UNIVERSITY OF OKLAHOMA

GRADUATE COLLEGE

DEVELOPING AN ALGORITHM TO IDENTIFY SECONDARY INCIDENTS

A THESIS

SUBMITTED TO THE GRADUATE FACULTY

in partial fulfillment of the requirements for the

Degree of

MASTER OF SCIENCE IN ELECTRICAL AND COMPUTER ENGINEERING

By

VARUN GUPTA
Norman, Oklahoma
2018

DEVELOPING AN ALGORITHM TO IDENTIFY SECONDARY INCIDENTS

A THESIS APPROVED FOR THE
SCHOOL OF ELECTRICAL AND COMPUTER ENGINEERING

BY

_____
Dr. Hazem H. Refai, Chair

_____
Dr. Thordur Runolfsson

_____
Dr. Kam Wai C. Chan

Dedication

To my parents,

Rani and Chaman,

&

my beloved sisters,

Pallavi and Shilpi

Acknowledgements

I would like to thank my advisor and mentor Dr. Hazem Refai for his support, guidance, and insights without whom this work would not have been possible. I want to thank Dr. Thordur Runolfsson and Dr. Kam Wai C. Chan for their time to review this thesis. I also want to thank my friends at the University of Oklahoma for their encouragement and kindness.

Table of Contents

Acknowledgements ..... **iv**

List of Tables ..... **vii**

List of Figures ..... **viii**

Abstract ..... **x**

Chapter 1: Introduction **1**

1.1 Secondary incidents – Definition 1

1.2 Defining the spatial and temporal boundaries 2

1.3 Creating Graphical User Interface 3

1.4 Creating statistical and artificial neural network models 4

1.5 Understanding the importance of different features 4

1.6 Contribution of the thesis 5

Chapter 2: Related Work **7**

Chapter 3: Parameters and Development of Graphical User Interface **11**

3.1 Dataset acquisition and processing 11

3.2 Algorithm parameters 13

3.2.1 Queuing – How vehicles behave on roads 14

3.2.2 Lane blockage – Effect of incidents on highway capacity 17

3.2.3 Traffic intensity – Distribution of average daily traffic over time 19

3.2.4 Changing Lanes behavior of vehicles in case of an incident 21

3.2.5 Spatial and temporal influence of the incident 22

3.3 Graphical User Interface 24

Chapter 4: Statistical and Artificial Neural Network Modelling **29**

4.1 Dataset preparation 29

4.2 Statistical models for classification of incidents 31

4.2.1 The Logit Model 31

4.2.2 The Probit Model 32

4.3 Artificial Neural Network 33

4.4 Results - Statistical and Artificial Neural Network Models 35

4.4.1 Results from the Logit Model 36

4.4.2 Results from the Probit Model 38

4.4.3 Results from the Artificial Neural Network Model 40

Chapter 5: Factors Influencing Secondary Incidents 42

5.1 Connection Weight Algorithm 42

Chapter 6: Conclusion and Future Work 46

References 48

Appendix A: Neural Network Connection Weights 51

List of Tables

Table 1. Probabilities of number of lanes closed 19

Table 2. Features used for modelling and attributes..... 30

Table 3. Coefficients and odd ratio for Logit Model..... 36

Table 4. Confusion matrix legend 37

Table 5. Coefficients and odd ratio for Probit Model 39

Table 6. Comparative analysis of different models..... 41

Table 7. Input to hidden connection weights..... 43

Table 8. Hidden to output connection weights..... 43

Table 9. Product of input-hidden connection weights to hidden output connection weights 44

Table 10. Variable importance by Connection Weight approach 44

Table 11. Legend for weather conditions..... 45

Table 12. Legend for lightning conditions 45

List of Figures

Figure 1. Representation of secondary incident 2

Figure 2. State highways of city of Tulsa 12

Figure 3. Highways of state of Oklahoma 13

Figure 4. The process to spatiotemporal area of influence 14

Figure 5. Vehicles moving on a two-lane highway 15

Figure 6. Inter-arrival time vs. number of vehicles 16

Figure 7. Traffic intensity at various times of the day 20

Figure 8. Movement of vehicles from affected lane to adjacent lanes 21

Figure 9. Code for lane changing feature in algorithm 22

Figure 10. Spatiotemporal influence area of an incident 23

Figure 11. Primary and secondary incidents plotted in Google Maps 24

Figure 12. Frame 1 of the GUI 25

Figure 13. Frame 2 of the GUI 25

Figure 14. Frame 3 of the GUI 26

Figure 15. Frame 4 of the GUI 26

Figure 16 Frame 5 of the GUI 27

Figure 17. Incidents on Google Maps 28

Figure 18. Correlation matrix for various features in the dataset 31

Figure 19. A simple ANN 34

Figure 20. ROC for logit model 38

Figure 21. ROC for probit model 39

Figure 22. Loss vs. epochs for ANN 40

Figure 23. Accuracy vs epochs for ANN 40

Figure 24. ROC for ANN 41

Figure 25. Artificial Neural Network 43

Figure 26. Independent variable influence for secondary incident detection..... 45

Abstract

Highways are the most frequently used means of transportation in today's world and the leading source of travel mishaps. Crashes or incidents on highways—both primary and secondary—constrain highway capacity, threaten passenger safety, and increase travel time, resulting in delays and wasted traffic management resources. This thesis aims to expand field knowledge about the detection of secondary incidents by analyzing primary incidents and their spatiotemporal influence on traffic.

Analytical and statistical methods including logit, probit, and artificial neural network models were designed for automating incident classification by processing vehicle count, weather conditions, and traffic flow, among other parameters. The logit and probit model showed similar performance with an accuracy of 67% in the former and 66% in the latter and an identical precision of 48%. The contribution of each independent feature was gauged using odds ratio. The artificial neural network (ANN), on the other hand, out-performed the logit and probit model. A simple 3-layer ANN was used for incident classification which showed an accuracy of 91% and a precision of 89%. The improved performance of ANN can be attributed to its ability to learn complex relations.

A novel connection-weight algorithm was then used to determine the importance of the various features on the dependent variable and how they affect the model. Results were encapsulated in a graphical user interface for facilitating data collection and analysis.

Chapter 1: Introduction

Highway incidents (or 'crashes') handicap the networked transportation system by restricting traffic flow, threatening vehicle passenger safety, causing extended vehicle queues, lengthening travel time, and expending traffic management resources. Although secondary incidents (i.e., those occurring subsequent to and within the spatiotemporal area-of-influence of a primary incident) occur less frequently than primary incidents, they are often severer and amplify consequences of an incident. This research focuses on incidents classified as rear-end and side-swipe 'crashes.' This term will be used interchangeably with the term 'incidents'.

1.1 Primary and Secondary Incidents

Highway incidents are classified as *primary* or *secondary.* The former occurs in a free-flowing traffic and are attributed to human or vehicle error. Often, they cause lane blockage, restrict traffic flow, and reduce highway capacity. The latter occur in the spatiotemporal area-of-influence of the former, posing further constraints on the highway network system. Depending on the extent of damage to and number of vehicles involved in an incident, lane blockage is possible and could affect traffic flow accordingly. Limited roadway capacity often causes queues of vehicles to build up increasing travel time. The effect of a primary incident can be interpreted in terms of distance (d) and time (t)[1], where d is the length of queue formed as a result of compromised capacity and t is the average time a vehicle remains in the queue. Figure 1 demonstrates the continuum of a primary incident, followed by a secondary incident occurring at a given distance and time.

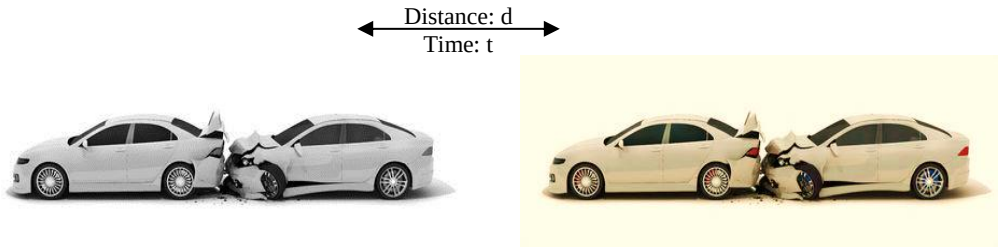

Figure 1. Representation of secondary incident following a primary incident.

1.2 Defining the spatial and temporal boundaries

To fully understand the spatiotemporal area-of-influence of a primary crash, the typical movement of vehicles on highways should be considered. In free-flowing traffic conditions, vehicle *service time* (i.e., time to travel a specified section of highway) is less than the inter-arrival time of vehicles. This actuality ensures that traffic moves at an optimal speed and no queues should exist. Subsequent to a primary crash, vehicle service time in the affected section of highway increases as vehicles slow down and change lanes to avoid affected roadway lanes. Inter-arrival time remains relatively constant. However, any increase in service time results in queue-formation, which can be interpreted as an effect of the primary incident in terms of the distance (*d*) moving traffic is affected. Queue length can be calculated by multiplying the number of vehicles in the queue by the average vehicle length. This thesis work is based on the average car class of vehicles, measuring up to 4.1 meters in length.

Likewise, an effect of the primary incident can be measured in terms of time (*t*), Time (*t*) can be expressed in terms of the amount spent in a queue—where traffic is not moving—and expressed as reporting time, response time, vehicle clearing time, and time taken to resume normal flow of traffic. Together, distance (*d*) and time (*t*) express the maximum *spatiotemporal area-of-influence* of a primary incident.

The highway travel concepts defined above can further explained with the use of vehicle movement simulations that incorporate various effects (e.g., vehicle slow down, number of lanes blocked, vehicle lane change from those affected by the crash to in service lanes).

1.3 Creating Graphical User Interface

A graphical user interface (GUI) was created to encapsulate the process of utilizing information from Oklahoma Department of Transportations (ODOT) databases to calculate spatiotemporal area-of-influence for a given incident. The GUI facilitates data collection and research by automating the process of fetching data from the database and plotting the spatiotemporal area-of-influence relative to a primary incident and other incidents in proximity.

The multi-frame GUI leverages Tkinter library in Python. Dynamic plots were incorporated, using zoom in/out and save-image functionality. A Google Map API was also implemented to plot incidents onto a Google Map, aiding in the depiction of traffic flow directionality affected by an incident. The resulting dataset—based on the GUI representation of primary and secondary incidents—was then used to generate automated models for classifying incidents.

1.4 Creating statistical and artificial neural network models

After manually creating the dataset of incidents and classify an incident either a primary or secondary, the next step was to automate the process of classification, given a set of parameters. This will help us in having a better understanding of secondary incidents and their causative factors [12] [13]. Statistical models including logit and probit were used in the analysis. The dependent variable for the models is categorical in

nature. Notably, logit models are widely used in the fields of health sciences and ecology; probit models are typically used for econometrics.

Furthermore, a simple, three-layered, ANN was used for modeling the classification algorithm for primary and secondary incidents [21]. Its use enabled the ability to learn and model non-linear complex relationships, such as those occurring in real-life, input and output events of a complex nature. Also, the predictive capabilities of a neural network facilitate an inference of classes in unseen data.

1.5 Understanding traffic factors influencing secondary accidents

A connection weight algorithm was used [16] to understand how different features used as input to ANNs affect the classification. While neural networks provide little explanatory insight into independent variables and their effects on incident classification, the connection weight algorithm uses input-hidden weights and hidden-output weights to quantify variable importance.

1.6 Contribution of the thesis

This thesis presents details to enhance the understanding of secondary incidents by using the data provided by the Oklahoma Department of Transportation (ODOT) Accident and Incident databases. A process was developed to analyze various chosen features in a database for gauging the effects of a primary incident on a highway network system and plot the incident's spatiotemporal area-of-influence.

Major contributions of the thesis include:

- The proposed process can be applied to any transport-related data, as it clearly outlines the process for using available traffic data to gauge the spatiotemporal effect of a primary incident toward identifying secondary incidents.

- The lane-changing feature incorporated into the algorithm is unique (to the best of the author's knowledge) and incorporates the effect vehicles that are changing lanes following an incident has on incoming traffic and how it increases congestion on lanes still in service.
- A novel graphical user interface (GUI) was developed to encapsulate the process for measuring the spatiotemporal area-of-influence of a primary incident. Python's Tkinter was used as a backend for development.
- An approach was defined for decoding the neural network and gaining explanatory insight to lend an understanding of the importance of various traffic and road input features. Connection weight algorithm aided in gauging feature importance—the knowledge from which can be used to address such issues and move towards improved data collection.

The remainder of the thesis is organized into five additional chapters. Chapter 2 addresses related work that has previously been reported in this field. Chapter 3 explains the proposed process and its use for a) measuring the spatiotemporal area-of-influence of a primary incident, b) explaining how various parameters corresponding to the problem at hand were developed, and c) describing how the GUI functions for encapsulating the entire process and generating plots. Chapter 4 introduces the concept of using the data gathered via the GUI and automating the incident classification process using logit, probit, and ANN models. The chapter also highlights results for each method. Chapter 5 discusses the connection weight algorithm that was utilized for measuring feature importance using weights from a

neural network. Finally, Chapter 6 provides concluding thoughts and forecasts future work.

Chapter 2: Related Work

The work in this thesis derives inspiration from the field of transport engineering machine learning. Chapter 2 provides an overview published work that promotes an understanding of secondary incidents as an effect of primary incidents. Moore, *et al.* [1] explained that secondary incidents are defined as those that occur within a predefined spatiotemporal region of a primary incident and cause reduced roadway capacity. These types of incidents are common on highways, restricting highway capacity and initiating traffic delays. When compared with primary incidents, secondary incidents are typically severer. Hence, identifying and understanding secondary incidents facilitates more efficient use of traffic control resources whilst increasing highway safety [3].

Although early research in the field of secondary incidents tends to utilize pre-defined spatiotemporal boundaries, it has failed to incorporate various features of incidents or consider their dynamic nature.

Raub (1997) [2] presented an algorithm for spatiotemporal analysis of secondary incidents in urban highways, assuming a fifteen-minute clearance time and one-mile spatial effect. The study failed to incorporate features like incident type, highway traffic, number of vehicles involved etc. Effect was measured for a distance of only 1,600 meters (one mile). If an accident occurred within this fixed spatial threshold, the accident was considered secondary. Later, Moore, *et al.* (2004) [1] improved this approach, incorporating directionality of incident and a queueing mechanism. Moore conducted his research using data collected on Los Angeles freeways with special data resources and continued using the static spatial boundary for defining spatiotemporal area-of-influence.

Zhan, *et al.* (2008) [8], developed a method for defining dynamic boundaries for primary incidents using incident and traffic data. The method was based on a cumulative arrival and departure traffic model for estimating queue length and traffic delays while considering lane blockage during the process. The result provided a superior method for determining spatiotemporal area-of-influence and incorporating real-world features, queueing highway vehicles on highway, and considering lane blockage.

Zhang and Khattak (2010) [3] introduced a dynamic queue method to aid in understanding the distant effect of a primary incident. Queue length was determined by leveraging the deterministic D/D/1 model based on average traffic traveling on a highway. Sun and Chilukuri (2010) [4] suggested using video-based traffic data caused by secondary incidents for determining the threshold of spatiotemporal area-of-influence. An incident progression was suggested based on the incident severity and the volume over capacity ratio of highway traffic [16]. Kerner, Rehborn, Aleksic, and Haug (2004) [5] used Automatic Jam Recognition and Forecasting for traffic objects based on Kerner's Three Phase Traffic Theory. This method for detecting traffic jams and their spatiotemporal influence is completely dependent on live, video data of the traffic jam.

To improve on the system of queueing and to better understand the dynamic nature of primary incidents, this thesis presents a method to accommodate the dynamic system of queuing by using the memoryless M/M/1 scheme acting on the data obtained from ODOT Accident database. Poisson arriving rate in M/M/1 is better descriptive of traffic than the average arriving rate used in D/D/1 models. The truncated Poisson-like process, furnished with the various features of traffic data, is shown to mimic real-world behavior of vehicles traveling on highways.

Karlaftis, et al. (1999) [6] suggested that clearance time, season, vehicle type, and lateral location of the primary crash are significant factors.

Karlaftis, *et al.* (1999) [6] examined primary crash characteristics that influence the likelihood of secondary incident. The authors suggested that clearance time, season, vehicle type, and lateral location of the primary crash were critical factors for predicting a secondary incident.

Vlahogianni, *et al.* (2010) [18] introduced a Bayesian framework for combining crash and queue information, suggesting a dynamic range for distance and time effect following a primary highway incident. Later that year, Zhang and Khattak (2010) [3] developed a logit model for determining the relationship between primary and secondary incidents. Their work reported an important finding that duration of primary incident, lane blockage, and vehicle number involved in a primary incident were the most influential factors in determining a secondary incident. Karlaftis and Vlahogianni (2011) [18] used the logit model to establish a relationship between primary and secondary incidents. The researchers also considered the use of ANN [9] for efficient predictive models for this type of classification. Tu (1996) [10] summarized the advantages of neural networks over logit models, suggesting that although neural networks provide acceptable results and robust models, they are not sufficient for modeling. Statistical models (e.g., logit and probit) provide necessary explanatory data.

Neural network use for establishing a relationship between primary and secondary incidents continue to perplex the research community. Vlahogianni, Karlaftis, and Orfanou (2002) [18] suggested using mutual information—a method used to determine variable information before training the network. Though useful, this method does not

explain the effect of variables on classification after training the neural network. To solve this problem, this thesis uses the connection weight algorithm introduced by Olden, Joy, and Death, (2003) [20], which established the relative importance of each variable using the final weights of the neural network.

Chapter 3: Parameters and Development of a Graphical User Interface

Chapter 3 details three important facets for developing an algorithm to plot the spatiotemporal effects of a primary incident. Sections below provide information necessary for understanding the framework required for processing the data collected on various state highways in Oklahoma. Various limitations and challenges associated with utilizing data in the Oklahoma Department of Transportation (ODOT) Accident and Incident databases are discussed. Parameters for and steps involved in algorithm creation are described, followed by a description of how the entire process was encapsulated using a Graphical User Interface (GUI) created in Python.

3.1 Dataset acquisition and processing

Data records of various state highways in Oklahoma were obtained from ODOT databases. Two divisions were instrumental in providing the data—Traffic Engineering Division [Accident database] and Intelligent Transportation System (ITS) Division [Incident database]. The former was composed of all incidents occurring in the state of Oklahoma and was accessible via an online portal. Individual or query-related records functionality allowed easy access to incident records reported by county or city. The present study was based on a total of 65,000 incident records from 2014 in the city of Tulsa, Oklahoma.

The latter (i.e., Incident database) was composed of 3,026 records collected between 2014 and 2015 from a variety of highways throughout the state, including incident duration, number of lanes closed, and directionality of lane closure. Notably, the Accident database is more comprehensive than the Incident database. Combined, the two

databases provide a comprehensive view of incidents that occurred in the state of Oklahoma. Databases were combined using a simple correlation detailed in section 3.2.2. The highways network structure for the cities of Tulsa and Oklahoma City are presented below (See Figures 2 and 3).

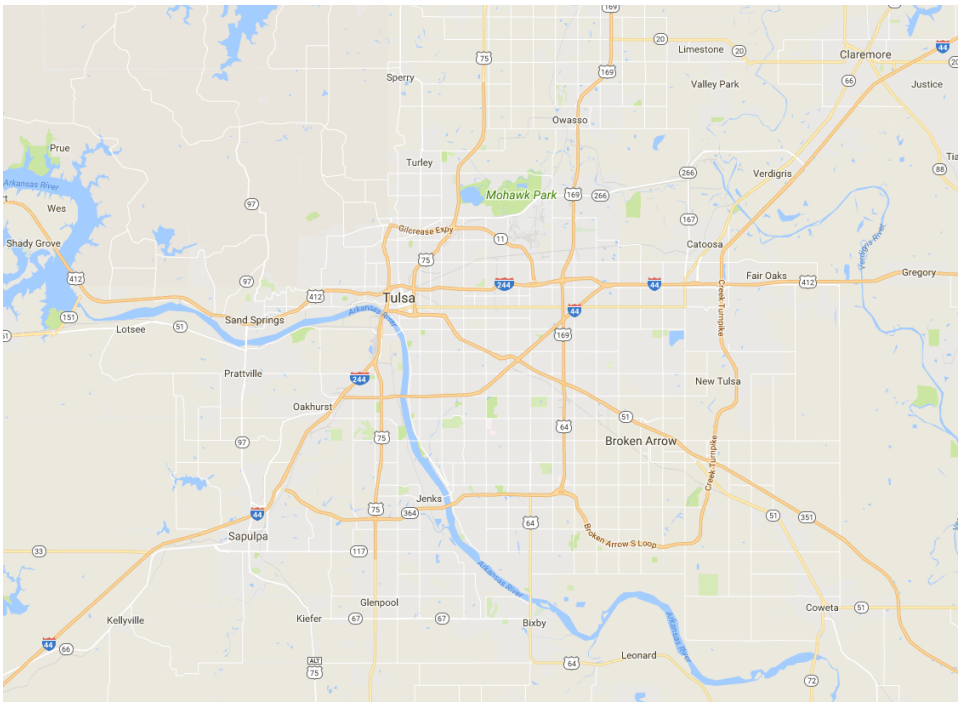

Figure 2. State highways in the city of Tulsa

Figure 3. Highways in the state of Oklahoma

Database features were used to determine characteristics of primary incidents, in terms of the effect on distance (*d*) and time (*t*), with the goal of detecting secondary incidents. Determining these is dependent upon various parameters, which are explained below.

3.2 Algorithm parameters

This section introduces the internal working of the newly developed algorithm and various associated parameters. Distance (*d*) and time (*t*) are subject to understanding and calculating multiple parameters. When combined in a GUI, this information generates a plot that provides potential maximum spatial and temporal boundaries from-the primary incident under-test. These are essential for determining the likelihood of secondary incidents.

The following parameters are important for such predictions.

- Queuing—ways in which vehicle drivers respond to highway incidents, and how vehicles queue up on highways post incidents.
- Lane blockage and its effects—primary incident impact on highway capacity.

- Time distribution of traffic flow—the effect of traffic intensity at different times during a day and how it affects the movement of vehicles on roads
- Lane changing behavior—driver's lane-changing behavior
- Spatiotemporal effect—distance and time boundaries with respect to the primary incident

These parameters can be used to generate average vehicle wait time and queue length for the highway network system (See Figure 4).

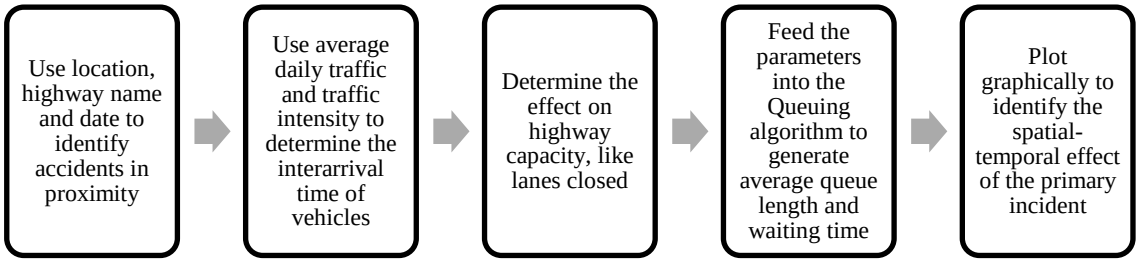

Figure 4. The process to generate queue length and average waiting time

3.2.1 Queuing

Capturing vehicle movement on the roadway is important for increasing algorithm effectiveness and can be accomplished by simulating the inter-arrival times of vehicles in a real-world situation. Deterministic and memoryless queueing models were leveraged to assess relatedness of detecting secondary incidents. In the more prevalent deterministic queuing model, inter-arrival times remain constant for all vehicles, whereas in the memoryless queuing model, inter-arrival times are not mutually dependent. In this study, vehicle arrival rate on highways resembled a memoryless model [M/M/n], where the number of servers or equals the number of highway lanes and each lane is an independent queue. The memoryless model of queuing used in the algorithm defines a memoryless interarrival rate for the vehicles in the system and a memoryless servicing time.

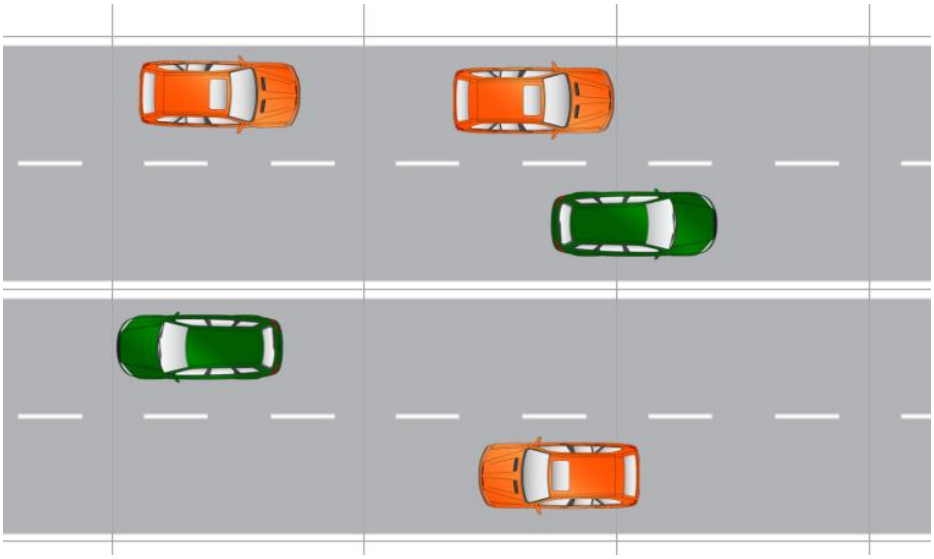

Figure 5. Vehicles moving on a two-lane highway

Highway vehicles move in a pattern similar to a truncated Poisson process. At any given point in the day, arrival rate of subsequent vehicles is not dependent on the arrival rate of the present vehicles. However, inter-arrival time is generally bound by an upper and lower limit, hence the name truncated Poisson. The plot in Figure 6 shows the distribution of vehicle inter-arrival times used in simulations. These ranged from 4 to 5s for incoming vehicles. This feature aided in simulating a real-world vehicle arrival pattern.

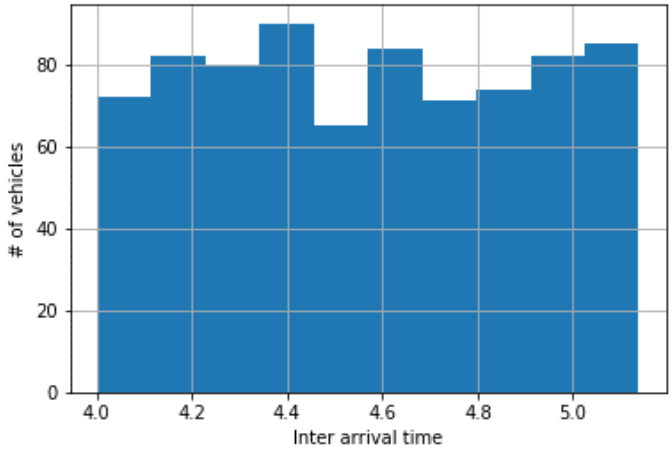

Figure 6. Inter-arrival times versus number of vehicles

Results of the queuing model provided average length of queue (d) and average time a vehicle spent in the queue (t). Both variables, *d* and *t*, are important for establishing the spatiotemporal boundaries of a primary incident. Variable values could be used to identify and detect the likelihood of secondary incidents.

The mathematical overview of the queuing process used in this research is presented below. M/M/c queuing variables are defined, as follows.

- $Poisson\ arrivals\ with\ rate\ \lambda$
- $Exponential\ service\ times\ with\ parameter\ \mu$
- $c\ servers\ (highway\ lanes)$
- $Arriving\ customers\ (vehicles)\ finds\ n\ customers\ (vehicles) in\ the\ system$
    - $n < c: it\ is\ routed\ to\ any\ idle\ lane$
    - $n \geq c: it\ joins\ the\ waiting\ queue -$

      $when\ all\ lanes\ are\ busy\ (blocked)$
- $P_Q = Probability\ of\ queueing$
- $p_o\ and\ p_n\ are\ the\ stationery\ distributions$

$$Expected\ number\ of\ vehicles\ waiting\ in\ the\ queue - not\ in\ service$$

$$N_Q = \sum_{n=c}^{\infty}(n-c)p_n = p_o\frac{(c\rho)^c}{c\,!}\sum_{n=c}^{\infty}(n-c)\rho^{n-c} = p_o\frac{c\rho}{c!}\frac{\rho}{(1-\rho)^2}$$

$$= P_Q(1-\rho)\frac{\rho}{(1-\rho)^2} = P_Q\frac{\rho}{1-\rho}$$

$$Average\ waiting\ time\ in\ queue$$

$$W = \frac{N_q}{\lambda} = P_Q\frac{\rho}{\lambda(1-\rho)}$$

$$Average\ time\ in\ system\ (queued + serviced)$$

$$T = W + \frac{1}{\mu}$$

$$Expected\ number\ of\ customers\ in\ the\ system$$

$$N = \lambda T$$

The expected number of vehicles waiting in queue multiplied by average vehicle length (e.g., a sedan in this study) determines average length of the queue (d) and establishes the spatial effect of an incident.

3.2.2 Lane blockage – Effect of incidents on highway capacity

Understanding the effect of incidents on highway capacity is important for determining the effect on the ability for vehicles to travel without delay, or, in other words, determining the number of vehicles serviced by a highway in a given unit of time. An increase in service time results in a longer queue, hence a longer wait for vehicles within the highway network system at the time of an incident. Such a phenomenon effectively increases the area of spatial and temporal effects of a primary incident.

The Accident database was unable to provide information concerning the effect of a primary incident on highway capacity or lanes blockage. The Incident database, however, was able to do so, although data was limited to 623 instances in Tulsa county. Information garnered from the Incident database lacked the comprehensiveness of that in the Accident database. Hence, it was necessary to fetch time and location parameters from both databases to identify a total of 42 incidents that were common in both sets.

To establish a correlation between various features (e.g., number of vehicles involved in an incident, number of highway lanes, number of lanes blocked, and extent of vehicle damage [ODOT specified], an association analysis that considered rear-end and side-swipe incidents was used. Association analysis [14] is a rule-based machine-

learning method for discovering interesting relations between variables in datasets. The intended purpose is identifying strong rules in databases using some measure of interestingness.

The support, confidence, and lift of a rule determine the extent of its correctness. Given the following problem: *Let X be an itemset, $X \Rightarrow Y$ an association rule, and T a set of transactions of a given database*, features include:

Support: An indication of how frequently a given itemset appears in a dataset. The support of X with respect to T is defined as the proportion of transactions t in the dataset that contains the itemset X.

$$supp(X) = \frac{|\{t \in T; X \subseteq t\}|}{|T|}$$

Confidence: An indication of how often the rule has been found true; interpreted as the probability of finding the right-hand side of the rule in transactions, given that the transactions also contain the left-hand side.

$$conf(X \Rightarrow Y) = \frac{supp(X \cup Y)}{supp(X)}$$

Lift: the ratio of observed support for the rule to that expected, given that the antecedent and consequent are independent.

$$lift(X \Rightarrow Y) = \frac{supp(X \cup Y)}{supp(X) * supp(Y)}$$

Using the above-mentioned measures, rules were extracted to determine the effect of an incident on highway capacity. Parameters included the number of vehicles involved in an incident; the number of lanes on the highway; and the extent of damage. Once rules were determined, the total number of times the rules were followed in the Incident database were calculated. For example, when the number of vehicles involved in an

incident equaled two and the damage extent was less than three i.e. functional damage to the vehicle, there was a 45% chance that one lane was closed. Similar calculations were executed for various numbers of vehicles—ranging from one to greater-than-or-equal-to-four—and damage extent ranging from one to five, as defined in the ODOT user dictionary.

No. vehicles	lanes_closed_1	lanes_closed_2	lanes_closed_3	lanes_closed_4
1	0.98	0.02	0	0
2	0.45	0.54	0.01	0
3	0.31	0.49	0.19	0.01
4 or more	0.01	0.6	0.3	0.09

Table 1. Probabilities of the number of lanes closed

The table above represents various probabilities that relate the number of lanes closed in an incident with the numbers of vehicles and lanes. Results can then be used in the algorithm for calculating the spatial and temporal effects of a primary incident, as well as determining average queue length and average waiting time for a vehicle in the highway network system.

3.2.3 Traffic intensity – Distribution of average daily traffic over time

Traffic intensity is the number of vehicles travelling a highway at a given day time. This figure dictates vehicle inter-arrival time. The less the inter-arrival time is, the longer the queue build-up will be in the event of an incident, and vice versa. Results in this study were obtained utilizing data from a project conducted by the Washington State Transportation Center [16] in which an urban area was considered for observing and recording the distribution of traffic for 24 hours during a weekday. A bimodal traffic pattern was attributed to the geographic location of the area and roadway function. Specific peak height differed from location to location and was dependent on various

traffic generators. Two peak-traffic distributions were associated with directional commuter trips.

Traffic intensity measured at different times of the day aids the algorithm in determining inter-arrival rate limits for vehicles at different times of the day. The effect of traffic intensity on highway capacity indicates that when traffic intensity is high, inter-arrival times decrease. Hence, an accident with merely low damage extent could potentially result in long queue times, effectively increasing the time spent by a vehicle on the highway network system or a vehicle waiting to be serviced. During peak night hours, inter-arrival time for vehicles increases, resulting in shorter queues and reduced waiting times.

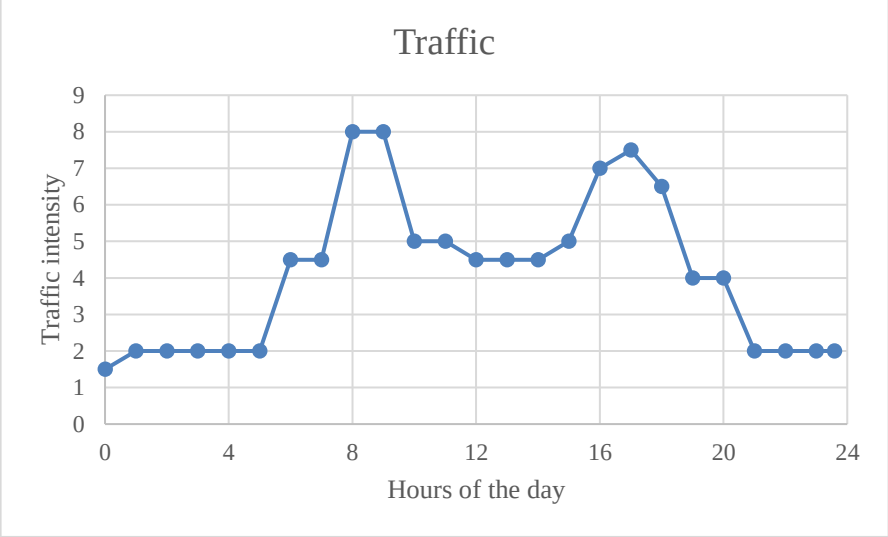

Figure 7. Traffic intensity at various times of the day

Traffic intensity can also be used to calculate inter-arrival times of vehicles on road as follows:

$$Traffic\ per\ hour\ = Average\ traffic\ (24\ hour) * Traffic\ intensity$$

$$Inter - arrival\ rate = \frac{3600}{Traffic\ per\ hour}\ seconds\ /\ car$$

The inter-arrival rate calculated above acts as a mode for the inter-arrival rates of vehicles, where the inter-arrival rate of the previous and next hour act as upper and lower bound.

3.2.4 Changing Lanes behavior of vehicles in case of an incident

Following lane blockage, the serving capacity of the lane is reduced to zero, causing a stationery queue to build up in that particular lane. Vehicles typically continue to travel even after an incident on the highway, moving away from the lanes which are blocked because of the primary incident to the lanes which are still in service. This effect increases the inter-arrival time of vehicles at the server and reduces their speed, contributing to a greater primary incident area-of-influence as longer queues are formed, and vehicles spend more time on the highway network system.

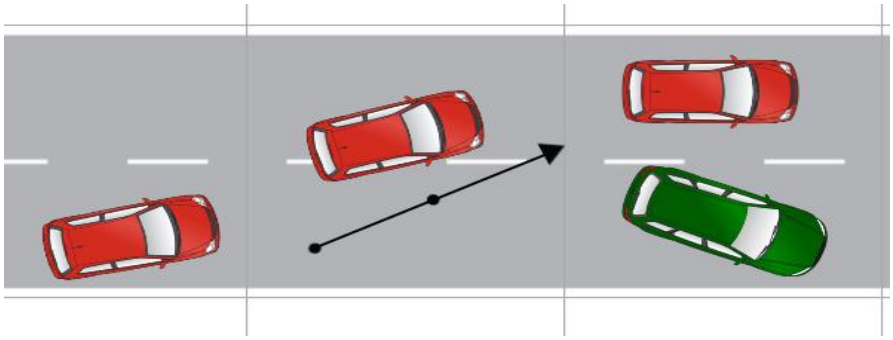

Figure 8. Movement of vehicles from the affected lane to an adjacent lane

This provides valuable insight about how vehicles move during an incident and its spatiotemporal effect of a primary incident. To simulate this factor in the algorithm, a custom arrival node was created in the CIW library [22] written in Python; it randomly pushes vehicles in the affected lane to an adjacent lane, thus effectively keeps traffic moving. A threshold of a single car was set for research purposes. For example, after an incident and the blocked lane has more than one car in queue, the systems begin to move incoming cars from the blocked lane(s) to lanes still in service. Pseudocode in Figure 9

shows that if an incident takes place in a two-lane highway and lane 1 is blocked, incoming vehicles in lane 1 are directed to move to lane 2 in an effort to keep traffic moving.

```
    # lanes = 2 and blocked lanes = 1
    class CustomArrivalNode21(ciw.ArrivalNode):
        def send_individual(self, next_node, next_individual):
            self.number_accepted_individuals += 1
            if ((Q.nodes[1].number_of_individuals) <= -1):
                Q.nodes[1].accept(next_individual, self.next_event_date)
            else:
                self.simulation.nodes[2].accept(next_individual,
self.next_event_date)
```

Figure 9. Code for lane changing feature in algorithm

The act of lane change can be fed into the algorithm with the aforementioned parameters to define the area of the spatiotemporal boundaries of a primary incident.

3.2.5 Spatial and temporal influence of the incident

Once all parameters have been accounted for and calculated—based on an incident's unique features, results are entered into the algorithm to determine the potential maximum spatiotemporal effect of a primary incident. The effect can then be plotted by leveraging the GUI for further analysis. Additional incidents within the area-of-influence can be considered secondary incidents.

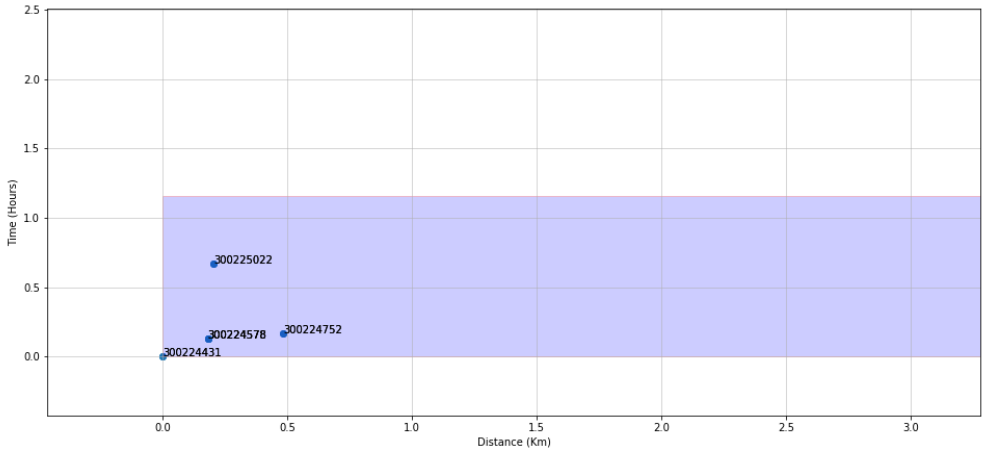

Figure 10. Spatial-Temporal influence of an incident

In the above example, incident ‘300224431’ was considered the primary incident. The shaded area represents the spatial-temporal area-of-influence of the primary incident, and other incidents in the shaded area may be considered secondary incidents. The primary incident occurred on Oklahoma Highway I-44 was a rear-end car crash involving two vehicles and the extent of damage was graded a 3. The incident resulted in one out-of-service lane for approximately 70 minutes, causing vehicle service time to increase. A queue build-up increased the time vehicles spent on the highway network.

Although this type of graphical representation provides adequate information about the spatiotemporal effect of an incident, it does not depict the direction of vehicle travel and affected highway sections. This problem was solved using the Gm-Plot library in Python in conjunction with the Google Maps API. Latitude and longitude coordinates contained in the Accident database were utilized for plotting the incident location on a Google Map. Coordinates are calculated up to the fifth decimal place and are accurate up to 1.1m. Using this method for studying the location of incidents provides insight about the highway section affected by the incident, as well as the direction of vehicle flow.

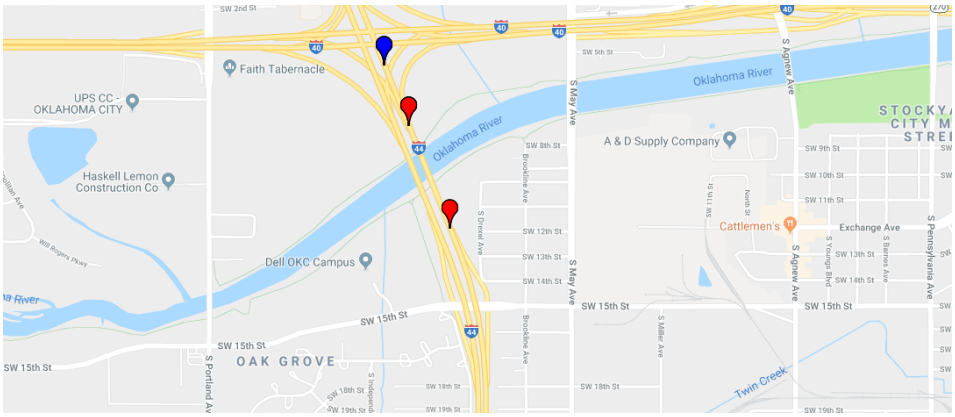

Figure 11. Primary and secondary incidents plotted in Google Maps (Primary-Blue, Secondary-Red)

Combining both, graphical representation aids in identifying secondary incidents and in locating them on a map.

3.3 Graphical User Interface

Figure 3.1 highlighted the process for encapsulating the GUI. The interface was created using Python with Tkinter as backend and facilitated the process of collecting data. The GUI facilitated the analysis of individual incidents from the Accident database by calculating their spatiotemporal area-of-influence and helping to locate secondary incidents. The GUI was also connected with a Google Maps API to plot incidents on a satellite map, according to their latitude and longitude coordinates. The GUI was designed with a simplified framed structure and ease of navigation, making use of only three input values.

Frame 1 allows users to input date in MM/DD/YYYY format and highway chosen from the drop-down menu. The GUI then fetches the corresponding data from the Accident database for use in subsequent frames.

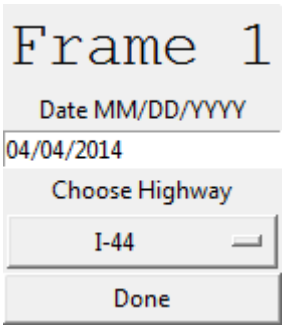

Figure 12. Frame 1 of the GUI

Frame 2 is navigational, 1) permitting users to advance to Graph 1 where incidents are plotted based on their differences in distance and time or 2) view the Doc_IDs of individual incidents for further analysis.

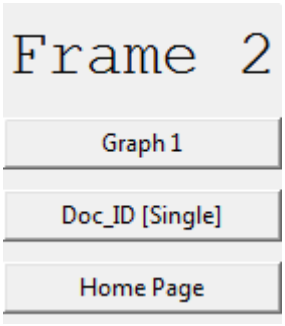

Figure 13. Frame 2 of GUI

Frame 3 shows the dynamic plot of all incidents on the specified date and highway, based on their differences in time and distance. Secondary incidents generally take place in close proximity to primary incidents; hence, visualizing clustered incidents is preferable in Graph 1. GUI plots are dynamic and can be zoomed in or out, saved, or shown in an altered graph size.

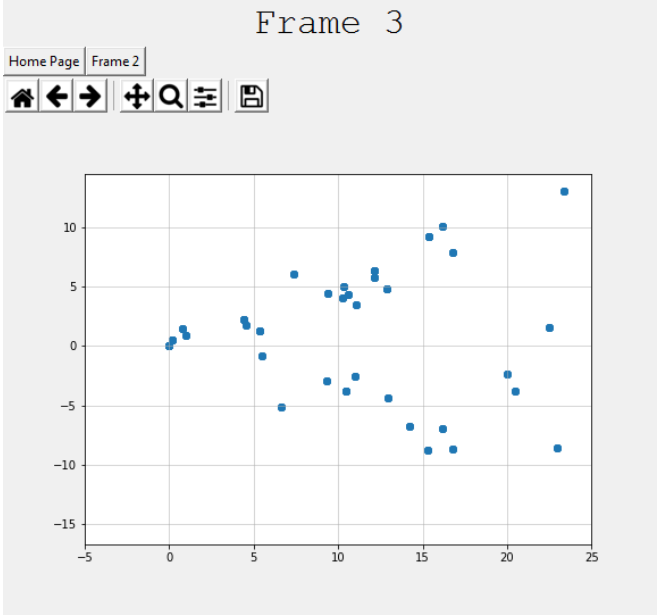

Figure 14. Frame 3 of GUI

Doc_IDs are unique and act as an identifier for the incidents. The Frame 4 of the GUI shows all the Doc_IDs corresponding to unique incidents on the chosen highway and their associated dates. Once the Doc_ID is chosen from the list, the button Graph 2 can be clicked to further analyze the incident and find out its spatiotemporal influence. Frame 4 also has a navigational option to return to the Frame 2 in case the inputs to the GUI need to be changed.

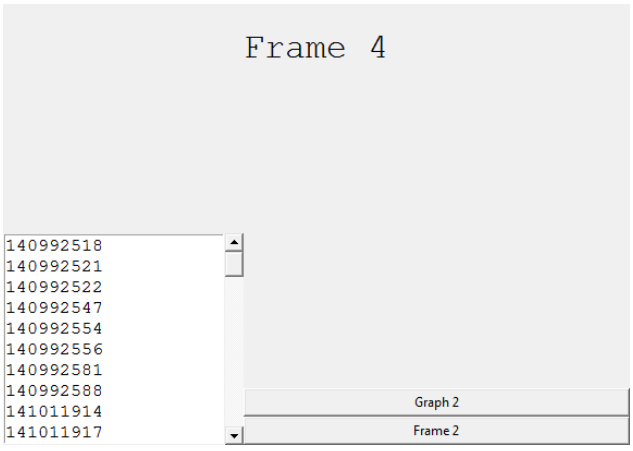

Figure 15. Frame 4 of GUI

Frame 5 illustrates the spatiotemporal analysis of a single incident. A single Doc_ID analyzed in the previous frame is processed in the backend algorithm utilizing different available features to calculate the incident's effect in terms of time (*t*) and distance (*d*) (which can also be interpreted as average wait time for a vehicle in the highway network and average queue length in response to an incident). The algorithm also calculates the difference in distance and time between user-selected incident and all other incidents recorded on that day. This information is then plotted, see Figure 3.10. The shaded region on the plot indicates the spatiotemporal boundaries within which incidents are considered secondary ones.

A Google Map API was added to the GUI to further validate secondary incidents by plotting them on a satellite map. The button Plot_gmaps plots incident points in the browser.

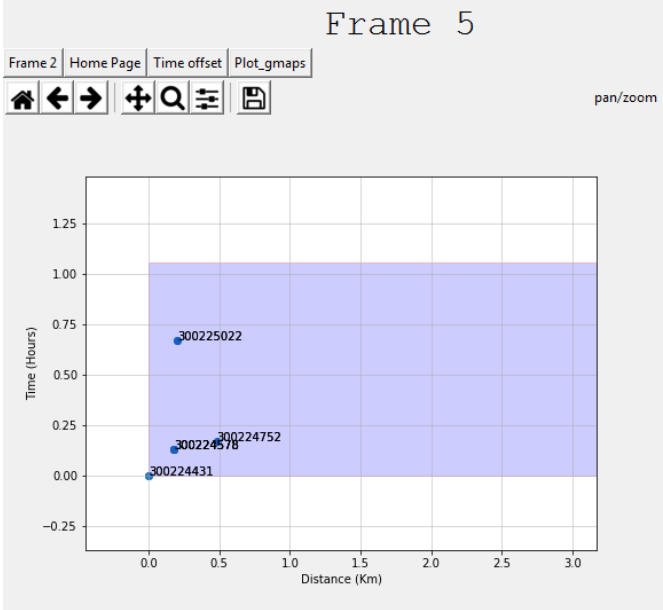

Figure 16. Frame 5 of GUI

The plot in the GUI represents all incidents in the spatiotemporal region of the primary incident. To ensure secondary incidents are located within the same highway

section, traffic flow direction is considered and then plotted into the Google Maps. The blue marker represents the primary incident, and the red marker represents all incidents within the primary incident spatiotemporal area of influence. A hover detail feature was added to display incident latitude, longitude, and Doc_ID.

The databases provided by ODOT had no indication of an incident being primary or secondary whatsoever. So, GUI was used to collect all the primary and secondary incidents used for statistical and artificial neural network modeling. The incidents were first analyzed using the GUI and then further validation was done using the google maps. Using this technique a small dataset was created with primary and secondary incidents classified.

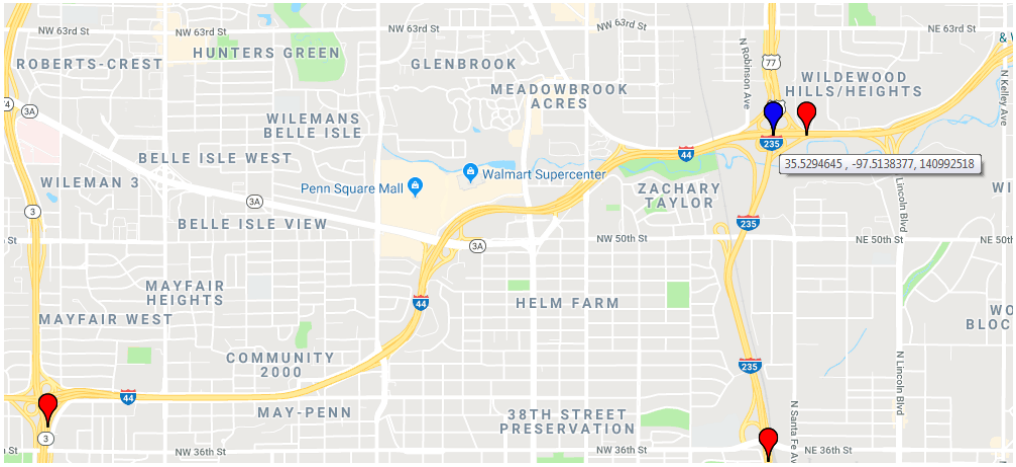

Figure 17. Incident plot in Google Maps

Chapter 4: Statistical and Artificial Neural Network Modelling

The previous Chapter introduced the concept of manually analyzing incidents using a GUI to determine which incidents are secondary. This information laid the foundation for understanding the effect of a primary incident and how additional incidents within its spatiotemporal area-of-influence could be considered secondary incidents.

Manually analyzing each incident in a database and classifying it as primary or secondary for the purpose of studying its parameters can be a tedious task. Hence, the need for automation is required. An algorithm was designed using statistical and neural network models to process incident databases and classify them as primary or secondary. This type of automated classification aids in assessing a large number of incidents and determining which factors can be used to identify incidents. This Chapter offers an overview of how ODOT data was processed for establishing an algorithm to accomplish this and informs about the variety of algorithms developed for modelling the incident classification system.

4.1 Dataset preparation

The Accident database provided by the ODOT Traffic Engineering Division was used to create statistical and neural network models. The dataset housed therein was characterized by an array of features related to the different aspects of an incident. After reviewing the literature, the following features were chosen as factors in the initial study.

Collision Time	Rush Hour Non-rush hour
Number of vehicles involved	The number of vehicles involved in the incident
Commercial vehicles involved	Indicates the presence of commercial vehicles involved in the incident
Number of lanes	Indicates the number of lanes on one side of the highway
Damage extent to the vehicle	01 - None 02 - Minor 03 - Functional 04 - Disabling 09 - Unknown
Average traffic	Indicates the average number of vehicles present on a highway in a day
Weather conditions	01 - Clear 02 - Fog/Smog/Smoke 03 - Cloudy 04 - Rain 05 - Snow 06 - Sleet/Hail (Freezing Rain/Drizzle) 07 - Severe Crosswind 08 - Blowing Snow 09 - Blowing Sand, Soil, Dirt 10 - Other
Light Conditions	01 - Daylight 02 - Dark / Unlighted 03 - Dark / Lighted 04 - Dawn 05 - Dusk 06 - Dark / Unknown Lighting

Table 2. Features used for modelling and their attributes

Only two major types of crashes — “rear-end collisions” and “side-swipe collisions”—were considered for the statistical and neural network models. After data processing, incident classification based on the factors listed in Table 2 became a binary. To address this, two statistical models, namely logit and probit, were selected to provide a starting point for classification and serve as a benchmark for the ANN modeling.

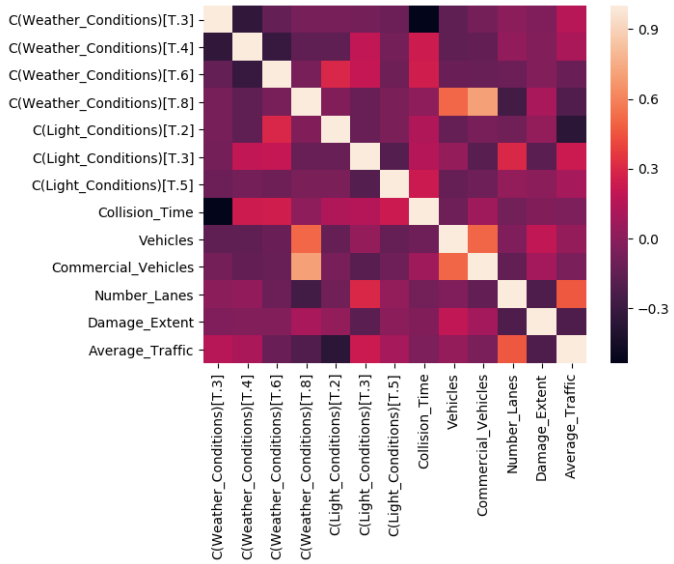

Figure 18. Correlation matrix for various features in the dataset

4.2 Statistical models for classification of incidents

To start of the automatic classification process, I chose to start with the logit model as it is simplistic and is widely accepted in many scientific areas. The rationale behind using two statistical models was to identify which model—logit or probit—worked better when processing transportation-related data.

4.2.1 Logit Model

The logistic regression (i.e., logit) model is one of the most widely known discrete choice models, often used in the fields of health sciences and ecological studies [24]. This particular statistical regression model is utilized in situations when the dependent variable is categorical. It performs well given the number of variables is limited.

The data detailed in this thesis is considered categorical/binary, representing primary or secondary incidents. The logit model considers one or more independent variables for determining an outcome. The model is an extension of the linear regression model, modified with the use of a link (or logistic) function. The logistic function bounds

the output between zero and one, which can then be interpreted as probabilities of independent variables assigned to a certain class.

Assume t is a function of an explanatory variable x that can be expressed as:

$$t = \beta_0 + \beta_1 x_1 + \beta_2 x_2 \ldots + \beta_n x_n$$

The logistic (or link) function $\sigma(t)$ is defined as:

$$\sigma(t) = \frac{e^t}{e^t + 1}$$

$$\sigma(t) = \frac{1}{e^{-t} + 1}$$

Logistic function can now be written as:

$$P(Y = 1|x) = F(x) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x_1 + \beta_2 x_2 \ldots + \beta_n x_n)}}$$

$F(x)$ can be interpreted as the probability of the dependent variable representing a case or success. β_0 is the intercept from the linear regression equation (i.e., the value of the criterion when the predictor is equal to zero), and β_1 is the regression coefficient multiplied by some value of the predictor. Base e denotes the exponential function.

4.2.2 The Probit Model

The probit model is also a regression model wherein dependent variable are assigned only binary values (e.g., zero or one, true or false) [25]. This is widely used in the fields of political sciences and econometrics. The probit model is based on the cumulative distribution function of the standard normal distribution and processes a set of independent variables for determining the probability of belonging to a certain class. It is based on the cumulative distribution function of the standard normal distribution. Much like the logit model, the probit model also uses a link function to bound the output between zero and one, called as the probit link function. The coefficients in the probit

model are calculated in a way similar to the logit model, using maximum likelihood estimation.

To better understand the probit model, assume t is a function of an explanatory variable x. Then, t can be expressed as:

$$t = \beta_0 + \beta_1 x_1 + \beta_2 x_2 \ldots + \beta_n x_n + \varepsilon = X\beta + \varepsilon$$

Now assume the model takes the following from:

$$\Pr(Y = 1|X) = \Phi(X\beta + \varepsilon) = K$$

where Φ is the Cumulative Distribution Function of the standard normal distribution and the value of $X\beta$ can determined from $'z'$ values.

The probit link function for the probit model can be defined as:

$$F(K) = \Phi^{-1}(K)$$

Additional parameters are estimated using maximum likelihood estimation. In the probit model, the value of $X\beta$ is the z-value of a normal distribution, where a higher value of $X\beta$ indicates the event is more likely to happen. The use of standard normal distribution causes no loss of generality when compared with the use of an arbitrary mean and standard deviation, primarily because adding a fixed amount to the mean can be compensated by subtracting the same amount from the intercept.

4.3 Artificial Neural Network

ANN is a machine learning classification (or prediction) algorithm heavily influenced by the structure of neurons in the brain [9]. Such algorithms have been successfully used in many fields of science. This section introduces ANN classification for incident-type detection.

ANN architecture is simple, yet powerful, performing well with limited dataset and input parameters. Data is divided into two sets, training and testing. The training data is fed into the neural network to train it for weights, which then can be used for classifications and in the next part determining the relative importance of the features. The testing dataset is used to test how well is the neural network performing. The architecture considered for the ANN modelling resembles Fig 4.1, where it has thirteen input nodes, seven hidden nodes and a single output node to provide probabilities of incident belonging to a certain class.

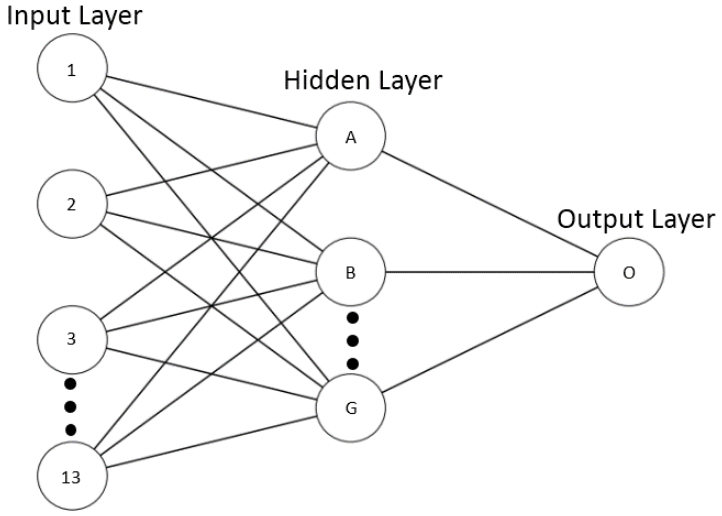

Figure 19. A simple Artificial Neural Network

Neural network classifiers are based on Multi-Layer Perceptron (MLPs) and similar to logistic models [21]. The neural network is divided into three layers—input, hidden, and output.

The output of the hidden layer can be calculated as follows, where w_{ik} is the connection weight between i^{th} input neuron and k^{th} hidden layer neuron:

$$net_k = \sum w_{ik} x_i + \theta_i$$

net_k is then passed through an activation function:

$$h_k = \frac{1}{1 + e^{-net_k}}$$

Similarly, weights are calculated for the output of the hidden layer to the output layer:

$$net_j = \sum w_{kj} h_k + \theta_j$$

where w_{kj} is the weight between neurons in the hidden layer and the output layer, and θ_j is the bias term. The output of the hidden layer is represented using h_k, and then normalized and bound between 0 and 1; the result is interpretable as probability and achieved using the following equation:

$$y_p = \frac{1}{1 + e^{-netj}}$$

In this research both models were considered collectively to increase the explanatory power of the features and understand their impact of features in classification.

Notably, ANN is trained using the training dataset before training weights are validated using the validation datasets. If accuracy or loss is not acceptable, changes can be made to the ANN structure by a) adding additional features, b) changing the number of nodes in the hidden layer or the number of epochs, or c) making similar adjustments etc.

4.4 Results - Statistical and Artificial Neural Network Models

The statistical and ANN models mentioned in the previous section were provided data from the Accident database prior to obtaining results. The models can then be used to automate primary and secondary incident classification.

This following section discusses the parameters used to gauge model and offers a comparative analysis to determine the optimal model given a particular dataset.

4.4.1 Results from the Logit Model

The logit model is a statistical model employing a logistic link function to bound the output between 0 and 1. The logistic model was created in Python leveraging the Statsmodels library. Data was divided into training and testing datasets with a split of 70-30, respectively. The model, then, was trained with the former dataset. Table 3 shows results derived from the logit model.

	coef	OddsRatio	P>|z|
Intercept	-1.3411	0.1913744	0.322
C(Weather_Conditions)[T.3L]	0.6222	2.111998	0.26
C(Weather_Conditions)[T.4L]	0.9132	3.286169	0.027
C(Weather_Conditions)[T.6L]	1.1729	2.811445	0.044
C(Weather_Conditions)[T.8L]	20.8792	4.35728E+12	1
C(Light_Conditions)[T.2L]	-0.5897	2.995745	0.564
C(Light_Conditions)[T.3L]	0.1713	1.259635	0.631
C(Light_Conditions)[T.5L]	1.9051	15.69422	0.014
Collision_Time	0.0129	1.017518	0.796
Vehicles	0.4181	1.73322	0.017
Commercial_Vehicles	0.7481	1.634679	0.092
Number_Lanes	-0.4668	0.5377444	0.039
Damage_Extent	0.0632	0.9575034	0.697
Average_Traffic	7.23E-06	1.000012	0.203

Table 3. Coefficients and odd ratio for Logit Model

The logit model uses maximum likelihood estimation to optimize parameters. Coefficients alone can be somewhat difficult to interpret and are only used for calculating probability using the mathematical form described in Section 4.2.1. The odds ratio can, however, be interpreted as the effect an additional unit of a feature would have on classification. For example, given that the odds ratio for Number_Lanes is .5 (which can be interpreted for each additional lane added to the highway), the chances of a secondary incident are .5 times as large. The p-values provided in the results table offer valuable insight into the data and demonstrate the effect on the overall model. A p-value of less

than 0.10 defies the null hypothesis, asserting that values in the dataset are important to model functionality. A p-value of approximately 1 signifies that the dataset values are not as important. If provided with mean of data, results would be the same. A confusion matrix was generated to analyze and understand the accuracy and precision of the model.

True Positives (TP)	False Positives (FP)
False Negative (FN)	True Negatives (TN)

Table 4. Confusion Matrix representation

Accuracy and precision from the confusion matrix can be calculated as:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

$$Precision = \frac{TP}{TP + FP}$$

where TP, TN, FP, FN represent the true positive, true negative, false positive, and false negative components of the matrix. The logit model performed with 67% accuracy and 48% precision. Although accuracy was acceptable, the model did not meet the acceptable precision objective.

A true positive rate vs. false positive rate plot was generated to determine the effectiveness of the prediction method. A curve leaning to upper left corner would represent an ideal scenario. Figure 20 shows that the logit model Receiver Operating Characteristic (ROC) curve stays above the dashed diagonal line, meaning that the model is predicting rather than guesstimating the class of an incident. However, performance was less than desirable, as many incidents were classified as false positives and false negatives. The area under the ROC curve can be used to compare the performance of different models.

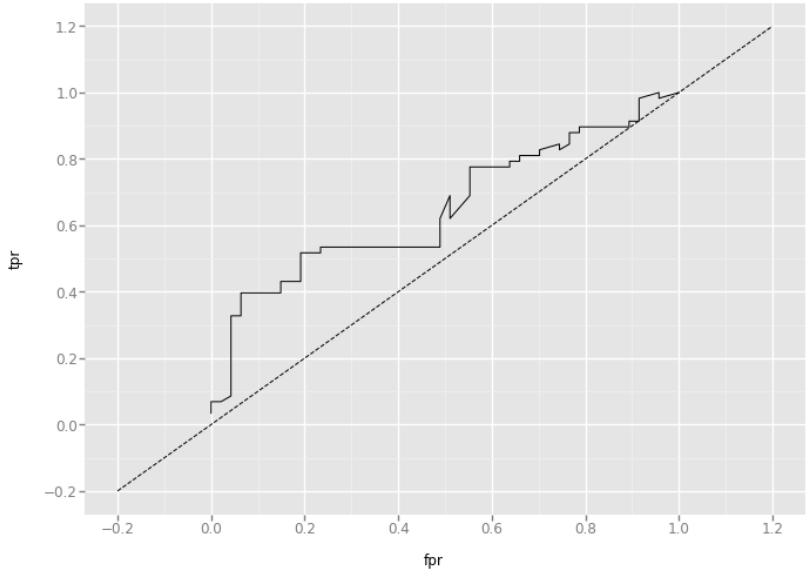

Figure 20. ROC for logit model

4.4.2 Results from the Probit Model

The statistical probit model can be used for binary classification and is typically utilized in the econometric field. The model is similar to the logit model, the only difference being the activation function. The probit model uses the cumulative distribution of the standard normal distribution to bound the output between 0 and 1. Like the logit model, the p-values in the probit model provide insight to the uniqueness of the data in the dataset. The odds ratios provide an insight to an increase of one unit in a feature affects the odds of classification as secondary incident and p-values less than 0.10 represent a feature that defies the null-hypothesis and contribute to the classification of the incident because of its uniqueness.

Table 5 enumerates results obtained after executing the probit model using the training dataset.

	coef	OddsRatio	P>|z|
Intercept	-0.7949	0.451621	0.332
C(Weather_Conditions)[T.3L]	0.3369	1.400595	0.31
C(Weather_Conditions)[T.4L]	0.4982	1.645807	0.037
C(Weather_Conditions)[T.6L]	0.6641	1.942813	0.057
C(Weather_Conditions)[T.8L]	7.5359	1.345443	1
C(Light_Conditions)[T.2L]	-0.3857	0.679976	0.529
C(Light_Conditions)[T.3L]	0.1041	1.109763	0.635
C(Light_Conditions)[T.5L]	0.9966	2.709062	0.013
Collision_Time	0.0102	1.010238	0.735
Vehicles	0.2492	1.282982	0.018
Commercial_Vehicles	0.4607	1.58515	0.094
Number_Lanes	-0.2821	0.754233	0.036
Damage_Extent	0.0327	1.033282	0.729
Average_Traffic	4.58E-06	1.000005	0.177

Table 5. Coefficients and odd ratio for Probit Model

The probit model performed with a 66% accuracy and 48% precision. The area under the ROC curve for the probit model was less than that of the logit model, demonstrating a decrease in accuracy. In fact, the probit model performed much like the logit model with the same precision. Suboptimal precision in both statistical models makes them difficult to use, as a classification model should have better accuracy and precision.

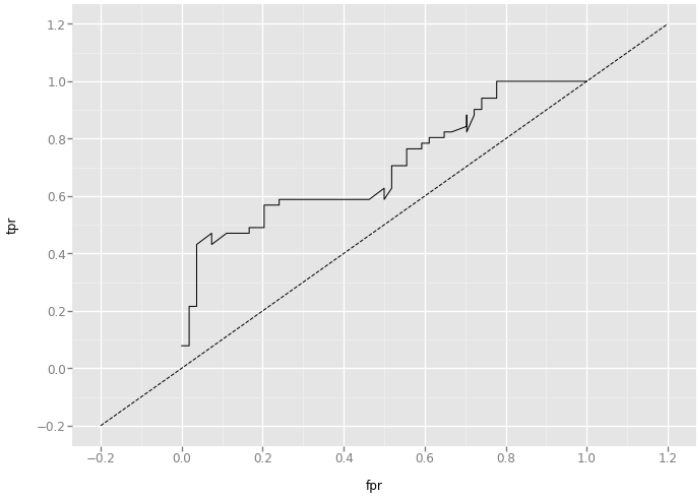

Figure 21. ROC for Probit model

4.4.3 Results from the Artificial Neural Network Model

ANNs have proven acceptable for both accuracy and precision in classification solutions and been used as classifier for many problems. In this work, the ANN model used the same data as the statistical model, with a 70-30 divide between training and testing datasets.

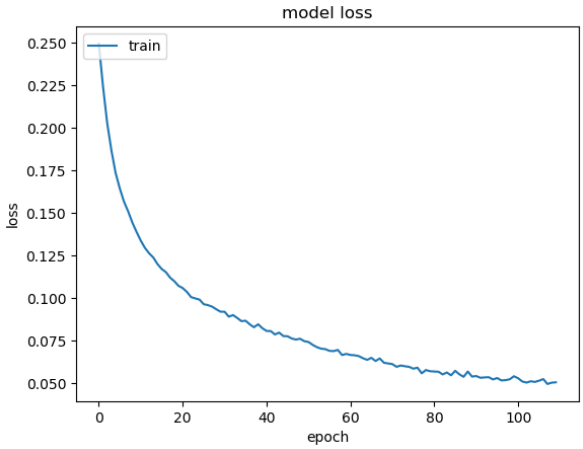

Figure 22. Loss vs epoch plot

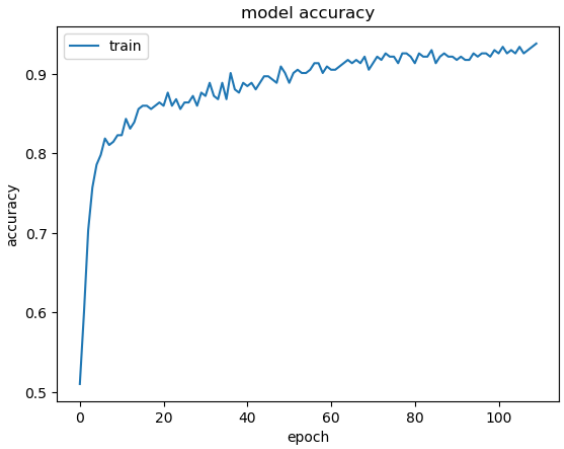

Figure 23. Accuracy vs epoch plot

Neural network training was performed for 110 epochs. Figure 22 illustrates the point at which training should cease. Similarly, accuracy vs. epoch plot commences forming a flat tail when raining approaches 110 epochs.

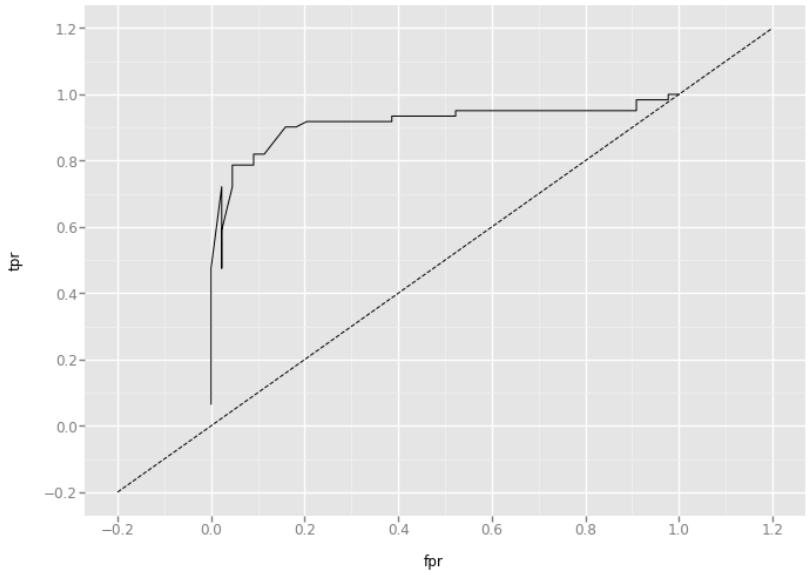

Figure 24. ROC for ANN model

The ROC plot takes on a good shape, indicating that the model is accurate. The area under the ROC curve in the ANN model was substantially superior to that of the statistical logit and probit models. ANN achieved 91% accuracy and 89% precision using the same testing dataset.

An overview of the accuracy and precision scores for the various tested models is shown in Table 6. ANN clearly outperforms statistical models for both accuracy and precision.

	Logit	**Probit**	**ANN**
AUC	0.669981188	0.665143779	0.91104182
Precision	0.48	0.48	0.890909091

Table 6. Comparative analysis of different models

Chapter 5: Factors Influencing Secondary Incidents

Having been utilized in various scientific fields, ANN has been proved a suitable model for classification problems. Despite the obvious benefit of providing impressive results for classification, ANN results can sometimes be hard to interpret and provide little or no information on the significance of independent variables. However, in transportation engineering, knowing independent variables that influence the model classifying certain incidents as primary or secondary can be extremely beneficial. For example, knowing that the presence of commercial vehicles on a highway is related to classifying a particular incident as secondary incident, the transportation department can take preventative steps for controlling and/or avoiding certain situations.

To address this challenge, a connection weight algorithm [15] was used to ~~gauge~~ determine variable importance in ANN. Such information can be used to better understand the influence of certain independent variables on classification and to make recommendations for improving data collection or identifying scenarios where secondary accidents are highly likely to happen.

5.1 Connection Weight Algorithm

The connection weight algorithm is the preferred method for accessing variable importance in simple feed-forward neural networks. Characteristics of such neural networks can be described as having an input layer and a hidden layer, as well as being fully connected and trained, using the back-propagation algorithm. Although characterized by simple architecture, these neural networks are good predictors. The connection weight algorithm considers neuron final weight between input layer and

hidden layer, as well as between hidden layer and output layer. To further understand the process, consider the following neural network.

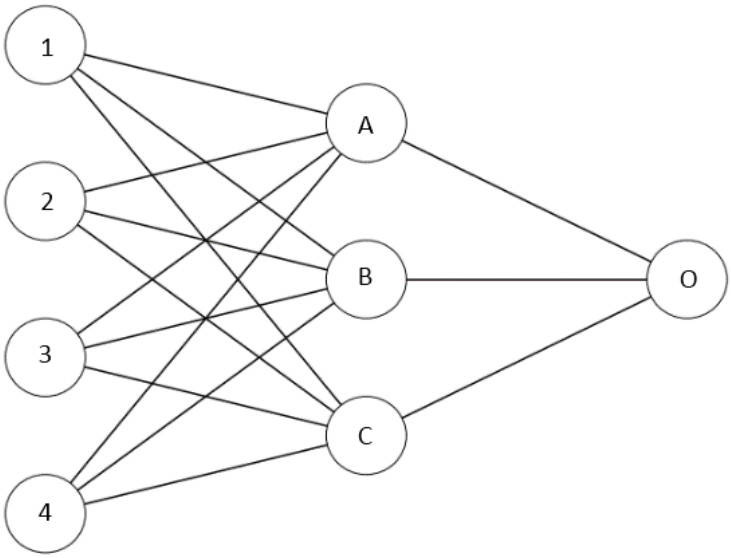

Figure 25. A simple three-layer neural network

Simple neural network architecture follows the rules of the connection weight algorithm. Weights from the input layer to the hidden layer are as follows.

Input-Hidden Connection Weights		Hidden A	Hidden B	Hidden C
	Input 1	-0.93	-1.49	0.37
	Input 2	-0.57	1.74	-0.14
	Input 3	-0.85	0.09	0.84
	Input 4	0.25	0.36	0.05

Table 7. Input to hidden connection weights

Weights from the hidden layer to the output layer are as follows.

Hidden -Output Connection Weights				
		Hidden A	Hidden B	Hidden C
	Output	-1.75	1.13	-1.01

Table 8. Hidden to output connection weights

The next step is calculating the product of the input- to hidden-layer connection weight with the hidden- to output-layer connection weight. The resulting products are as follows.

Connection Weights Products		Hidden A	Hidden B	Hidden C
	Input 1	1.63	-1.68	-0.37
	Input 2	1.00	1.97	0.14
	Input 3	1.49	0.10	-0.85
	Input 4	-0.44	0.41	-0.05

Table 9. Product of input-hidden connection weights to hidden output connection weights

After calculating the product, corresponding variable importance can be calculated by summing results across the hidden nodes.

	Importance	Rank
Input 1	-0.43	4
Input 2	3.11	1
Input 3	0.74	2
Input 4	-0.08	3

Table 10. Variable importance by Connection Weight approach

The connection weight approach offers a fair idea of how an independent variable influences the model for certain classification. Similarly, the effect of variable influence on secondary incident classification was calculated and can be seen in Figure 25. Weight from training the neural network is available in Appendix D.

Results from executing the connection weight algorithm demonstrated that features like collision time, number of vehicles, and presence of commercial vehicles heavily contributed to the model for accurately detecting secondary incidents. Similarly, weather related conditions (e.g., normal or light fog) did not affect the system as much as snow or icy conditions.

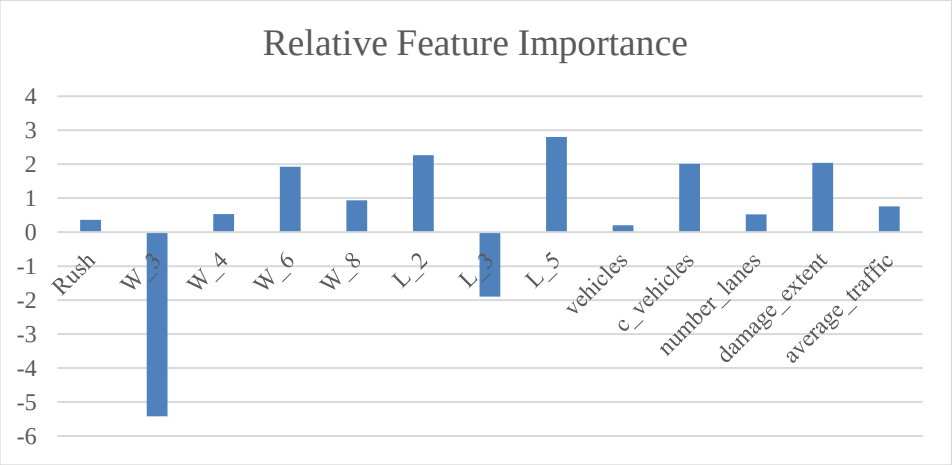

Figure 26. Independent variable influence for secondary incident detection

W_1	Clear
W_2	Fog/Smog
W_3	Cloudy
W_4	Rainy
W_5	Snow
W_6	Sleet/Hail
W_7	Severe Crosswind
W_8	Blowing Snow

Table 11. Legend for weather conditions

L_1	Daylight
L_2	Dark-Not Lighted
L_3	Dark-Lighted
L_4	Dawn
L_5	Dusk

Table 12. Legend for light conditions

Neural network complexity can be understood, and methods like connection weight algorithm can be used to gain explanatory insights into how independent variables influence the models' decision-making ability.

Chapter 6: Conclusion and Future Work

The work in this thesis presented an approach for identifying secondary incidents and understanding the spatiotemporal area-of-influence of primary incidents and various associated parameters. Results suggest that detecting secondary incidents can improve vehicle passenger safety and aid in effective utilization of transportation network resources.

Data from ODOT was utilized to define a dynamic boundary for primary incident spatiotemporal area-of-influence, and then leveraged using a memoryless queuing mechanism. Incorporating real-world highway dynamics (e.g., lane blockage, lane change, variable inter-arrival time, inconsistent traffic conditions at given times of day, variety of vehicles, weather, lightning) resulted in a more comprehensive and accurate algorithm. The combination of such parameters with a memoryless queueing mechanism proved to be an improved solution and provide an adaptive system for determining the spatiotemporal area-of-influence of primary incidents. A newly designed GUI encapsulated the process for determining the spatiotemporal area-of-influence and facilitating data collection and research.

Logit and probit, as well as ANN, models aided in understanding the relationship between primary and secondary incident features, as well as the likelihood of a secondary incident to occur. This research suggests the use of neural networks for incident classification based on a given number of features. Neural networks outperformed logit and probit models in terms of accuracy and precision.

A connection weight algorithm was used to define the relative feature importance of various independent variables and aided in further understanding of features affecting

secondary incident classification in further detail. The importance of this discovery will advance the base knowledge of possible root causes for secondary incidents and improve data collection for future studies.

Future work related to analyzing and identifying secondary incidents could include generating a more comprehensive dataset, including features like duration of incident and lane blockage during an incident. This type of information will aid in modeling scenarios for future studies and forecasting while also validating the accuracy of the current approach for detecting the threshold of primary incident spatiotemporal area-of-influence.

References

[1] Moore, J.E., Giuliano, G., and Cho, S. (2004). Secondary Accident Rates on Los Angeles Freeways. Journal of Transportation Engineering 130.3: 280–285.

[2] Raub, R.A. (1997). Secondary Crashes: An important component of Roadway Incident Management. TransportationQuarterly51.3: 93–104.

[3] Zhang H. and Khattak A. (2010). What Is the Role of Multiple Secondary Incidents in Traffic Operations, Journal of Transportation, Engineering, Vol. 136, No. 11, November 1, 986-997.

[4] Sun, C. and Chilukuri V. (2010). Dynamic Incident Progression Curve for Classifying Secondary Traffic Crashes, Journal of Transportation Engineering, Vol. 136, No. 12, 1153-1158.

[5] Kerner, B.S., Rehborn, H., Aleksic, M., Haug, A. "Recognition and Tracing of Spatial-Temporal Congested Traffic Patterns on Freeways", Trans. Rec. C, Vol. 12, 2004, pp. 369-400.

[6] Karlaftis, M. G., S. Latoski, P. Richards, J. Nadine, and K. C. Sinha. ITS Impacts on Safety and Traffic Management: An Investigation of Secondary Crash Causes. Journal of Intelligent Transportation Systems, Vol. 5, No.1, 1999, 39-52.

[7] Hirunyanitiwattana, W., and S. Mattingly. Identifying secondary crash characteristics for California highway system. CD-ROM. Transportation Research Board of the National Academies, Washington, D.C.,2006.

[8] Zhan, C., Gan, A., Hadi, M., (2009). Identifying Secondary Crashes and Their Contributing Factors, transportation Research Record: Journal of the Transportation Research Board, 2102, 68-75.

[9] McNelis P., and P. McAdam. Forecasting Inflation with Forecast Combinations: Using Neural Networks in Policy, Complexity Hints for Economic Policy New Part V, Economic Windows, 2007, pp. 253-270, Springer Milan.

[10] Tu, J. V. Advantages and Disadvantages of Using Artificial Neural Networks versus Logistic Regression for Predicting Medical Outcomes. *Journal of Clinical Epidemidemiology*, Vol. 49, No. 11, 1996, pp. 1225-1231.

[11] Zobel, C. W. and D. F.Cook. (2011). Evaluation of neural network variable influence measures for process control, Journal of Engineering Applications of Artificial Intelligence, 24(5), 803-812.

[12] Logistic Regression. https://en.wikipedia.org/wiki/Logistic_regression [online article].

[13] Probit Model. https://en.wikipedia.org/wiki/Probit_model [online article].

[14] Association Rule Learning. https://en.wikipedia.org/wiki/Association_rule_learning [online article].

[15] Olden, J.D., Jackson, D.A., 2002b. Illuminating the "black box": understanding variable contributions in artificial neural networks. Ecol. Model. 154, 135–150.

[16] Hallenbeck, M., Rice, M., Washington State Transportation Center (TRAC). Vehicle Volume Distributions Vehicle Volume Distributions by Classification by Classification. 16-17.

[17] Foteini P. Orfanou, Eleni I.Vlahogianni, Ph.D. and Matthew G. Karlaftis, Ph.D. Detecting Secondary Accidents In Freeways.

[18] Eleni I.Vlahogianni*, Ph.D., Matthew G.Karlaftis, Ph.D., Foteini P. Orfanou, 2002. Modeling the Effects of Weather and Traffic on the Risk of Secondary Incidents.

[19] Julian D. Olden *, Donald A. Jackson, 2002. Illuminating the ''black box'': a randomization approach for understanding variable contributions in artificial neural networks.

[20] Julian D. Olden, Michael K. Joy, Russell G. Death, 2003. An accurate Comparison of methods for quantifying variable importance in artificial neural networks using simulated data

[21] McNelis Paul D., Neural Networks in Finance: Gaining Predictive Edge in the Market. Elsevier Academic Press.

[22] CIW library, Python. http://ciw.readthedocs.io/en/latest/Reference/citation.html [online article].

[23] H. Rehborn and J. Palmer, 2008. ASDA/FOTO based on Kerner's Three-Phase Traffic Theory in North Rhine-Westphalia and its Integration into Vehicles

[24] David A. Freedman (2009). Statistical Models: Theory and Practice. Cambridge University Press, 121-122.

[25] Probit Regression. https://stats.idre.ucla.edu/stata/dae/probit-regression/ [online article]

Appendix A: Neural Network Connection Weights

Input hidden connection weights:

	A	B	C	D	E	F	G
Rush	0.69271	0.777883	0.684415	0.342325	-0.07679	0.811768	0.261276
W_3	-1.29719	-0.00645	0.128218	-0.99929	-1.10178	0.171364	0.040599
W_4	1.159608	0.213143	0.034387	-0.66016	-0.04609	0.442911	0.314572
W_6	0.520552	-1.20084	0.315501	0.340511	0.591987	-0.59904	-1.01597
W_8	0.053541	0.499754	0.912387	-0.17719	-0.02768	-0.42908	-0.65266
L_2	0.325479	0.597245	-0.39884	0.251416	0.73503	0.237695	0.566599
L_3	0.175793	-0.4718	1.291586	0.512648	-0.13926	0.110973	0.249066
L_5	0.665901	-2.58659	-1.04436	-0.37161	-0.09469	-2.1639	-1.3726
vehicles	-0.86972	0.315213	-0.20352	0.199212	0.834422	0.035555	0.216745
c_vehicles	1.093651	-0.59043	-0.62619	-1.19255	1.434648	0.238632	-0.05639
number_lanes	0.104821	-0.59005	0.094154	-0.22013	0.332132	-0.90319	0.32808
damage_extent	0.570255	0.138818	1.274808	0.921009	0.296363	-0.2668	-0.37151
average_traffic	0.597507	0.588416	-0.3907	-1.37449	-0.01554	-0.3746	0.690666

Hidden output connection weights:

	Hidden Nodes
A	1.809079
B	1.520681
C	-1.13517
D	1.137134
E	1.331394
F	-1.629735
G	-0.999267