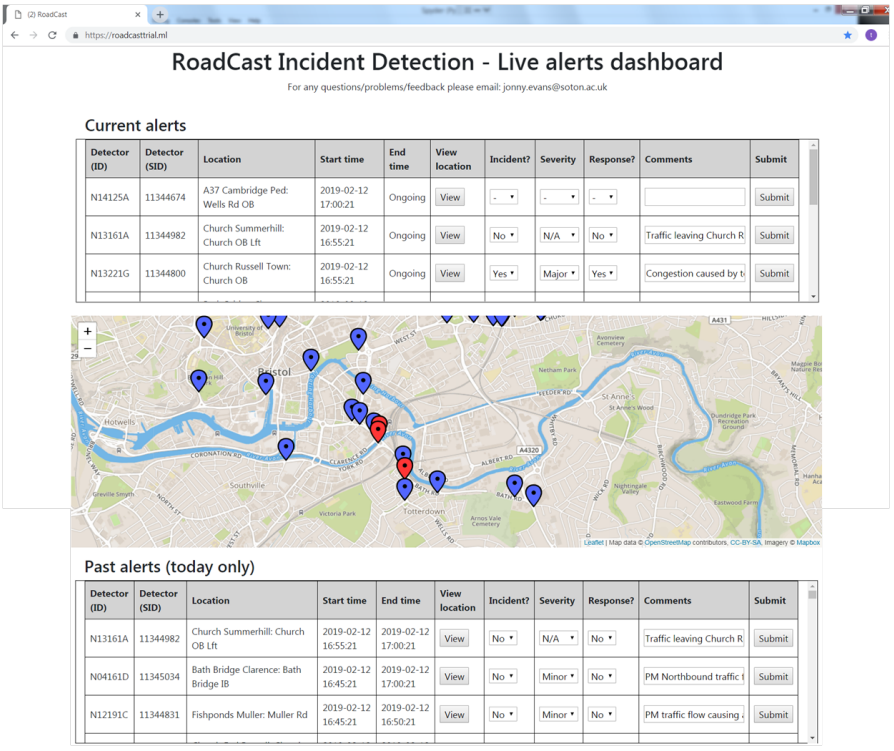


FIGURE 8.3: Screenshot of the web application developed for the online test of RCID.

The application consisted of three main section; a table of the alerts currently being raised by RCID, a map of detectors that were/had raised alerts, and a table of past alerts that had been raised on that day. It should be noted that feedback submitted by operators would remain in the table row, allowing operators to amend feedback submitted, or add to other operators' feedback. The lower table existed to allow operators to create or amend feedback for alerts that had recently ended.

A key consideration of the web app was the type of feedback that the TMC operators would be asked for. The aim of this feedback was to gain insight into the effectiveness and usability of RCID in practice. The first field in the feedback form asked operators to verify whether the alert corresponded to a nearby incident disrupting traffic at the detector, named 'Incident?'. The field was a drop-down menu with two options; 'Yes' and 'No'.

The second field asked whether the incident had already been detected by another method, named 'Known?'. This field would help to understand the benefit of the IDA to the TMC in comparison to their existing incident detection methods. Options were 'N/A' for the case that the alert did not correspond to an incident, 'Yes' and 'No'.

The next field asked about the severity of the incident, named 'Severity'. Options were 'N/A', 'Minor' for incidents that only affected the link on which the detector was located, and 'Major' for incidents that affected multiple links.

The fourth field asked whether the alert resulted in the operator making a response, named 'Response?'. Options were 'Yes' or 'No'.

Finally, a comments field was included to allow operators to note anything else on the alert. Operators were instructed to note what response was made (if any), how long RCID took to detect the incident (if known), an explanation of the cause of the alert (if known), and to note when a detector was faulty.

It should be noted that all fields and information required were explained to the operators in full before the trial started, and a description of the feedback required was included as a tooltip on the header of each table in the web application.

Another key feature of the web application was the map of detectors that were raising an alert. This map was created after consultation with TMC operators before the trial, who indicated that a map would be useful to locate incidents from the alerts. With the map, TMC operators could check the location of detectors that were raising alerts, allowing them to verify and respond to incidents more effectively. In red are detectors that are currently raising an alert, and in blue are detectors that have raised alerts at some point during the current day. When the 'View' button is pressed, the map zooms in on the relevant detector, and displays its ID and the number of alerts it has raised so far on that day.

8.5 Pre-test operator interviews

Interviews were conducted before and after the test commenced with each of the five operators in Bristol's TMC. The pre-test interviews aimed to understand what was required of RCID in order to be useful to the TMC. The interviews were conducted shortly after the operators were presented with information on the incident detection algorithm, web application and test that would take place.

Questions involved understanding what existing incident detection methods were in place before the trial, understanding how an IDA would best benefit Bristol's TMC in the task of incident detection, and what specifically the IDA would need to achieve in order to be suitable. These questions somewhat follow on from the questions asked in the TMC interviews conducted in section 3.3, but focus specifically on the RCID methodology and how it could benefit Bristol's TMC. The interviews were conducted with each operator independently in meeting rooms at Bristol's TMC. The interviews were based off of a set of questions, stated in appendix D.

The first finding of the pre-test interviews was the incident detection methods employed by Bristol's TMC at the time. There was unanimous agreement that the most commonly used method to detect incidents was by monitoring real-time CCTV footage across the network. Another commonly used method was to use various online sources of information. This included websites that display maps of incidents such as INRIX, TomTom and Google Maps, and social media feeds

such as TravelWest's Twitter feed. The operators also detected incidents by taking in reports from other local council services, including Avon and Somerset Police and local bus companies which took reports from their drivers. Finally, one operator reported to investigate the real-time traffic data graphs (such as loop detectors' average speed and flow), and make comparisons to previous weeks data in order to detect incidents.

Next, operators were asked how they responded to incidents that had been detected. The first response taken by operators was to log information on the incident on to Siemens' online traffic management software, STRATOS. This log allowed Bristol City Council managers to retrospectively analyse the number of incidents taking place in the city, and the responses that were being taken. The software also automatically formatted the logs into tweets, ready for operators to post on the Travel West twitter feed if necessary. Another common response was to inform INRIX, who could then pass the incident information to local radio stations to inform the public. The operators could also update Variable Message Signs (VMS), and alter traffic signal strategies in order to reduces the disruption. If necessary, the operators could also inform local bus services and Avon and Somerset Police. Finally, for incidents that caused sufficient news coverage, operators were at times required to inform the Council's senior managers about the incident, so that they could make a public statement to the media.

Many of the operators were not aware of any IDA ever being used at Bristol's TMC. However, one operator noted that the UTC system used in Bristol has the IDA INGRID built in (reviewed in section 2.4.1.5), but the TMC chose not to use it due to the TMC Manager deeming that it produced too many false alerts to be useful. Another operator noted that they had seen presentations of IDAs in the past, but no system was mature enough to be implemented. Finally, one operator reported that two years ago they had investigated a number of systems that reported the ability to detect incidents. These systems were developed by TomTom, INRIX and Waze. The operator found that they were only able to detect congestion, rather than incidents. They also found that the existing manual methods used in Bristol's TMC were able to detect this congestion more quickly than any of the systems investigated.

The next question asked operators about how RCID could best help improve the TMC's abilities to detect incidents. A common first response made was that RCID could make the most of their SCOOT and ANPR traffic data, which could not be fully monitored by the operators manually. As stated previously, one operator manually investigated this traffic data by observing graphs of real-time and historical data. They reported that they were only able to do this rarely over a small number of detectors in the network because of the vast amount of data available and the short amount of time that operators could afford to spend on this task. Hence, it was thought that RCID could use this data as an additional source of information for the TMC to detect incidents. Another common response was that RCID would be beneficial if it could detect incidents more quickly than the TMC's existing methods. It was reported that the existing methods let operators detect a sufficient proportion of incidents in the Bristol network, but an improvement in the speed of detection would allow operators to respond more effectively. Multiple operators suggested that RCID should be the method that first indicates the presence of an incident, and operators could check its alerts using CCTV for verification. Finally, it was suggested that RCID

would be beneficial if it could free up the time of operators which was spent undertaking manual methods, such as monitoring CCTV, i.e. to reduce the operators' reliance on labour intensive manual methods.

The final question asked operators which features were most important for RCID to have, and whether there were thresholds for which they would deem performance to be acceptable. The answers to this were more varied than the previous questions. One operator stated that the detection rate would be the most important feature for RCID to have, given that the average time to detect and false alert rate were within reasonable bounds, which were stated as a false alert at most once every half an hour (across the entire network), and an average time to detect of at most 15 minutes. The next operator agreed that the detection rate was most important, and stated that it was more important to detect all incidents that occur than it was to have a low false alert rate. They stated that any false alert rate below one per minute would be acceptable, the detection rate should be at least 50%, and incidents should be detected with 10 minutes. The next operator stated that detection rate, false alert rate and average time to detect were all important features for RCID to have and noted that the performance would be a trade off of each. Acceptable performance was stated as at most 80 false alerts per day, at least 90% detection rate and at most 15 minutes average time to detect during peak periods and 30 minutes during non-peak periods. Finally, one operator stated that RCID's detection rate should be at least 80%, and at most 30 minutes average time to detect, but could not decide on a reasonable false alert rate until they had used the IDA.

To summarise, the operators felt that RCID would be beneficial if it could effectively take advantage of the existing traffic data produced in the network to detect incidents, because this data source was not being used by the existing incident detection methods at the time. It was thought that the most effective role RCID could play would be to be the first indicator of incidents (i.e. faster detection than their existing methods), even at the expense of having a worse false alert rate or detection rate than other methods, because the alerts produced could be quickly verified using CCTV. The median acceptable bounds stated for RCID were at least 85% detection rate, at most 64 false alerts per day (across the entire network), and at most 15 minutes mean time to detect. It should be noted that the acceptable bounds stated by Ritchie and Abdulhai (1997) were at least 88.3% detection rate and at most 1.8% false alert rate.

8.6 Alert feedback findings

In this section, the feedback given by operators on RCID's incident alerts during the test period is analysed. The aim is to understand how RCID's alerts benefited and hindered the operators.

Of the 3,221 alerts raised in the study period, 1,809 were raised during the hours in which the network was actively being monitored by operators. Unfortunately, the website was faulty on the 6^{th}, 27^{th}, 28^{th} 29^{th} of April, and so no alerts were displayed during these days.

Of the 1,809 alerts raised during monitoring periods, only 175 received feedback. The reason for this was that only one or two operators were present at the TMC at any one time (due to shifts and operators being on leave), and that active monitoring of the network is only one of many responsibilities held by the operators. As such, many alerts were missed simply due to operators being busy with other responsibilities. Some operators reported that short duration alerts (e.g. five minutes) were missed due to being busy with other responsibilities.

175 of the alerts received feedback, and 75 of these were found to correspond to actual incidents, as fed back by operators. Given that this is a very different way of defining detection rate than has been used in the offline test of this study, direct comparisons to the offline test's detection rate are not possible. However, in combination with the findings of the post test interview (in the next section), an appreciation of the detection rate in this online test, and hence RCID in practice, may be possible.

Table 8.2 shows the feedback given on these 75 incident alerts.

	Yes	No	Unanswered
Major?	27 (36%)	41 (55%)	7 (9%)
Known of?	57 (76%)	11 (15%)	7 (9%)
Response made?	49 (65%)	18 (24%)	8 (11%)

TABLE 8.2: Feedback made by Bristol TMC's operators during the trial, presented as the count and percentage.

The majority of the alerts were for incidents that only disrupted traffic conditions across a single link. Over three quarters of the alerts were for incidents that were known of beforehand. However, the 11 that were not demonstrate the benefit of using RCID over the existing methods at Bristol's TMC. Finally, around two thirds of the alerts were reported to have elicited a response from operators. This is a surprising finding given that many of the alerts were for incidents that were known of already. After conversations with operators near the end of the trial, it appeared that some operators had understood this field to mean 'did I make a response to this incident?' whereas others correctly understood the field to mean 'did this particular alert elicit a response to this incident?' However, if it still holds that RCID's alerts did result in responses being taken, this shows the impact that RCID had on Bristol's TMC.

Some of the incident alerts had feedback given in the comments box describing the reponse taken by the operator. Table 8.3 shows these responses. It should be noted that of the alerts for which a response was made, only a fraction included information of the method of response. It should also be noted that some alerts had multiple responses made. The most common responses were changing VMS signs and altering signal timing strategies. This table demonstrates the impact made by RCID at Bristol's TMC in the period of the trial.

Response	Count
VMS changed	17

Altered signal timing strategies	8
Informed INRIX	2
Tweet on local traffic information Twitter account	1
Intervention on STRATOS	1

TABLE 8.3: Responses made to RCID's alerts by Bristol's TMC operators during the trial.

Clearly an important part of this online test was to understand RCID's effectiveness at times when contexts were causing disruption. Figure 8.4 shows the number of alerts produced by RCID on each day of the test. It should be noted that the days in which no alerts raised were caused by the website being unavailable. The number of alerts produced each day remains close to the mean of 87 alerts for the most part. However, there is a clear increase in the number of alerts produced during the Easter holiday, which was from the 19th to the 22nd April, particularly on Easter Monday on the 22nd. Although it could be argued that more incidents occur during the Easter holidays, the clear increase in alerts produced indicates that RCID had a higher false alert rate during Easter than on typical days.

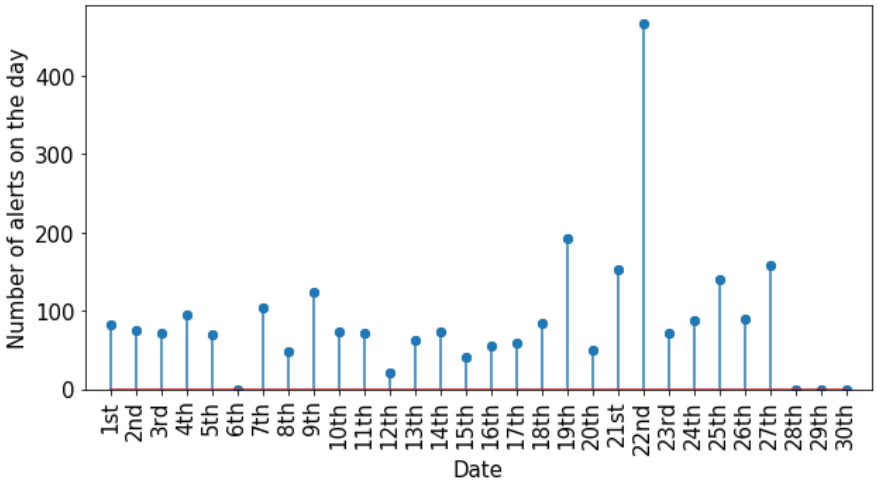


FIGURE 8.4: The number of alerts raised by RCID on each day of the online test.

The offline test results showed how traffic conditions during Easter could be predicted more accurately by including contextual data. However, despite this increase in accuracy, the conditions on these days were still less predictable than on typical days, as was described in section 7.3.6, due to the increase in traffic condition variability, and the fewer representative messages available to RCID. This reduction in accuracy likely resulted in a greater frequency of false alerts, but to a lesser extent than would be expected had contexts not been included.

8.7 Post-test operator interviews

The post-test interviews aimed to understand the extent to which RCID aided the TMC in the task of incident detection, how it was useful, and how it could have been more useful. Similarly to the pre-test interviews, the post-test interviews were conducted with each operator independently in meeting rooms at Bristol's TMC. In this case, four operators were interviewed due to one being on annual leave. The interviews were based off of a set of questions, stated in appendix D.

Firstly, an understanding was gained into how the RCID web app was being used by operators. When detecting incidents, most operators would check CCTV while keeping an eye open for a notification from the RCID web app. When a notification was raised, the RCID website would be checked, the alerts location would be discovered by using the inbuilt map, then the alert would be verified by observing CCTV at the location. After the incident had been verified or deemed a false alert, any response would be made if necessary, and then feedback on the alert would be written on the RCID website. One operator noted that they would use the RCID web app alongside Siemens' traffic management software in order to detect incidents, and another used it along with information from TomTom.

Next, operators were asked about the ways in which RCID algorithm was useful to detect incidents (if any). All operators stated that RCID had the benefit of identifying the specific location in which an incident was causing disruption, rather than just a description of a road or area. It also had the benefit of being able to detect incidents in locations that did not have CCTV coverage or locations where CCTV cameras typically were not checked. This allowed the TMC to detect more incidents than would have been detected by their existing methods. Three operators also noted that RCID would typically detect incidents more quickly than their existing methods.

Next, operators were asked about ways in which RCID could have been more useful. Every operator was unanimous in mentioning that RCID tended to raise multiple alerts for the same incident in a short period of time, and that many nearby detectors could raise alerts for the same reason (typically during incidents). This would occur typically at detectors with noisey data and at times when the trend of traffic flow was very close to the edge of a prediction interval. Due to the persistence test, alerts could be raised at each detector as often as once every 20 minutes (one message for an alert being stopped, 15 minutes for the persistence test to raise the next alert). This was a hassle for operators given that they were needed to give feedback for each alert, even though this feedback would be often be the same in these cases. It was also noted that the web app would be improved if there was a more clear notification that an incident was occurring, because often the operators would be using a different window and so could not notice RCID's notifications on the tab, which led to some alerts only being noticed long after they had been raised. One operator stated that the RCID system was too time consuming given the number of alerts raised, the time needed to verify each alert, and the time needed to provide feedback (feedback would not be necessary outside of the trial). Hence the operator identified that RCID could be improved if it produced fewer false alerts. Another operator suggested that

it would be useful if the algorithm could indicate which detector first raised an alert for a given incident, then indicate how to disruption spread to nearby detectors. This would allow operators to respond more effectively e.g. give more clear messages to the public via Twitter, INRIX or VMS signs.

Operators were then asked about whether they felt that the detection rate, false alert rate and mean time to detect were sufficient in order to be useful. The answers to this question were mixed. Two operators stated that the right balance between detection rate and false alert rate was achieved, and that the rates were sufficient for the IDA to be useful. Two operators stated that the false alert rate was too high at times to be usable, and one stated that on occasion there was an incident which was not detected by RCID, typically for incidents which caused small (or no) disruption. All operators stated that the mean time to detect was typically faster than their existing incident detection methods, except for when incidents were observed taking place on CCTV.

The next question asked operators to compare RCID to Bristol TMCs existing incident detection methods. A variety of different comparisons were made in answer to this question. One operator stated that RCID gave the specific location that an incident was causing disruption, rather than just the name of a road or area, which was a benefit not provided by many of their existing methods. Another stated that it was a useful complement to other methods, in particular because it took advantage of a data source (loop detectors) that the existing methods did not use. Finally, one operator stated that they primarily detected incidents using CCTV monitoring, but RCID allowed them to only monitor CCTV cameras when RCID alerts were raised, and only at those locations, which meant that the time and search space of CCTV monitoring was largely reduced.

Next, operators were asked whether they would choose to use RCID in their daily work in the future. One operator stated that they would find it useful, and would use it occasionally but not regularly until the false alert rate had been lowered, and fewer alerts were raised for the same incident. The other three operators each stated that it was useful enough during the trial to be used regularly as part of their responsibility to detect incidents on the network. Two of these operators stated that they would be inclined to use it more if it was incorporated within Siemens traffic management software, so that fewer websites would need to be checked. One operator added that they would be more inclined to use RCID after the trial because they would not be required to fill in feedback for each alert.

Finally, operators were asked whether they had anything else to add. At this point, two operators stated that in general, RCID was a useful addition to their existing incident detection methods. One operator stated that they had conducted a review of real-time sources of incident information, including TomTom, Waze, a police website that displayed traffic disruption alerts, and INRIX. They stated that RCID was more effective than these methods, principally because it appeared to detect incidents more quickly.

8.8 Limitations

In this section, limitations of RCID that were found during the online test are described and explained.

In general, the aspect in which the performance of RCID could be improved the most was the false alert rate. Clearly, the false alert rate, detection rate and mean time to detect have trade-offs that can be made by altering the sensitivity of the algorithm. Hence, in the offline test, the most suitable parameter settings were set to make RCID most suitable for the test. In this case, it was thought that if the sensitivity were to be changed to improve the false alert rate, the mean time to detect or detection rate would have to be changed so as to make the IDA unsuitable. However, despite the improvement made by incorporating contexts, and the tuning of parameters to alter sensitivity, false alert rate was the biggest limitations found by operators. As such, it is thought that further improvement in this aspect of performance could create the most benefit to IDA use in TMCs.

The post-test operator interviews highlighted an issue related to the number of false alerts produced by RCID. Many operators identified that for a given incident, many different detectors would raise alerts, and many would raise alerts multiple times. This was a distraction for operators that ultimately worsened the user experience, and hence made RCID less usable in practice. This effect is a symptom of the fact that RCID is trained and implemented on detectors independently. To rectify this, it was suggested that alerts could be grouped together, such that operators would only be displayed with one alert per incident, and that that alert could be investigated to understand the details of it e.g. which detectors it was affecting and when. This improvement could either be made by altering the methodology of RCID itself, or simply by re-designing the user interface.

A number of improvements were suggested to improve the method in which RCID's alerts were displayed during the online test. This included the suggestion of indicating which detector first raised an alert, and to display the evolution of the disruption amongst affected detectors. It was also suggested that operators should have been more clearly notified of alerts when they arose, such as by a pop-up or noise notification.

8.9 Conclusions

In this chapter, the findings of an online test in which RCID was implemented and evaluated in Bristol's TMC were described. The aim of this test was to understand the performance and usability of RCID that could be expected in a TMC in operation. An additional aim was to understand the role that RCID can play to be as useful as possible to the TMC operators. These aims were fulfilled by analysing the real-time feedback provided by the operators on RCID's incidents alerts, and from the pre-test and post-test interview responses.

The simple takeaway from the pre-test interview responses was that RCID would benefit the TMC if it could provide a benefit that was not provided by the TMC's existing incident detection methods. The most commonly suggested method of achieving this was that it could be the first method to indicate the presence of an incident (i.e. quicker than the existing methods), and then operators could verify the alert by monitoring CCTV. The median acceptable bounds stated by the operators was 85% detection rate, at most 64 false alerts per day (across the network), and at most 15 minutes time to detect.

The real-time feedback from the online test showed that many alerts corresponded to incidents, and resulted in responses such as changing VMS signs and signal timing strategies, demonstrating the impact and and benefit that RCID has during the test. It was also found that RCID produced more alerts during a day with a large amount of context disruption, Easter, than it did on a typical day.

The key message from the post-test interviews was that each of the operators found RCID useful, and three of the four would choose to use it in their daily work after the trial. This is in contrast to many real-world tests of IDAs in the past (as was found in section 3), and demonstrates the potential industrial impact of this research project. In particular, three operators found that RCID detected incidents more quickly than their existing methods, and hence many used it as the first indicator of an incident before verifying using CCTV imagery, as expected. The main limitation of RCID's usability in the trial was stated as the false alert rate, and in particular the issue of raising multiple alerts at multiple detectors for each incident. Despite the improvement made by the incident detection methodology design, and the incorporation of contextual data, it appears that IDAs would benefit from further progress in this regard.

The findings of this test also have implications for the transferability of RCID to different locations. In both Bristol and Southampton, it has been clear that RCID has shown competency in differentiating incidents from contexts, and has shown the ability to improve performance by incorporating contextual data. Of course, to make conclusions on RCID's location transferability more confidently, implementing RoadCast in more locations would be beneficial.

With regards to objective three of this research project (see section 1.8), this test's results demonstrate that RCID is able to perform effectively in practice in a TMC, as well as in offline tests on historical data. The findings of the tests were also used to address objective four of this research project, i.e. to provide recommendations on details such as where limitations exist, opportunities for improvement and implementation requirements.

Chapter 9

Contributions and conclusions

Since the 1970s, IDAs have been developed to aid TMC operators detect incidents more effectively. However, operators have found that these IDAs produced too many false alerts to be usable, and so have disabled or simply ignore or them. In the literature, IDAs' high false alert rates are most often reported to be caused by a failure to differentiate between disruption from incidents and contexts.

This research project aimed to improve IDA performance by enabling an IDA to make the distinction between incidents and contexts more readily. First, this chapter will present a summary of the work presented in this thesis, highlighting how each of the objectives have been met. Then the chapter will investigate the contribution made by this research project, and highlight the limitations and opportunities for future work made.

9.1 Summary

This section highlights the key conclusions of this research, and describes how each of this research project's objectives were fulfilled. These objectives were:

1. Improve the understanding of the 'state of the art' in incident detection, and highlighting any limitations and opportunities for improvement.
2. To develop an IDA that is able to address one or more identified limitations of state of the art IDAs.
3. To evaluate the developed IDA and make comparisons to the state of the art in order to determine whether the IDA has addressed the limitation(s) identified, and has improved on the state of the art.
4. To provide recommendations based on the findings of this research project.

The first part of this research project was to gain an understanding of the current state of the art in incident detection. Firstly, a review of previously presented IDAs was undertaken. This

review covered many different types of IDA, including various data sources, approaches, and types of network. Next, a review of surveys of TMC operators was undertaken in order to understand how these IDAs were being used in practice. Because the surveys reviewed were undertaken 10 years ago, and in different countries, interviews with TMC operators were also undertaken to provide a more relevant perspective. Together, these reviews and surveys provided an understanding of the current state of incident detection, limitations that remained, and where opportunities for improvement could be made. As such, this chapter addressed objective 1 of this research project. It was concluded that the IDAs implemented in practice had been unsuitable for the needs of TMCs, and that the primary cause of this was a high false alert rate. In the literature, the commonly cited reason for IDAs' high false alert rates was from failing to differentiate disruption from incidents and contexts. As such, this limitation was chosen as the focus of this research project.

The novel approach taken in this research project was to gain a better understanding of traffic conditions that could be 'expected' to occur by incorporating contexts within a traffic forecasting algorithm. With this forecast, the distinction between incident and non-incident conditions (including disruption from contexts) was to be made by identifying a sufficient difference between the forecasts and real-time traffic conditions.

The first part of this approach was to develop the traffic forecasting algorithm, named RoadCast. Chapter 5 developed this algorithm, and chapter 6 evaluated it. RoadCast was found to be 4.4% more accurate than a benchmark historical average predictor in terms of mean squared error when forecasting flow, and 4.0% when forecasting average speed. Much of this improvement came from RoadCast's ability to more accurately forecast traffic conditions at times when contexts were causing disruption. It was found that with more training data, more contexts could be accounted for and hence more accurate forecasts could be achieved. It was also found that the forecast accuracy decreased as the forecast horizon increased. However, with both variation in the amount of training data used and the forecast horizon, RoadCast was consistently more accurate than the historical average. Finally, methods to interpret RoadCast's decision making process were implemented, and revealed some insight into how contexts were being used by RoadCast. The contribution method also showed promise as a potential tool which could be used to inform an operator about the reasons behind certain traffic conditions being forecasted.

Next, the traffic forecasting algorithm, RoadCast, was developed into the incident detection algorithm, RCID. Chapter 7 describes the development and evaluation of this algorithm. In offline tests, RCID performed better when using contextual data as input, both in terms of false alert rate and detection rate. It also outperformed existing IDAs RAID and McMaster, and achieved performance that was within the 'acceptable bounds' stated by in a TMC operator survey (Ritchie and Abdulhai, 1997). Analysis revealed how RCID was able to use contextual data inputs to better differentiate disruption from incidents and contexts.

In chapter 8, an online test of RCID in Brisol's TMC showed that the IDA was also usable in practice, and improved on the TMC's existing methods. The tests also showed that the largest

limitations of RCID was still the false alert rate and usability, but a number of ways in which these could be improved were stated.

The findings of this research demonstrate that disruption from contexts and incidents can be more readily differentiated by incorporating contextual data within incident detection algorithms. It was found that this can also be achieved in a way that improves on the state of the art, and that is suitable for use in TMCs. The ultimate implication of this finding is that incidents can be detected more quickly and reliably, and hence operators can respond in a way that reduces their disruption. Throughout this research project, recommendations were made as to how further progress in this area could be made in the future.

9.2 Contribution

In section 1.9, a number of ways in which this research project was expected to contribute were outlined. These contributions were split into contributions to academia and contributions to industry. The following sections describe whether this research has achieved the expected contributions outlined in this section.

The main expected contribution to academia was to allow for future research in developing state of the art IDAs to build on this research project's findings, specifically by addressing limitations with current state of the art IDAs. By demonstrating how and why IDAs can improve their performance by incorporating contextual data, this research project has contributed in this way. This has been demonstrated via publication in major conferences and journals (details of which can be found in appendix E), as well as via communication and collaboration with many academics. By describing the key findings and limitations of the proposed method, researchers in the future could build on this research project to design new IDA methodologies that further improve on the performance of the current state of the art.

The other main expected contribution of this research project was to industry. TMCs would be able to benefit from the research directly if the created IDA performed well in practice (i.e. in live applications and on field data), and would be easy to implement, maintain and transfer to other locations. When implemented, the IDA would aid TMCs to respond more quickly and hence effective to incidents, and hence reduce the impact of the negative consequences associated with them.

The offline tests of RCID in chapter 7 showed that RCID improves on the performance of state of the art IDAs, and successfully addresses the issue of differentiating incidents from contexts. Chapter 8 showed that operators in Bristol's TMC found the IDA to be a useful addition to their existing incident detection methods. Siemens Mobility Limited have recognised the potential of RoadCast and RCID, and have plans to implement both within their software products. Towards this goal, the algorithms have been included in a number of successful bids for work, and a production-ready version of RoadCast's code has been developed. Also, an externally funded

research project to reduces the amount of manual effort required to collect contextual data for RoadCast is currently ongoing. These real-world applications show the direct contribution of this research project to industry. In future, it is hoped that RCID can aid operators in many TMCs, and RoadCast can become a key part of many ITS applications that require an accurate traffic forecast at a horizon of multiple days.

9.3 Implementation considerations

9.3.1 Contextual data collection and processing

The incorporation of contextual data within RCID was seen as a worthwhile trade-off between the increase in performance made possible, and the challenges involved in the collection and pre-processing of the data. However, the additional requirements of incorporating contextual data should be considered when implementing RCID in a TMC.

For this research project, sources of contextual variation were identified by searching the internet for contexts which intuitively may have been causing variation. By comparing the contextual data with traffic data, suitable contexts, and their websites, were identified. The contextual data was collected using web scrapers (an automated process of data retrieval) for some websites, and manually for others. A simple algorithm was then run to convert this data into the form described in table 5.2.

Each of the tasks described above added to the complexity of implementing RCID in TMCs. Firstly, it is hoped that the TMC operators could identify which contexts would cause variation in traffic conditions in the network covered by their TMC. But once identified, there exists non-trivial task to automate the process of identifying relevant websites, and scraping and pre-processing the contextual data. Without automating this process, the user would be required to complete this process every time RoadCast is re-trained (and/or forecasts a different time period). These tasks could be somewhat generalised for some contexts, e.g. public holidays could be generalised across England. But for other contexts, such as street parades, the task is more difficult because the website holding the data is likely different for different TMCs.

9.3.2 Re-training frequency

Another consideration when implementing RCID is the frequency at which re-training occurs. Section 6.7 showed that RoadCast's forecast accuracy decreased as its horizon increased, but it remained more accurate than the historical average, and could still be seen to forecast contexts more accurately. Retraining should occur as frequently as is feasible, such that the performance of RCID remains as high as possible. However, as RoadCast's accuracy was found to deteriorate slowly as the horizon increases, retraining could occur as infrequently as yearly if needed.

9.3.3 Incident detection sensitivity

As was noted in section 3.2.4, the IDA performance measures of detection rate, false alert rate and average time to detect can be seen as a trade-off. If an IDA is tuned to become more sensitive in raising alerts, it will be prone to having a higher detection, but also higher false alert rate.

The sensitivity of RCID can be altered depending on the requirements of the RCID. Section 3.3 suggested that two comparatively large TMCs, Bristol and Cardiff, would prefer an IDA to be the first indicator of an incident, and operators could manually check each alert via CCTV. This role of an IDA would require a comparatively low average time to detect, at the expense of a high false alert rate and low detection rate. The Head of Consultancy Services at Siemens Mobility Limited suggested that smaller (e.g. unmanned) TMCs would prefer an IDA that only raised alerts when it was "99% sure" that an incident had occurred, i.e. a high detection rate and low false alert rate, at the expense of a longer average time to detect.

When implementing RCID in a TMC, consideration should be given to the sensitivity of the algorithm, i.e. the percentile value at which the prediction intervals are produced. This may need to be an iterative process, where operators give feedback until the desired sensitivity is achieved.

9.4 Future work

This research project has answered the objectives set out in section 1.8, but further areas of research have been identified during the project. This section describes areas for which further research would build upon and improve the methods presented in this thesis, but were seen as out of scope of this research project.

9.4.1 Contextual data collection

In section 6.10, RoadCast was described to be limited in that before implementation in a new network: local knowledge of disruptive contexts, identification of data sources of such contexts, and methods to collect such data would be required. With future research however, it is thought that these tasks could be automated to some extent.

Perhaps the most difficult task to automate would be to automatically identify disruptive contexts, and their data sources. This would likely require an agent to search for relevant data sources on the internet, and identify their relevance by making comparisons to a traffic dataset. To automate the process of collecting data of disruptive contexts, a web scraper would likely be required. However, a challenge can be seen in forming a web scraper that would be transferable to any form and location of data that could be found on the internet.

9.4.2 Algorithm performance

Despite RCID being shown to improve on existing IDAs McMaster and RAID in terms of false alert rate and detection rate, and showing improvement in false alert rate by incorporating contextual data, operators still found that RCID's biggest limitation was its false alert rate.

A number of reasons were suggested for the outstanding false alert rate. Likely the largest contributor comes from the noise in the data leading to innaccurate predictions. This noise could come from unaccounted for contexts, innaccurate data collection from loop detectors, random noise in the variation of individual vehicle's behaviour etc. Some practical methods in which this noise could be accounted were suggested, including incorporating roadwork and traffic signal timing data and using alternative data sources such as flow counts from CCTV cameras. Each of these areas have been studied in the literature, and are tractable ways in which IDA performance can continue to improve in the future.

The traffic forecasts achieved in this research project could also be improved if unrepresentative training data were removed in a more effective manner. Firstly, for periods where detectors were faulty and produced a value of zero in the dataset. And secondly, for training data that had been disrupted by incidents. In this study, this data simply remained included based on the premise that incidents were rare. But if this data could be identified, for example by an offline IDA, or from TMC logs, it could be removed from the training dataset.

The traffic forecasts could also have been improved by investigating more ways to encode features used in the random forest model, in particular the contextual features. For example, alternatives methods of encoding features in a categorical manner could be investigated.

9.4.3 Operator user experience

As well as further areas of research, the online test of RCID demonstrated the importance of the user experience of incident detection algorithm systems in the real-world. This includes the systems in place to display incident alerts, the user interface, and the additional features available to the operator. Together, these strongly influence the usability of IDAs, and hence affect their usability in the real-world. The online test resulted in a number of suggestions by operators to improve on the test's user experience, including grouping individual detectors' alerts for each incident in order to reduces the amount of excess information displayed to operators.

9.4.4 Further evaluation

Further insight into the effectiveness of the IDAs compared in this research project could have been gained with the use of more evaluation metrics. For example, in this project, the mean time to detect could not be evaluated because the ground truth data used in both tests did not provide sufficient information to do so. With the collection of appropriate data, a comparison of

RCID to the state of the art could be made.

RoadCast plays a key part in determining the performance of RCID, and so a more thorough evaluation of RoadCast would also be beneficial. For example, the other algorithms that were compared in section 5.4. could be developed further for this problem, by improving feature encoding, hyper-parameter tuning etc. If this was the case, it may be that a different algorithm design is better suited to the problem.

9.5 Conclusions

In summary, this research project has provided novel methods for predicting road traffic conditions, and detecting incidents. These methods have been found to outperform the state of the art, and have found to be usable and beneficial in a real-world test. These methods have contributed to the academic research via publication and communication, and have created direct contributions to industry through this project's collaboration with Siemens Mobility Limited. Ultimately, this research has helped operators detect incidents more quickly and reliably, allowing their consequences to be managed more effectively, saving time and money for both TMCs and the travelling public.

References

Abdulhai, B. and 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, M. M., Abdel-Aty, M. and Yu, R. (2012), 'Assessment of interaction of crash occurrence, mountainous freeway geometry, real-time weather, and traffic data', *Transportation Research Record* **2280**.

Ahmed, S. and Cook, A. R. (1982), 'Application of time-series analysis techniques to freeway incident detection', *Transportation Research Record* **841**, 19–21.

Al Hassan, Y. and Barker, D. J. (1999), 'The impact of unseasonable or extreme weather on traffic activity within lothian region, scotland', *Journal of Transport Geography* **7**(3), 209–213.

Amit, Y. and Geman, D. (1997), 'Shape quantization and recognition with randomized trees', *Neural computation* **9**(7), 1545–1588.

Anbaroglu, B., Heydecker, B. and Cheng, T. (2014), 'Spatio-temporal clustering for non-recurrent traffic congestion detection on urban road networks', *Transportation Research Part C: Emerging Technologies* **48**, 47–65.

Andersen, O. and Torp, K. (2016), A data model for determining weathers impact on travel time, *in* 'proceedings of the International Conference on Database and Expert Systems Applications', pp. 437–444.

Bajwa, U. and Kuwahara, M. (2003), 'A travel time prediction method based on pattern matching technique', *Publication of: ARRB Transport Research, Limited* **21**, 997–1010.

Balke, K. N. (1993), 'An evaluation of existing incident detection algorithms', *Interim Report No. FHWA-RD-75-39 for the Texas Department of Transportation* .

Bojarski, M., Del Testa, D., Dworakowski, D., Firner, B., Flepp, B., Goyal, P., Jackel, L. D., Monfort, M., Muller, U. and Zhang, J. (2016), 'End to end learning for self-driving cars', *arXiv preprint arXiv:1604.07316* .

Bowers, D., Bretherton, R., Bowen, G. and Wall, G. (1996), 'Traffic congestion incident detection', *TRL Report 217* .

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

Brown, R. G. (2004), *Smoothing, forecasting and prediction of discrete time series*, Courier Corporation.

Bruce, L., Balraj, N., Zhang, Y. and Yu, Q. (2003), 'Automated accident detection in intersections via digital audio signal processing', *Transportation Research Record* **1840**(1), 186–192.

Buylaert, W. (1999), 'Reducing the severity of road injuries through post impact care', *European Journal of Emergency Medicine* **6**, 271–274.

Cambridge Systematics Inc. (2001), 'Reliability: Providing a highway system with reliable travel times', *Special Report - National Research Council, Transportation Research Board* **260**, 113–116.

Changnon, S. A. (1996), 'Effects of summer precipitation on urban transportation', *Climatic Change* **32**(4), 481–494.

Cherrett, T., Bell, H. and McDonald, M. (2001), Estimating vehicle speed using single inductive loop detectors, *in* 'proceedings of Institution of Civil Engineers: Transport', Vol. 147, pp. 23–32.

Cherrett, T., Waterson, B., McDonald, M., Clarke, R., Bangert, A. and Morris, R. (2002), 'Improved network monitoring using UTC detector data 'RAID'', *Traffic Engineering and Control* **43**(4), 135–137.

Chin, S.-M., Franzese, O., Greene, D. L., Hwang, H.-L. and Gibson, R. (2004), *Temporary losses of highway capacity and impacts on performance: Phase 2*, United States. Department of Energy.

Chrobok, R., Kaumann, O., Wahle, J. and Schreckenberg, M. (2000), Three categories of traffic data: Historical, current, and predictive, *in* 'proceedings of the 9th IFAC Symposium Control in Transportation Systems', Vol. 33, pp. 221–226.

Chrobok, R., Kaumann, O., Wahle, J. and Schreckenberg, M. (2004), 'Different methods of traffic forecast based on real data', *European Journal of Operational Research* **155**(3), 558–568.

Chung, E. (2003), Classification of traffic pattern, *in* 'proceedings of the 11th World Congress on ITS', pp. 687–694.

Collins, J. F., Hopkins, C. M. and Martin, J. A. (1979*a*), 'Automatic incident detection - TRRL algorithms HIOCC and PATREG'. TRRL Supplementary Report 526.

Collins, J., Hopkins, C. and Martin, J. (1979*b*), Automatic incident detection-trrl algorithms hiocc and patreg, Technical report.

CompTIA (2016), 'Sizing Up the Internet of Things'. Research Brief. Accessed July 22, 2019. **URL:** *https://www.comptia.org/resources/sizing-up-the-internet-of-things*

Connelly, L. B. and Supangan, R. (2006), 'The economic costs of road traffic crashes: Australia, states and territories', *Accident Analysis and Prevention* **38**(6), 1087–1093.

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

Cookson, G. and Pishue, B. (2016), 'INRIX Global Traffic Scorecard'. Accessed July 22, 2019. **URL:** *http://inrix.com/resources/inrix-2016-traffic-scorecard-uk/*

Crawford, F., Watling, D. and Connors, R. (2017), 'A statistical method for estimating predictable differences between daily traffic flow profiles', *Transportation Research Part B: Methodological* **95**, 196–213.

Deniz, O. and Celikoglu, H. B. (2011), Overview to some existing incident detection algorithms: a comparative evaluation, *in* 'proceedings of the 15th Meeting of the Euro Working Group on Transportation.', pp. 1–13.

Department for Transport (2018), 'Accident and casualty costs'. Accessed November 22, 2019.
URL: *https://www.gov.uk/government/statistical-data-sets/ras60-average-value-of-preventing-road-accidents*

Dia, H. and 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.

Dudek, C. L., Messer, C. J. and Nuckles, N. B. (1974), 'Incident detection on urban freeways', *Transportation Research Record* **495**, 12–24.

Dudek, C. L., Weaver, G. D., Ritch, G. P. and Messer, C. J. (1975), 'Detecting freeway incidents under low-volume conditions', *Transportation Research Record* **533**, 34–47.

Eliasson, J., Hultkrantz, L., Nerhagen, L. and Rosqvist, L. (2009), 'The Stockholm congestion charging trial 2006: Overview of effects', *Transportation Research Part A: Policy and Practice* **43**(3), 240–250.

European Global Navigation Satellite System Agency (2018), 'eCall emergency alert system launched'. Accessed 5th November, 2018.
URL: *https://www.gsa.europa.eu/newsroom/news/ecall-emergency-alert-system-launched*

Evanco, W. M. (1996), *The impact of rapid incident detection on freeway accident fatalities*, United States. Joint Program Office for Intelligent Transportation Systems.

Federal Highway Administration (2006), 'Traffic Detector Handbook: Third Edition, Volume I, Chapter 2: Sensor Technology'. Accessed July 22, 2019.
URL: *https://www.fhwa.dot.gov/publications/research/operations/its/06108/02.cfm*

Gall, A. I. and Hall, F. L. (1989), 'Distinguishing between incident congestion and recurrent congestion: a proposed logic', *Transportation Research Record* (1232).

Geopy contributors (2017), 'geopy'. Accessed 3rd November, 2017.
URL: *https://pypi.python.org/pypi/geopy*

Ghosh, B. and Smith, D. P. (2014), 'Customization of automatic incident detection algorithms for signalized urban arterials', *Journal of Intelligent Transportation Systems* **18**(4), 426–441.

Goodwin, L. C. (2002), 'Weather impacts on arterial traffic flow', *Federal Highway Administration, Washington, prepared for Road Weather Management Program* .

Google Maps (2017), 'Distance Matrix API'. Accessed 3rd November, 2017.
URL: *https://developers.google.com/maps/documentation/distance-matrix/*

Goves, C., North, R., Johnston, R. and Fletcher, G. (2016), 'Short term traffic prediction on the uk motorway network using neural networks', *Transportation Research Procedia* **13**(1), 184.

Gu, Y., Qian, Z. and Chen, F. (2016), 'From Twitter to detector: Real-time traffic incident detection using social media data', *Transportation Research Part C: Emerging Technologies* **67**, 321–342.

Guin, A. (2004), An incident detection algorithm based on a discrete state propagation model of traffic flow, PhD thesis.

Gummer, J. (2016), UK climate action following the Paris Agreement, Technical report, U.K. Committee on Climate Change. Accessed July 22, 2019.
URL: *https://www.theccc.org.uk/wp-content/uploads/2016/10/UK-climate-action-following-the-Paris-Agreement-Committee-on-Climate-Change-October-2016.pdf*

Hagel, J. and Seely Brown, J. (2013), From exponential technologies to exponential innovation, Technical report, Deloitte. Accessed July 22, 2019.
URL: *https://www2.deloitte.com/insights/us/en/industry/technology/from-exponential-technologies-to-exponential-innovation.html*

Hampshire County Council (2018), 'ROMANSE Twitter account'. Accessed March 8, 2018.
URL: *https : //twitter.com/romanse*

Hawas, Y. E. (2007), 'A fuzzy-based system for incident detection in urban street networks', *Transportation Research Part C: Emerging Technologies* **15**(2), 69–95.

Hawas, Y. E. and Mohammad, M. S. (2015), A system for incident detection in urban traffic networks, *in* 'proceedings of the IEEE Conference on Intelligent Transportation Systems (ITSC)', pp. 2405–2411.

Horvitz, E. J., Apacible, J., Sarin, R. and Liao, L. (2012), 'Prediction, expectation, and surprise: Methods, designs, and study of a deployed traffic forecasting service', *arXiv preprint arXiv:1207.1352* .

Hou, Y., Edara, P. and Sun, C. (2015), 'Traffic flow forecasting for urban work zones', *IEEE Transactions on Intelligent Transportation Systems* **16**(4), 1761–1770.

Hu, J., Kaparias, I. and Bell, M. G. (2009), 'Spatial econometrics models for congestion prediction with in-vehicle route guidance', *IET Intelligent Transport Systems* **3**(2), 159–167.

ImageNet (2017), 'Large Scale Visual Recognition Challenge 2017 (ILSVRC2017)'. Accessed May 25, 2017.
URL: *image-net.org/challenges/LSVRC/2017/*

Ivan, J. N. (1993), Arterial street incident detection using multiple data sources: plans for advance, *in* 'proceedings of Pacific Rim TransTech Conference', Vol. 1, pp. 429–435.

Ivan, J. N., Schofer, J. L., Koppelman, F. S. and Massone, L. L. (1995), 'Real-time data fusion for arterial street incident detection using neural networks', *Transportation Research Record* (1497), 27–35.

Jia, Y., Wu, J. and Xu, M. (2017), 'Traffic flow prediction with rainfall impact using a deep learning method', *Journal of Advanced Transportation* **2017**(Article ID 6575947).

Kaggle (2017), 'Kaggle homepage'. Accessed May 25, 2017.
URL: *www.kaggle.com/*

Kamga, C. and Yazici, M. A. (2014), 'Temporal and weather related variation patterns of urban travel time: Considerations and caveats for value of travel time, value of variability, and mode choice studies', *Transportation Research Part C: Emerging Technologies* **45**, 4–16.

Kamijo, S., Matsushita, Y., Ikeuchi, K. and Sakauchi, M. (2000), 'Traffic monitoring and accident detection at intersections', *IEEE transactions on Intelligent transportation systems* **1**(2), 108–118.

Kastrinaki, V., Zervakis, M. and Kalaitzakis, K. (2003), 'A survey of video processing techniques for traffic applications', *Image and vision computing* **21**(4), 359–381.

Kerner, B. S. and Konhäuser, P. (1993), 'Cluster effect in initially homogeneous traffic flow', *Physical Review E* **48**, R2335–R2338.

Khan, S. I. (1997), 'Modular neural network architecture for detection of operational problems in urban arterials', *Transportation Research Part A* **1**(31), 64.

Khan, S. I. and Ritchie, S. G. (1998), 'Statistical and neural classifiers to detect traffic operational problems on urban arterials', *Transportation Research Part C: Emerging Technologies* **6**(5), 291–314.

Ki, Y.-K. (2007), 'Accident detection system using image processing and MDR', *International Journal of Computer Science and Network Security IJCSNS* **7**(3), 35–39.

Kim, C., Park, Y.-S. and Sang, S. (2008), Spatial and temporal analysis of urban traffic volume, *in* 'proceeedings of ESRI International User Conference', Vol. 10, p. 2013.

Kourou, K., Exarchos, T. P., Exarchos, K. P., Karamouzis, M. V. and Fotiadis, D. I. (2015), 'Machine learning applications in cancer prognosis and prediction', *Computational and Structural Biotechnology Journal* **13**, 8–17.

Lam, W. H., Tam, M. L., Cao, X. and Li, X. (2013), 'Modeling the effects of rainfall intensity on traffic speed, flow, and density relationships for urban roads', *Journal of Transportation Engineering* **139**(7), 758–770.

Lam, W. H., Tam, M. L. and Li, X. (2016), 'Automatic traffic incident detection algorithm for both rain and no-rain conditions', *Asian Transport Studies* **4**(2), 330–349.

Laña, I., Del Ser, J., Padró, A., Vélez, M. and Casanova-Mateo, C. (2016), 'The role of local urban traffic and meteorological conditions in air pollution: A data-based case study in Madrid, Spain', *Atmospheric Environment* **145**, 424–438.

Lee, J.-T. and Taylor, W. C. (1999), 'Application of a dynamic model for arterial street incident detection', *ITS Journal - Intelligent Transportation Systems Journal* **5**(1), 53–70.

Lee, S., Krammes, R. A. and Yen, J. (1998), 'Fuzzy-logic-based incident detection for signalized diamond interchanges', *Transportation Research Part C: Emerging Technologies* **6**(5), 359–377.

Leo Breiman (2017), 'Random forests'. Accessed 16th October, 2017.
URL: *https://www.stat.berkeley.edu/∼breiman/RandomForests/cc_home.htm*

Leshem, G. and Ritov, Y. (2007), Traffic flow prediction using adaboost algorithm with random forests as a weak learner, *in* 'proceedings of World Academy of Science, Engineering and Technology', Vol. 19, pp. 193–198.

Levin, M. and Krause, G. M. (1978), 'Incident detection: a bayesian approach', *Transportation Research Record* **682**, 52–58.

Li, J. (2017), 'Bagging and Random Forests'. Accessed July 22, 2019.
URL: *https://newonlinecourses.science.psu.edu/stat508/lesson/11/11.9*

Li, M.-T., Zhao, F. and Wu, Y. (2004), 'Application of regression analysis for identifying factors that affect seasonal traffic fluctuations in southeast Florida', *Transportation Research Record* (1870), 153–161.

Louppe, G. (2014), Understanding random forests: From theory to practice, PhD thesis.

Mahmassani, H. S., Haas, C., Zhou, S. and Peterman, J. (1999), 'Evaluation of incident detection methodologies', *Research Report (9/97 9/98) for the Texas Department of Transportation* p. 1.

Mai, E. and Hranac, R. (2013), Twitter interactions as a data source for transportation incidents, *in* 'proceedings of the Transportation Research Board 92nd Annual Meeting'.

Martin, P. T., Perrin, J. and Hansen, B. (2001), 'Incident detection algorithm evaluation', *Prepared for the Utah Department of Transportation* .

Meinshausen, N. (2006), 'Quantile regression forests', *Journal of Machine Learning Research* **7**(Jun), 983–999.

National Traffic Incident Management Coalition (2017), 'National Unified Goal (NUG)'. Accessed 16th October, 2017.
URL: *http://ntimc.transportation.org/Pages/NationalUnifiedGoal(NUG).aspx*

Nguyen, H., Liu, W., Rivera, P. and Chen, F. (2016), Trafficwatch: real-time traffic incident detection and monitoring using social media, *in* 'Pacific-asia conference on knowledge discovery and data mining', pp. 540–551.

NHS England (2017), 'NHS ambulance services'. Accessed February 25, 2019.
URL: *https://www.nao.org.uk/wp-content/uploads/2017/01/NHS-Ambulance-Services.pdf*

Nielsen, T. D. and Jensen, F. V. (2009), *Bayesian networks and decision graphs*, Springer Science and Business Media.

Nofal, F. H. and Saeed, A. A. W. (1997), 'Seasonal variation and weather effects on road traffic accidents in Riyadh City', *Public Health* **111**(1), 51–55.

Nookala, L. S. (2006), Weather impact on traffic conditions and travel time prediction, PhD thesis.

Oskarbski, J., Zawisza, M. and Zarski, K. (2016), 'Automatic incident detection at intersections with use of telematics', *Transportation Research Procedia* **14**, 3466–3475.

Ozbay, Kaan; Kachroo, P. (1999), 'Incident management in intelligent transport systems', pp. 1–248. Monograph. Accessed July 22, 2019.
URL: *https://digitalscholarship.unlv.edu/ece_fac_articles/103/*

Palczewska, A., Palczewski, J., Robinson, R. M. and Neagu, D. (2014), 'Interpreting random forest classification models using a feature contribution method', *Integration of Reusable Systems* pp. 193–218.

Pan, B., Demiryurek, U., Gupta, C. and Shahabi, C. (2015), 'Forecasting spatiotemporal impact of traffic incidents for next-generation navigation systems', *Knowledge and Information Systems* **45**(1), 75–104.

Pan, B., Demiryurek, U. and Shahabi, C. (2012), Utilizing real-world transportation data for accurate traffic prediction, *in* 'proceedings of the IEEE 12th International Conference on Data Mining', pp. 595–604.

Parkany, E. and Xie, C. (2005), 'A complete review of incident detection algorithms and their deployment: what works and what doesn't'. Report for the New England Transportation Consortium.

Parr, T., Turgutlu, K., Csiszar, C. and Howard, J. (2018), 'Beware default random forest importances'. Accessed October 31, 2018.
URL: *http://explained.ai/rf-importance/index.html#6.3*

Payne, H. J. and Tignor, S. C. (1978), 'Freeway incident-detection algorithms based on decision trees with states', *Transportation Research Record* (682), 30–37.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R. and Dubourg, V. (2011), 'Scikit-learn: Machine learning in Python', *Journal of Machine Learning Research* **12**(Oct), 2825–2830.

Persaud, B. N. and Hall, F. L. (1989), 'Catastrophe theory and patterns in 30-second freeway traffic data: Implications for incident detection', *Transportation Research Part A: General* **23**(2), 103–113.

Persaud, B. N., Hall, F. L. and Hall, L. M. (1990), 'Congestion identification aspects of the mcmaster incident detection algorithm', *Transportation Research Record* (1287).

Purple and Cisco (2017), 'Purple and Cisco to get Manchester moving thanks to a grant from InnovateUK'. Accessed 15th November, 2017.
URL: *https://purple.ai/purple-cisco-manchester-innovateuk/*

Ritchie, S. G. and Abdulhai, B. (1997), 'Development testing and evaluation of advanced techniques for freeway incident detection', *California Partners for Advanced Transit and Highways (PATH)* .

Ritchie, S. G. and Cheu, R. L. (1993), 'Neural network models for automated detection of nonrecurring congestion', *California Partners for Advanced Transit and Highways (PATH)* .

Saeedmanesh, M. and Geroliminis, N. (2017), 'Dynamic clustering and propagation of congestion in heterogeneously congested urban traffic networks', *Transportation research procedia* **23**, 962–979.

Schroff, F., Criminisi, A. and Zisserman, A. (2008), Object class segmentation using random forests., *in* 'proceedings of British Machine Vision Conference 2008', pp. 1–10.

scikit-learn (2017*a*), 'Random forest regressor'. Accessed 24th August, 2017.
URL: *http://scikit-learn.org/stable/modules/generated /sklearn.ensemble.RandomForestRegressor.html*

scikit-learn (2017*b*), 'Random forest regressor user guide'. Accessed 24th August, 2017.
URL: *http://scikit-learn.org/stable/modules/ensemble.html#forest*

scikit-learn (2017*c*), 'Recursive feature elimination user guide'. Accessed July 22, 2019.
URL: *http://scikit-learn.org/stable/modules/generated/sklearn.feature_selection.RFE.html*

SciPy (2018), 'Paired t-test'. Accessed May 29, 2018.
URL: *https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.ttest_rel.html*

Sheu, J.-B. and Ritchie, S. G. (1998), 'A new methodology for incident detection and characterization on surface streets', *Transportation Research Part C: Emerging Technologies* **6**(5), 315–335.

Singh, K. and Xie, M. (2008), 'Bootstrap: a statistical method', *Unpublished manuscript, Rutgers University, USA* . Accessed July 22, 2019.
URL: *http://www.stat.rutgers.edu/home/mxie/RCPapers/bootstrap.pdf*

Singliar, T. and Hauskrecht, M. (2010), 'Learning to detect incidents from noisily labeled data', *Machine Learning* **79**(3), 335–354.

Siripanpornchana, C., Panichpapiboon, S. and Chaovalit, P. (2015), Effective variables for urban traffic incident detection, *in* 'proceedings of the IEEE Vehicular Networking Conference (VNC)', pp. 190–195.

Somerset Intelligence (2017), 'Road casualties - rural and urban'. Accessed May 22, 2017.
URL: *www.somersetintelligence.org.uk/road-casualties-rural-and-urban/*

Song, Y. and Miller, H. J. (2012), 'Exploring traffic flow databases using space-time plots and data cubes', *Transportation* **39**(2), 215–234.

Southampton City Council (2018), 'SCC Highways' Twitter account'. Accessed March 8, 2018.
URL: *https : //twitter.com/scchighways*

Stathopoulos, A. and Karlaftis, M. (2001), 'Temporal and spatial variations of real-time traffic data in urban areas', *Transportation data and Information Technology* (1768), 135–140.

Statnikov, A., Wang, L. and Aliferis, C. F. (2008), 'A comprehensive comparison of random forests and support vector machines for microarray-based cancer classification', *BMC bioinformatics* **9**(1), 319.

Steinwart, I. and Christmann, A. (2008), *Support vector machines*, Springer Science & Business Media.

Stephanedes, Y. and Hourdakis, J. (1996), 'Transferability of freeway incident detection algorithms', *Transportation Research Record* (1554), 184–195.

Strobl, C., Boulesteix, A.-L., Zeileis, A. and Hothorn, T. (2007), 'Bias in random forest variable importance measures: Illustrations, sources and a solution', *BMC Bioinformatics* **8**(1), 25.

Syrjarinne, P. (2016), Urban Traffic Analysis with Bus Location Data, PhD thesis.

Tarko, A. and Rau, L.-K. (2002), Logic-based incident detection on signalized streets with heterogeneous data, *in* 'proceedings of the IEEE 5th International Conference on Intelligent Transportation Systems', pp. 708–713.

Thancanamootoo, S. and Bell, M. (1988), 'Automatic detection of traffic incidents on a signal-controlled road network', *Transport Operations Research Group report* **76**(76).

Thomas, T. and van Berkum, E. C. (2009), 'Detection of incidents and events in urban networks', *IET Intelligent Transport Systems* **3**(2), 198.

Thomas, T., Weijermars, W. and van Berkum, E. (2008), 'Variations in urban traffic volumes', *European Journal of Transport and Infrastructure Research (EJTIR)* **3**(8), 71–80.

TrafficVision (2017), 'TrafficVision Case Studies'. Accessed May 22, 2017.
URL: *www.trafficvision.com/case-studies/*

UK Department for Transport (2017), Provisional road traffic estimates, Great Britain: July 2016-June 2017, Technical report, UK Department for Transport.

UK Department for Transport (2019), Reported road casualties in great britain: quarterly provisional estimates year ending june 2019, Technical report, UK Department for Transport.

UK Government (2017), 'Table NTS0905: Car occupancy, England: since 2002'. Accessed 3rd November, 2017.
URL: *https://www.gov.uk/government/statistical-data-sets/nts09-vehicle-mileage-and-occupancy*

UK Government (2018), 'Road Safety Data'. Accessed March 8, 2018.
URL: $https://data.gov.uk/dataset/road-accidents-safety-data$

Van Der Voort, M., Dougherty, M. and Watson, S. (1996), 'Combining Kohonen maps with ARIMA time series models to forecast traffic flow', *Transportation Research Part C: Emerging Technologies* **4**(5), 307–318.

Vanajakshi, L. and Rilett, L. R. (2004), A comparison of the performance of artificial neural networks and support vector machines for the prediction of traffic speed, *in* 'proceedings of the IEEE Intelligent Vehicles Symposium, 2004', pp. 194–199.

Vlahogianni, E. I., Golias, J. C. and Karlaftis, M. G. (2004), 'Short-term traffic forecasting: Overview of objectives and methods', *Transport Reviews* **24**(5), 533–557.

Wang, K.-f., Jia, X. and Tang, S. (2005), A survey of vision-based automatic incident detection technology, *in* 'proceedings of IEEE International Conference on Vehicular Electronics and Safety, 2005.', pp. 290–295.

Wardrop, J. G. (1952*a*), 'Road paper. some theoretical aspects of road traffic research.', *Proceedings of the institution of civil engineers* **1**(3), 325–362.

Wardrop, J. G. (1952*b*), 'Road paper. some theoretical aspects of road traffic research.', *Proceedings of the institution of civil engineers* **1**(3), 325–362.

Weijermars, W. A. M. (2007), Analysis of urban traffic patterns using clustering, PhD thesis.

Weil, R., Wootton, J. and Garcia-Ortiz, A. (1998), 'Traffic incident detection: Sensors and algorithms', *Mathematical and Computer Modelling* **27**(9), 257–291.

Westerman, M., Litjens, R. and Linnartz, J.-P. (1996), 'Integration of probe vehicle and induction loop data: Estimation of travel times and automatic incident detection', *California Partners for Advanced Transit and Highways (PATH) Research Report* .

Williams, B. M. and Guin, A. (2007), 'Traffic Management Center Use of Incident Detection Algorithms: Findings of a Nationwide Survey', *IEEE Transactions on Intelligent Transportation Systems* **8**(2), 351–358.

Williams, B. M. and Hoel, L. A. (2003), 'Modeling and forecasting vehicular traffic flow as a seasonal ARIMA process: Theoretical basis and empirical results', *Journal of Transportation Engineering* **129**(6), 664–672.

World Health Organisation (2017), 'Road traffic deaths, data by country'. Accessed 16th October, 2017.
URL: *http://apps.who.int/gho/data/node.main.A997*

Yan, W. (2006), Application of random forest to aircraft engine fault diagnosis, *in* 'proceedings of the Multiconference on Computational Engineering in Systems Applications', Vol. 1, pp. 468–475.

Yang, Z., Lin, C. and Gong, B. (2009), Support vector machines for incident detection in urban signalized arterial street networks, *in* 'proceedings of the 2009 International Conference on Measuring Technology and Mechatronics Automation (ICMTMA)', Vol. 3, pp. 611–616.

Yasdi, R. (1999), 'Prediction of road traffic using a neural network approach', *Neural Computing and Applications* **8**(2), 135–142.

Yass, I. (2017), Delays due to serious road accidents, Technical report, Prepared for the RAC foundation. Accessed July 22, 2019.
URL: *https://www.racfoundation.org/wp-content/uploads/2017/11/road-accident-delays-yass-april-report.pdf*

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

Yuan, L. Y., Chen, B. Y. and Lam, W. H. (2014), Effects of rainfall intensity on traffic crashes in Hong Kong, *in* 'proceedings of the Institution of Civil Engineers-Transport', Vol. 167, pp. 343–350.

Zarei, N., Ghayour, M. A. and Hashemi, S. (2013), Road traffic prediction using context-aware random forest based on volatility nature of traffic flows, *in* 'proceedings of the Asian Conference on Intelligent Information and Database Systems', pp. 196–205.

Zhang, K. and Taylor, M. A. P. (2006), 'Effective arterial road incident detection: A Bayesian network based algorithm', *Transportation Research Part C: Emerging Technologies* **14**(6), 403–417.

Zhang, R., Shu, Y., Yang, Z., Cheng, P. and Chen, J. (2015), Hybrid traffic speed modeling and prediction using real-world data, *in* 'proceedings of the IEEE International Congress on Big Data', pp. 230–237.

Zhang, Y. and Bruce, L. M. (2004), Automated accident detection at intersections, Technical report. Prepared for the Federal Highway Administration and Mississippi Department of Transportation. Accessed July 22, 2019.
URL: *https://rosap.ntl.bts.gov/view/dot/24144*

Zhou, T., Gao, L. and Ni, D. (2014), Road traffic prediction by incorporating online information, *in* 'proceedings of the 23rd International Conference on World Wide Web', pp. 1235–1240.

Zou, Y., Shi, G., Shi, H. and Wang, Y. (2009), Image Sequences Based Traffic Incident Detection for Signaled Intersections Using HMM, *in* 'proceedings of the 9th International Conference on Hybrid Intelligent Systems', Vol. 1, pp. 257–261.

Appendix A

IDAs in practice TMC interview questions

This appendix states the questions which were used for the TMC interviews undertaken during the literature review for this research project.

1. What is your job's role?
2. What area of road network is your TMC responsible for?
3. What types of roadways are managed in this network?
4. What is the role of a TMC operator?
5. What methods are employed to detect incidents in your TMC?
6. How are incidents responded to?
7. Has an IDA been implemented in your TMC? If yes, the following apply:
   (a) Is the IDA being used currently?
   (b) How does the IDA detect incidents?
   (c) What type of detectors does the IDA use?
   (d) What traffic variables are used?
   (e) Are you satisfied with the IDA?
   (f) How does your IDA aid the TMC's task of incident detection?
   (g) What are the benefits and drawbacks of the IDA?
   (h) Can you compare the IDA to others implemented previously? (if yes, go to question (a))
8. If a new IDA were to be developed for use in your TMC, what would be the most important features for it to have?

9. What would be an acceptable level of detection rate, false alert rate, and average time to detect be? (these metrics were explained in section 2.3.3)

Questions 1-4 were introductory questions that set the context in which the operator and the TMC manage incidents. Although not directly answering the research questions that these interviews aimed to answer, they could provide insight into the reasoning behind the operator's answers to later questions, such as what features the operator values most in an IDA.

Questions five and six aimed to understand how incidents were detected and managed in TMCs. From background research before these interviews, it was clear that the value placed on detecting and responding to incidents varied between different TMCs. Hence, the answer to these questions could explain the differences to later questions, such as the acceptable level of performance.

Question seven asked of previous experience operators may have had with IDAs. If this experience existed, it would be a useful complement to the literature review of online tests of IDAs. These experiences could also be used to affirm or deny the findings of the reviewed surveys of TMC operators

Question 8-9 were asked to understand what performance and features would be required by the IDA to be developed in this research project. These questions would again affirm or deny the findings of the reviewed TMC surveys. Together, the findings would be used to shape the design of the proposed IDA of this research project.

Appendix B

Southampton detector selection method

This section describes the process in which detectors in the Southampton dataset were selected for use. The detector's data were used in chapter 5, 6 and 7. The data was provided by Southampton City Council, who's TMC covers the city centre and surrounding boroughs. The network consists of arterials (such as A roads) and urban streets and signalised junctions, but no motorways or dual carriageways are covered.

Firstly a region of Southampton was chosen which had a high density of detectors and covered many types of travel demand and road network topology. This region covered the city centre, Portswood, Swaythling, Northam and Bitterne. The loop diagrams of the selected areas were then studied, and detectors which were indicated to still be working were taken forward.

Data from each of the remaining detectors was studied manually, by observing the graphs of flow and average speed. Detectors were removed if they appeared to show erroneous data for at least one month of the study period continuously. Table B.1 describes which detectors were removed and the reasons why. The remaining detectors were used in the Southampton case study.

Detector	Reason for exclusion
N06221G	Only values of 0 flow in test set
N06222E	Only values of 0 flow for entire study period
N06251B	Flows of 300 throughout April and May 2016
N07131Y	In loop diagrams but no data available
N07381D	Only values of 0 flow from 20th March 2015 to 5th May 2015
N07421A	Flow values of around 300 for the entire study period
N07221B	Only values of 0 flow in test set

Table B.1: Reasons for excluding certain detectors in the chosen region of Southampton's network.

Appendix C

RCID methodology alterations

In the previous section, an initial incident detection methodology was developed and tested in order to prove this research project's hypothesis. Although this initial methodology proved the hypothesis to be true, it is clear that there is room for improvement in the performance of RCID. As such, this section describes a number of alterations to RCID's methodology, and the performance of these methodologies on the same Southampton dataset. The aim is to find an improved methodology of RCID.

The changes made to RCID can be divided into two categories; changes to RCID's incident detection methodology, and changes that aim to take advantage of the spatial relationships between different detector's traffic data.

C.1 Methodological changes

The first alterations to be developed and tested are related to the methodology of the incident detection part of RCID. The method developed in section 7.2 was simplistic in order to focus on whether the general approach of RCID was able to demonstrate the benefit of incorporating contextual data within an IDA, and create an improvement on the state of the art. The methodological changes that will be presented and tested in this section aim to push the boundaries of the performance that can be achieved by RCID.

C.1.1 Changes made

The first change made was to alter the methodology of the persistence test. The goal of the persistence test was to ensure that alerts were being raised based on the trend of real-time traffic conditions, rather than raising alerts based on noise. The simple persistence test used in section 7.3 raised an alert if each of the latest three messages fell outside of RCID's prediction intervals. However, this approach relied on every one of the three messages not being affected by noise by a large amount. If during an incident one of the three messages was affected by noise such

that it fell inside of the prediction interval (unrepresentatively), the incident could go undetected.

It was thought that a persistence test would better account for the trend in traffic conditions if not every message in the persistence test was relied upon to be wholly representative. As such, the persistence test was replaced with a comparison of the average of the three most recent messages was created. This method simply takes the average of the three most recent values of a traffic variable, and compares this with the average of the average of the three corresponding values of the prediction interval's upper and lower bounds. If at any point the average of the three actual values are outside the average of the three upper or lower bound values, an alert is raised. This method is referred to as 'RCID-average'.

The next change was to experiment with including more traffic variables within RCID. The idea here was to more easily detect incidents that affected other traffic variables more than flow. For example, emergency roadworks that result in temporary speed limits may be detected more easily by using the average speed variable. It was thought that this change may also reduce RCID's false alert rate by allowing RCID to be less sensitive to raising alerts for flow due to the added insight gained from the other traffic variables.

The method of RCID with multiple traffic variables is to independently implement RCID on each traffic variable, run the IDA on each variable simultaneously, and then raise an alert when any traffic variable indicated an alert. This approach is seen as necessary in order to maintain the ability to detect incidents that only affected one traffic variable. Clearly, this method could be expected to produce more alerts than RCID's original methodology, so it is expected that the prediction interval value need be increased in order to reduce RCID's sensitivity accordingly. In this study, this methodology combines flow and average speed variables. This method is referred to as 'RCID-multi'.

C.1.2 Results

Figure C.1 shows the performances of the methodology alterations, along with the original version of RCID described in section 6.5.1. It should be noted that only the versions of RCID with contextual data were included in the results for clarity.

Every alteration resulted in a lower performance than the original version of RCID. Both alterations had similar performance levels at higher false alert rates, but a worse detection rate at lower false alert rates.

Comparing the FAR and DR at each prediction interval setting, RCID-multi had a higher DR, but also FAR, throughout. Taking the 90% prediction interval as an example, RCID-multi had a 1% higher DR (98%) than the original version of RCID, but a 3.87% higher FAR (5.35%). This effect was expected for the reason of the additional traffic variable resulting in more alerts being produced. It even resulted in RCID-multi having a 100% detection rate when using a prediction interval of 89% and below. However, it can be inferred from the results that the trade-off made

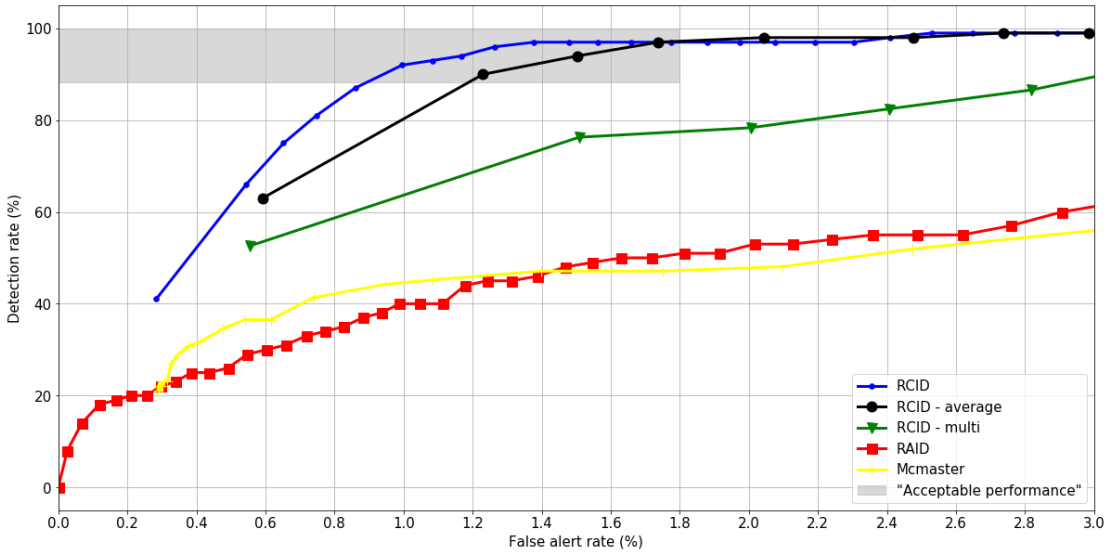


FIGURE C.1: RCID with various methodological alterations, false alert rates and detection rates. The grey area represents bounds for which Ritchie and Abdulhai (1997)'s survey of TMC operators deemed 'acceptable performance' (Ritchie and Abdulhai, 1997).

by incorporating the average speed variable was not as effective as simple altering the sensitivity by changing the prediction interval.

It should be noted that a downside of incorporating additional traffic variables is the manual and computational effort and time required. Adding one extra traffic variable doubles the time required to train RoadCast and increases the storage space required for the prediction intervals.

RCID-average also resulted in a higher DR and FAR throughout, but again insufficiently to result in an improvement in performance overall. At a 90% prediction interval, RCID-average had a 2% higher DR (99%) than the original version of RCID, but also a 2.56% higher FAR (4.04%). It was hoped that by using the average of the three messages, the effect of noise in real-time messages would be reducesd. However, because RCID required three consecutive real-time messages to be outside of the prediction interval, whereas RCID-average only required the average to be outside, RCID-average was more sensitive to raising alerts. Again, as can be seen in curve of performance of the two IDAs, this trade-off was unable to result in an improvement.

It is noted that the alterations made may have caused changes in other aspects of performance, such as the average time to detect or usability. Such changes may have revealed that there would be a benefit to one or more of the alterations. For example, RCID-multi may have had a lower average time to detect due to some incidents being detected faster using the average speed variable. Unfortunately, such aspects of performance could not be evaluated in this test. However, it is expected that both of the changes would not greatly impact RCID's average time to detect due to the requirement of three messages for the persistence test.

C.2 Spatial changes

To this point, the proposed incident detection algorithm has operated on each detector independently. This decision simplified the algorithm, and allowed the algorithm to be implemented with less manual calibration effort. It also meant that the hypothesis of this research project could be focused on more easily, as comparisons of the proposed algorithm to other state of the art algorithms could be made more readily.

However, it has often been found in the literature that the performance of incident detection algorithms can improve if a strategy is used to combine data from multiple detectors that are in some way spatially related. As such, investigation into whether RCID could improve its performance by implementing a spatial strategy was undertaken. Firstly, a literature review of previous spatial strategies within IDAs is undertaken, then an analysis of the behaviour of traffic during incidents is undertaken, then a spatial strategy is developed within RCID, and evaluated on real-world data.

C.2.1 Literature on spatial strategies

For IDAs designed for use on motorways, the most common spatial strategy is to detect incidents in-between a detector upstream and detector downstream of the incident. The logic here is that the incident causes a drop in capacity, which can create queuing upstream, causing the upstream detector to experience a drop in average speed and/or flow. Meanwhile, the downstream detector's flow can be reduced because the drop in capacity can reduce the flow of vehicles past the incident, but the average speed should remain the same. Hard-coded logic based on these assumptions (or similar) have been implemented within many motorway IDAs, and have often been found to improve performance (Payne and Tignor, 1978, Collins et al., 1979*a*, Gall and Hall, 1989, Ritchie and Cheu, 1993).

In an urban setting, the upstream/downstream strategy is not used as often (possible reasons for this are explained later). Instead, urban IDAs typically use spatial strategies that aim to make detectors more likely to raise alerts when nearby detectors indicate the presence of an incident. Lee et al. (1998) created hard-coded rules for detecting incidents on signalised diamond interchanges. its approach was to detect abrupt changes in traffic variable values over time. It used a rules based procedure based on traffic theory to estimate the location of incidents based on these abrupt changes. Hawas (2007) aimed to detect incident in-between nearby detectors by raising alerts when there were sufficient inconsistencies between same lane and adjacent lane detectors. Tarko and Rau (2002) used traffic theory based hard-coded rules in a similar spatial strategy, but did so in a way that was aimed to be suitable across all urban networks (including all types of signalised junctions). Anbaroglu et al. (2014) determined high link journey times in an urban network by comparing real-time data to historical averages. They then formed clusters by joining periods of high journey times if they were on adjacent links and overlapped in time. Such clusters are indicated as 'non-recurrent congestion', which was defined as an unexpected event that does not display a daily pattern (the definition is similar to that of an incident).

From the review undertaken, it is clear that many different types of spatial strategies can be employed to improve incident detection algorithms, but it is not clear which approach is most effective. However, it appears that different strategies are suitable for different types of network. In the literature, upstream/downstream strategies have appeared to be most suitable on motorways, but strategies based on detector proximity appear most suitable in urban areas.

The only suitable location to develop and evaluate RCID in this research project was found to be in the urban road network of Southampton. Given that the suitability of spatial strategies appears to depend on the type of network, RCID's spatial strategy would be developed to be as effective as possible on urban road networks only, even if this may introduce assumptions that make it unsuitable for other types of network e.g. motorways. This decision is inline with the scope of this research project, described in section 1.7.

C.2.2 Incident analysis

From the literature review undertaken in the previous section, it is apparent that many contrasting spatial strategies have been developed with the aim to improve IDA performance. To understand which would be most suitable for the road network of Southampton, an analysis of the incidents recorded in the city was undertaken. The aim was to further the understanding of the spatial behaviour of traffic conditions at times when incidents cause disruption. This analysis would then be used to help develop the spatial strategy of RCID, in order to maximise its performance.

The analysis was undertaken between the 15$^{\text{th}}$ December 2015 and 15$^{\text{th}}$ March 2016 (the last three months of the training data). Disruptive incidents were identified in the same way as described in section 7.3.1.3, i.e. by comparing tweets from Southampton's TMC with data from nearby detectors. All incidents that affected multiple detectors were included in this analysis, which in this study was 5 of 37.

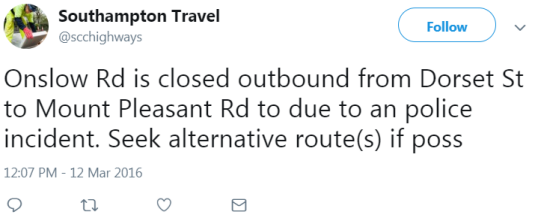
Case one

Date: 12th March 2016.

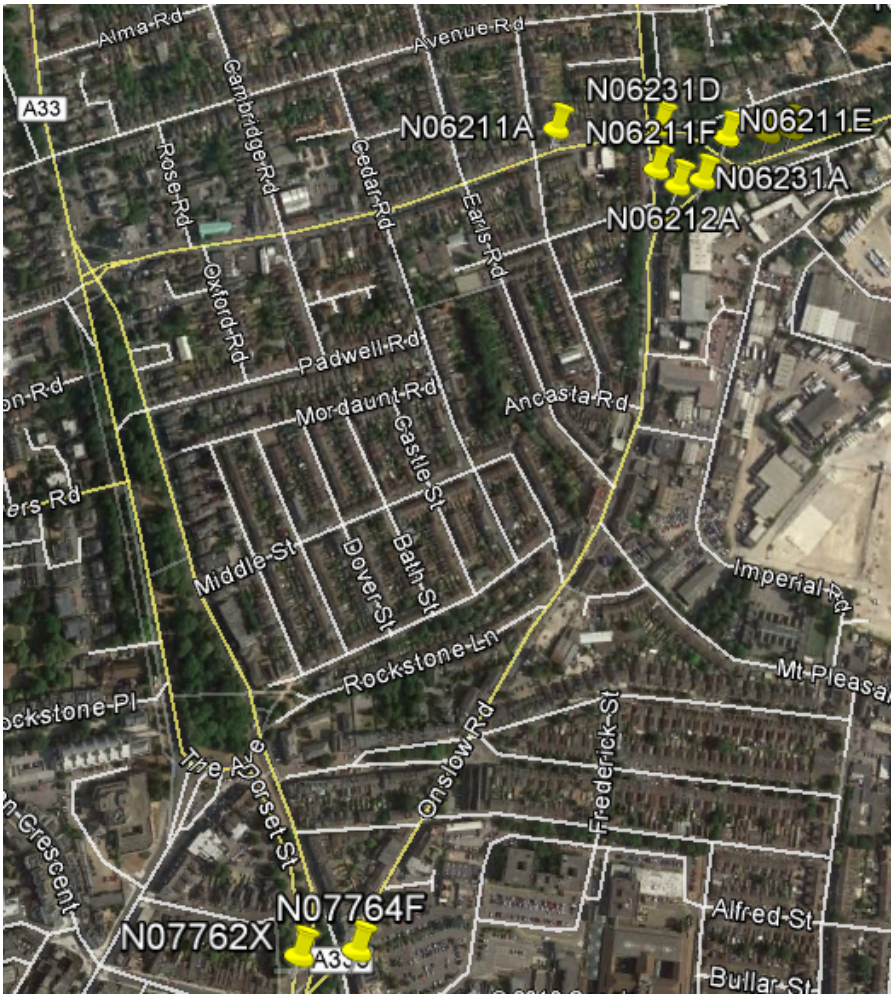
Location: Onslow road between Dorset St and mount Pleasant Rd.

Incident: Road closed due to a police incident.

Detector disruption: Increase in flow and decrease in average speed at N07762X, which is on the diversion route outbound (northbound) on Dorset St. Decrease in flow at N06212A, which is further along Onslow road, just after the road closure. It appeared that in this case, most travellers took a diversion, resulting in congestion on the road adjacent to the incident.



(A) Tweet.



(B) Map of affected detectors. Image created using Google Earth. ©2018 Google.

FIGURE C.2: Case one tweet and map of affected detectors.

Case two

Date: 10th March 2016.

Location: Stoneham way inbound (southbound) at the junction with Wide lane.

Incident: Road blocked due to road traffic incident.

Detector disruption: Decrease in flow on Stoneham way at N05161B due to travellers avoiding the area. N05161A is on the right hand turn lane from Stoneham way to Basset Green road, which experienced increased flow due to travellers taking a diversion from Stoneham way to Bassett Green road. N059111J is westbound on Bassett Green road, which had an increase in flow due to the diversion, but did not have a drop in average speed because the road was able to handle the increased demand at this point. N05151K also had a decrease in flow due to the diversion being taken. There was also a drop in average speed on Stoneham way at N05161A and N05161B, due to the queuing of vehicles trying to take the diversion route.

(A) Tweet.

(B) Map of affected detectors. Image created using Google Earth. ©2018 Google.

FIGURE C.3: Case two tweet and map of affected detectors.

Case three

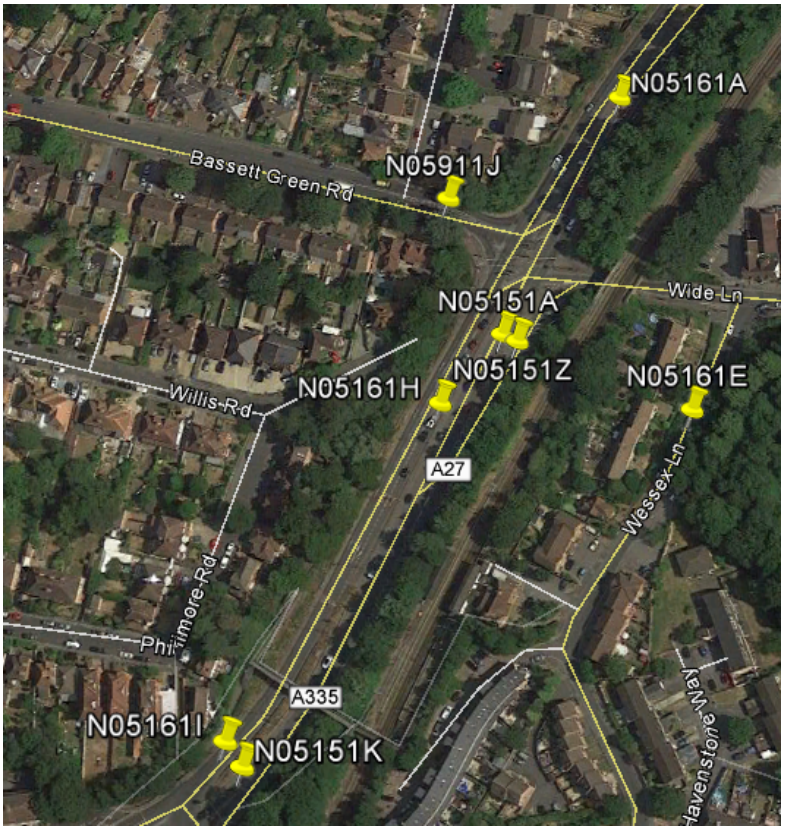
Date: 8th March 2016.

Location: Stoneham way outbound (northbound) after the junction with Basset Green road.

Incident: Road closed after junction with Basset Green road. Unknown cause.

Detector disruption: Increase in flow at N05161H, which is on the right hand turn lane from Stoneham way to Wide Lane, due to travellers taking a diversion on to Wide lane. Increase in flow at N05151K which is on Stoneham way southbound, due to travellers turning around and travelling the opposite direction. There was also a decrease in flow at N05161I on Stoneham way due to travellers avoiding the area. Each of the three detectors also experienced a decrease in average speed, most likely because of the incident causing increased congestion at the signalised junction (the junction between Stoneham way, Bassett Green road and Wide lane).

(A) Tweet.



(B) Map of affected detectors. Image created using Google Earth. ©2018 Google.

FIGURE C.4: Case three tweet and map of affected detectors.

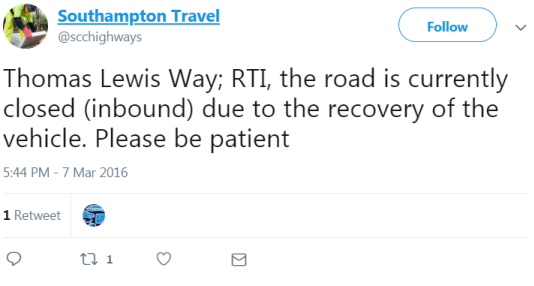
Case four

Date: 7th March 2016.

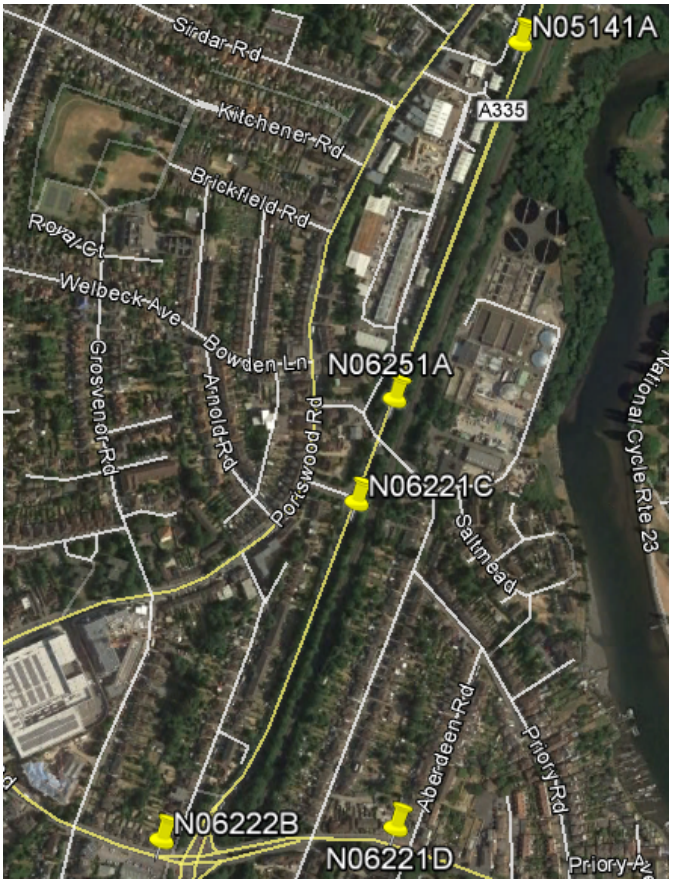
Location: Thomas Lewis Way inbound (southbound).

Incident: Road closed due to road traffic incident.

Detector disruption: Decrease in flow at N06251A and N06221C, which are both on Thomas Lewis Way inbound. The only disruption found was on the road in which the incident took place. This is likely because the incident only partially blocked the road, and it was on the main route into Southampton city centre, and so travellers simply took a slight delay on the road, rather than finding a diversion.



(A) Tweet.



(B) Map of affected detectors. Image created using Google Earth. ©2018 Google.

FIGURE C.5: Case four tweet and map of affected detectors.

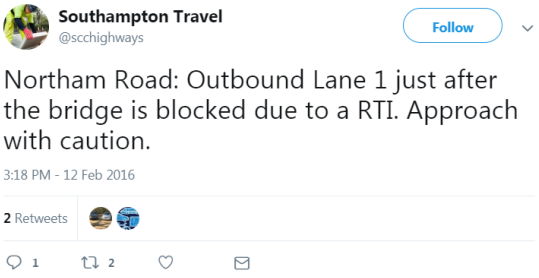
Case five

Date: 12th February 2016.

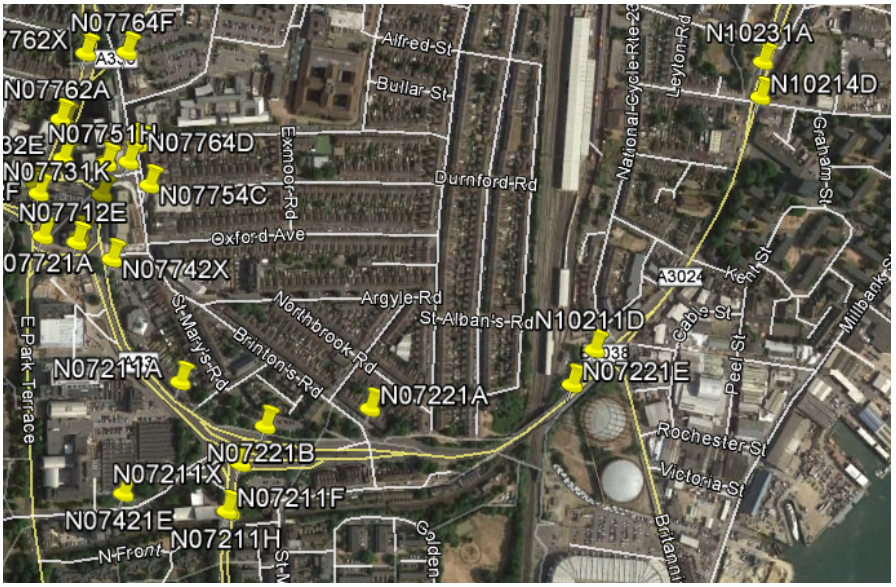
Location: Northam road outbound.

Incident: Lane 1 closed after the bridge due to road traffic incident.

Detector disruption: Decrease in flow and average speed on Northam road eastbound at N10231A, N10211D and N07211A, and a decrease in flow westbound at N07221E. It is likely that the average speed of N07221E was unaffected because it is at a stopline of a signalised junction, and so had typical average speeds of between 0 and 4mph, and so incident conditions could not be clearly differentiated. No diversion route could be identified. This could either be due to the lack of detector coverage on the surrounding roads, or because vehicles simply took the minor delay.



(A) Tweet.



(B) Map of affected detectors. Image created using Google Earth. ©2018 Google.

FIGURE C.6: Case five tweet and map of affected detectors.

It can be seen that the spatial patterns of incidents' disruption varied from case to case. In each case, incidents caused blockages in the road which reduced the road's capacity. However, the disruption at detectors surrounding the area varied from case to case. In cases one, two and three, disruption on both the road in which the incident took place and surrounding roads was observed. In each case travellers could be seen to take a diversion route. However, cases four and five only saw disruption on the road in which the incident took place, although in case five

disruption could be seen in the opposite direction (likely due to a change in signal timings).

The disruption of traffic conditions caused by incidents analysed in this study differ from the perceived knowledge of the disruption of incidents on motorways. Rather than simply causing queuing upstream and drop in flow downstream, adjacent roads and roads in the opposite direction are also often affected. It appears that determining how traffic conditions surrounding an incident will be affected is more complex in urban environments than on motorways. On motorways, typically only the road (and direction) of the incident are affected, congestion can occur upstream, and a drop in flow can occur downstream. In urban environments, it is difficult to predict whether vehicles will take a diversion route (and what that route will be), will turn around in the road, or simply take the delay. These responses may depend on the presence of possible diversion routes, the knowledge that drivers have of the incident and the local area, and their willingness to risk taking a diversion. As such, it is more complex to define a spatial strategy which will account for the traffic conditions in an urban network. However, a common thread that can be seen in this analysis is that disruption is often found at detectors in close proximity, in terms of distance to drive, to each incident.

This analysis could also explain why upstream/downstream spatial strategies are rarely used in urban networks, namely that the propagation of traffic during incidents can be more complex. Many urban networks have detectors placed at the entries and exits of signalised junctions, and are often less densely covered than on motorways. This can mean that upstream/downstream pairs often have a junction in-between, which could be used as a diversion. Another difference between urban and motorway incident disruption is that vehicles can cross on to the other side of the road to pass during lane blocking incidents, disrupting traffic conditions in the other direction. Another factor is that when incidents occur, traffic signal strategies are often changed to reduces disruption from the incident. Each of these factors mean that traffic behaviour is less predictable in urban networks compared to motorways, and so can make the task of IDAs more difficult.

C.2.3 Spatial methodology alterations

Based on the literature review and incident analysis undertaken above, a number of spatial based methodologies were developed for RCID.

The first was to alter the sensitivity of RCID to raising alerts when spatially nearby detectors were already raising alerts. That is, during a period in which a detector is raising an alert, increase the sensitivity of all nearby detectors by a specified amount. It could be seen from the incident analysis that incident disruption propagates along the driving route toward and away from the incident location. As such, the distance between detectors was measured based on the driving distance. This distance was calculated using the Google Maps Distance Matrix API (Google Maps, 2017), in the same way as was described in section 7.3.1.3. After initial testing on the training data, a suitable threshold for a detector being nearby was deemed to be 500 metres. The most suitable difference in sensitivity was deemed to be an alteration of the

prediction interval by 10%. This method is referred to as 'RCID-sensitivity'.

The next method was to combine data from similar detectors in order to utilise greater amounts of training data. It was thought that this increase in training data would produce more accurate forecasts, particularly at times of rarely occurring contexts, and hence allow RCID to perform better. This method is in contrast to the previous method, in that it looks for similarities between pairs of detectors. Two methods were developed to judge whether detectors' training data should be combined. The first was to use the driving distance, in the same way as was described in section 7.3.1.3. This method is referred to as 'RCID-spatial'. The second was to find the statistical correlation between the two sets of training data, and to combine the training data of detectors with a correlation within a pre-set threshold. This method is referred to as 'RCID-correlation'. A modification of this method with the correlation measured with a five minute time-period shift was also developed, which was designed to consider correlations where conditions took a certain amount of time to propagate, e.g. flow increasing from a stadium to the edge of a city after an event. This method is referred to as 'RCID-correlationTS'.

Hard-coded methods based on traffic theory or assumptions of traffic behaviour, such as upstream/downstream approaches, were also considered. These methods rely on predictable behaviour of traffic conditions between spatially nearby detectors. However, section C.2.2 demonstrated how unpredictable such traffic conditions can be in urban networks. Another downside is that detectors in the Southampton dataset were most often placed at the entry and exit of signalised junctions (because they were intended primarily for use in traffic signal management), meaning the behaviour between spatially nearby detectors is likely less predictable than similarly distanced detectors on motorways. Based on these factors, such hard-coded methods were not developed in this study.

C.2.4 Results

Figure C.7 shows the performances of the various methodology alterations. It should be noted that only the versions of RCID with contextual data were included in the results for clarity.

The difference made by each of the alterations was minimal. Firstly, RCID-sensitivity was typically below the curve of the original version of RCID. Comparing the FAR and DR of RCID-sensitivity to the original version of RCID at each prediction interval setting, RCID-sensitivity had a higher false alert rate but also equal or higher detection rate at every setting. For example at a 90% prediction interval, RCID-sensitivity had an equal DR (97%), but a 0.27% higher FAR (1.75%). This difference was expected because the modification made RCID more sensitive when nearby detectors were raising alerts. However, the increase in detection rate was clearly not worth the trade off of the increase in false alert rate, because the curve was typically lower than that of the original version of RCID. The difference in performance may also have been related to the fact that only 5 of the 37 incidents affected multiple detectors. As such, many increases in sensitivity at detectors would have been unwarranted due to the nearby detector being unaffected.

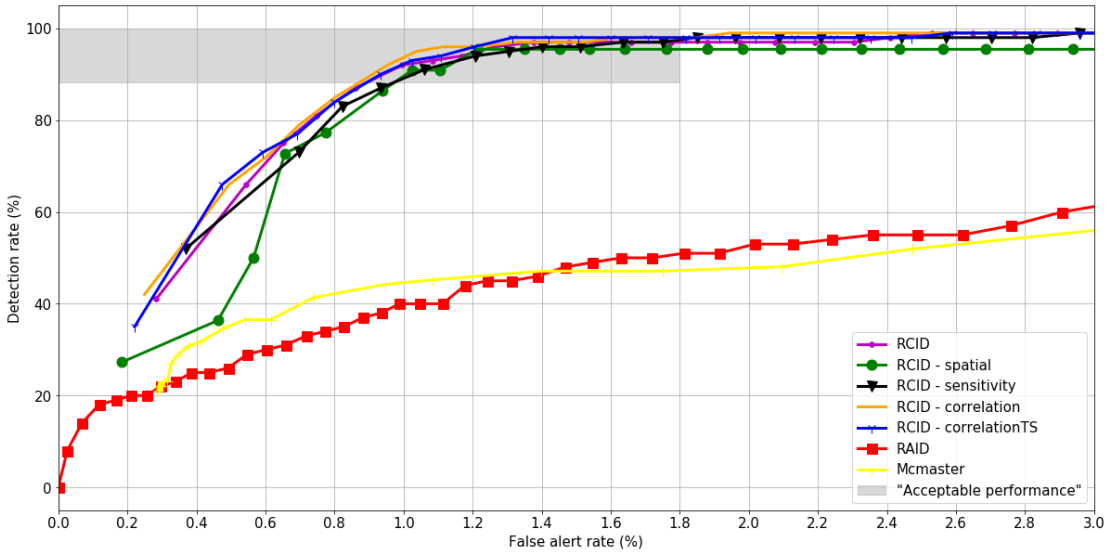


FIGURE C.7: RCID with various spatial alterations, false alert rates and detection rates. The grey area represents bounds for which Ritchie and Abdulhai (1997)'s survey of TMC operators deemed 'acceptable performance' (Ritchie and Abdulhai, 1997).

RCID-spatial was slightly below the curve of the original version of RCID for the majority of the curve, but was clearly lower at high prediction intervals. At high prediction intervals RCID-spatial achieved a better FAR but far worse DR, resulting in the clear difference in curves at this point. At a 99% prediction interval for example, RCID-spatial had a 0.46% FAR and 36% DR, whereas the original RCID had a 0.54% FAR and 66% DR. At lower prediction intervals however (90% and lower), the FAR and DR remained similar.

It is thought that this difference was caused by a reduction of certainty in RoadCast's forecasts. By combing nearby detectors training data, any differences in the patterns of traffic conditions between the detectors would result in prediction intervals that cover the differences. Take for example a pair of spatially nearby detectors, where the second has a far lower flow profile on Sundays due to it being on the turning into a school. Incidents that cause low flow due to congestion may go undetected at the first detector on Sundays, because RoadCast would produce wide prediction intervals that account for the second detector's typical traffic profile on this day.

At high prediction intervals, RCID was prone to using anomalous values to form its prediction intervals, particularly if unrepresentative data (such as incident disruption) was in the training data. Also, when spatially nearby detectors have differences in typical traffic profiles, the range of the prediction interval produced by RoadCast is expected to be wider. As such, RCID-spatial may have been more likely to include anomalous values (that could be present in any group of spatially nearby detectors) at high prediction intervals. The combination of these factors may explain why RCID-spatial had a lower DR when using high prediction intervals.

RCID-correlation had a very similar values of FAR and DR to RCID throughout the curve. Comparing the FAR and DR at each prediction interval setting, RCID-correlation typically had a worse DR and better FAR. At a 90% prediction interval, RCID-correlation had an equal DR

(97%), but 0.04% higher FAR (1.48%). However, the differences between the methods were so small that it is difficult to conclude whether one method has an advantage. Unlike RCID-spatial, RCID-correlation did not show a drop in detection rate at high prediction intervals compared to RCID. This is likely due to detectors being paired based on their correlation rather than location, meaning that pairs of detectors would have fewer differences in traffic profiles. RCID-correlationTS had negligible difference to RCID-correlation throughout the curve. At a 90% prediction interval, RCID-correlation had a 97% DR and 1.44% FAR, and RCID-correlationTS had a 98% DR and 1.42% FAR. This shows that comparing the correlations between detectors with a five minute difference was not necessary.

As was the case with the methodological changes, it is noted that the alterations made may have caused changes in other aspects of performance. However, it is expected that these changes would have minimal impact on RCID's average time to detect, due to each of the methodologies requiring three messages for the persistence test.

C.3 Summary

This section presented a number of alterations to the methodology of RCID, with the aim of improving the performance it achieved in the initial offline test in section 6.5. Each altered version of RCID was implemented and evaluated on the same dataset as in the initial offline test, and compared to the original version of RCID. Alterations included changes to the incident detection methodology, as well as changes that aimed to take advantage of spatial patterns of incident disruptions and other variations in traffic conditions.

None of the alterations made resulted in a clear improvement in the performance of RCID in terms of FAR and DR. As such, it was recommended that the original version of RCID would be most suitable for implementation. The original version has the simplest methodology, the fewest requirements to implement, and performed best in this offline test.

Appendix D

Bristol TMC operator interview questions

This appendix states the questions which were used for the operator interviews undertaken for the online test of RCID.

D.1 Pre-test interview questions

1. What methods do Bristol's TMC employ to detect incidents?
2. What methods do Bristols TMC employ to respond to incidents?
3. Has an IDA been implemented in Bristols TMC? If yes, the following apply:
   (a) Is the IDA being used currently?
   (b) How does the IDA detect incidents?
   (c) What type of detectors does the the IDA use?
   (d) What traffic variables are used?
   (e) Are you satisfied with the IDA?
   (f) How does your IDA aid the TMC's task of incident detection?
   (g) What are the benefits and drawbacks of the IDA?
   (h) Can you compare the IDA to others implemented previously? (if yes, go to question (a))
4. What role should RCID play to best aid your task in detecting incidents?
5. What are the most important features that RCID should have?
6. What would acceptable detection rate, false alert rate and average time to detect be? (definitions were provided)

Questions 1-3 aimed to understand how incidents were detected and managed in Bristol's TMC. This information would provide context and reasoning for how RCID could best aid the TMC, and would help to understand the operator's reasoning for later questions, such as what features are most important.

Question four and five struck at the heart of the aim of this interview. The answers would help to judge whether RCID was suitable for use in Bristol's TMC, and in what way it could benefit it. The answers would also influence the design decisions behind the interface that would display and receive feedback of RCID's alerts. Finally, the answers would be compared with answers to the post-test interviews, in order to understand whether RCID met the expectations that operators had before the trial.

Question six was asked to understand what performance level would be required by RCID to benefit the TMC. This would be used to assess RCID's suitability for use in Bristol's TMC, and would influence decisions related to the sensitivity of the algorithm during the trial.

D.2 Post-test interview questions

1. How was the RCID algorithm used?
2. In what ways was it useful (if any)?
3. In what ways could it have been more useful (if any)?
4. Was the false alert rate/detection rate/mean time to detect sufficient?
5. How did it compare to the currently used incident detection methods?
6. Would you choose to use it in your daily work (without the feedback)?
7. Any other comments on the experience using RCID?

The rationale behind question one was to understand in what way RCID was used. For example, was it used as the first indicator of an incident, to then be verified by manually checking CCTV, as was suggested in the pre-test interviews?

Question 2-4 aimed to understand the performance of the IDA from the perspective of the operators. Given that performance metrics could not be acquired in this test, the answers to these questions, along with the real-time feedback on the web application, would be used to assess the performance of RCID.

Question five asked operators to compare RCID with previously used IDAs, if any. This question would be used together with question three of the pre-test interviews to understand how RCID compared to other IDAs in practice.

The purpose of question six was to understand whether operators found RCID useful and provided a benefit to Bristol's TMC. If the experience was positive enough for operators to be willing to use the IDA in the future, this would be an indication that it is useful to the TMC.

Appendix E

Publications

Significant parts of this research project has been published in journal articles and presented at academic conferences, or will be in the future. Siemens were closely involved in the research project throughout and are currently planning to implement the algorithm developed so far (RoadCast). The following papers were published in journals or presented at conferences:

Evans, J., Waterson, B., Hamilton, A., 2018. RoadCast: An algorithm to forecast this year's road traffic. Transportation Research Board 97th Annual Meeting. Washington D.C., United States. 7th 11th January 2018. Poster presentation at the conference.

Evans, J., Waterson, B., Hamilton, A., 2019. A random forest incident detection algorithm that incorporates contexts. In proceedings of the 15th World Conference on Transport Research. Mumbai, India. 26th 31st May 2019. Oral presentation at the conference.

Evans, J., Waterson, B., Hamilton, A., 2019. Forecasting road traffic conditions using a context-based random forest algorithm. Transportation Planning and Technology (42, 6, p554-572).

Evans, J., Waterson, B., Hamilton, A., 2019. A random forest incident detection algorithm that incorporates contexts. International Journal of ITS Research, doi:10.1007/s13177-019-00194-1.

Evans, J., Waterson, B., Hamilton, A., 2019. The evolution and future of urban road incident detection algorithms. ASCE's Journal of Transportation Engineering, Part A: Systems. Accepted, awaiting publication.

It should be noted that the methodology of RCID, presented in chapters 5 and 7, is the final version of a methodology that has been iterated on throughout the study. Some of the published articles above may have used earlier iterations of the methodology, and so may have different results accordingly.