Title

Traffic Conflict Identification and Validation using Deep Unsupervised Learning

Author

Lu, Jiajian

Publication Date

2022

Peer reviewed|Thesis/dissertation

Traffic Conflict Identification and Validation using Deep Unsupervised Learning

by

Jiajian Lu

A dissertation submitted in partial satisfaction of the

requirements for the degree of

Doctor of Philosophy

in

Civil and Environmental Engineering

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Professor Mark Hansen, Co-chair
Doctor Offer Grembek, Co-chair
Professor Alexander Skabardonis
Professor Alexandre Bayen

Spring 2022

Traffic Conflict Identification and Validation using Deep Unsupervised Learning

Abstract

Traffic Conflict Identification and Validation using Deep Unsupervised Learning

by

Jiajian Lu

Doctor of Philosophy in Civil and Environmental Engineering

University of California, Berkeley

Professor Mark Hansen, Co-chair

Doctor Offer Grembek, Co-chair

Surrogate safety measures can provide fast and pro-active safety analysis and give insights on the pre-crash process and crash failure mechanism by studying traffic conflicts. Traffic conflict can be identified by the presence of evasive actions or the amount of temporal (spatial) proximity measures like time-to-collision (TTC). However, it is not enough to use only one kind of measures in some scenarios and it is hard to set a threshold for those measures. Moreover, validating surrogate safety measures by connecting them to crashes is still an open question.

We first study the problem of traffic conflict identification. We proposed a method to identify traffic conflict by learning the representation of TTC and driver maneuver profiles with deep unsupervised learning and clustering the representations into traffic conflict and non-conflict clusters. We first trained a transformer encoder to encode sequences of surrogate safety measures into some latent space with unsupervised pre-training. Second, we identified informative clusters in the latent space by calculating the statistic summaries and visualizing trajectory pairs of each cluster. Some clusters are interpreted as traffic conflict clusters because they have small TTC, large deceleration rate and intertwining trajectories and they can be further interpreted as rear-end or angle conflicts. Moreover, the identified traffic conflicts contain critical conditions from the two vehicles in an interaction and one vehicle perceives them as abnormal and takes evasive action to avoid crashes.

Secondly, we study the problem of traffic conflict validation. We proposed a method to connect surrogate safety measures to crash probability using probabilistic time series

prediction. The method used sequences of speed, acceleration and time-to-collision to estimate the probability density functions of those variables with transformer masked autoregressive flow (transformer-MAF). The autoregressive structure mimicked the causal relationship between condition, action and crash outcome and the probability density functions are used to calculate the conditional action probability, crash probability and conditional crash probability. The predicted sequence is accurate and the estimated probability is reasonable under both traffic conflict context and normal interaction context and the conditional crash probability shows the effectiveness of evasive action to avoid crashes in a counterfactual experiment.

To my family. Thank you for your unconditional support.

Contents

List of Figures

List of Tables

Acknowledgments

The past five years at Berkeley has been a wonderful and amazing journey for me. I have so many great experiences, and the memories will last forever in my mind. Most importantly, I know it is the great individuals I met along the way that have made this possible.

First, I am so grateful to have Professor Mark Hansen and Doctor Offer Grembek as my research co-advisors. Their broad vision, open-mindness and cheering guidance have made my research work an enjoyable experience. When I feel confused, They are always there to guide me through the darkness and keep me on the right track. And during our bi-weekly meeting, I can always learn something new and get to know the most popular research topics. My PhD work would be impossible without the help and guidance from them.

I would also like to thank the other members from my dissertation committee. Professor Alexander Skabardonis has introduced me to the field of intelligent transportation system and using big data to solve transportation problems. The individual research experience with him in my first year has built the foundation for my research journey. I am also really grateful for Professor Alexandre Bayen's help as a committee member.

I am also extremely thankful to the UC Berkeley Safe Transportation Research and Education Center (SafeTREC) that has provided me lots of research opportunities and funding supports. Especially my colleagues Dr. Aditya Medury, Dr. Praveen Vayalamkuzhi, and Dr. Julia Griswold have given me lots of advice on study, life, GSR work and PhD research since my first day at UC Berkeley. Their passion on transportation research and ways of thinking have motivated and inspired me along the journey.

Meanwhile, my life at Cal is full of great memories with my peers and friends (Frank Feng, Weili He, Monik Pamecha, Ang Li, Lu Dai, Lin Yang, Wesley Hsiao, Laurence Tam, William Wang etc.). I will always remember the time we spent hanging out together, trying great food, making jokes of each other, and playing basketball. I would like to thank them and wish them all the best in the future.

Last but not least, I want to thank my family, for raising me up and providing me with endless love and support.

Chapter 1

Introduction

1.1 Background and Motivation

With approximately 1.3 million people dying each year worldwide as a result of road traffic crashes and about $20 - 50$ million more people suffering non-fatal injuries that incur a disability as a result of their injury [86], traffic safety is still a serious concern. It is a priority to understand the causes of road traffic crashes and lots of research works have been done in the field of road safety analysis to discover the relationship between crash frequency/severity and their determining factors, understand the crash occurrence mechanism and provide safety polices and countermeasures.

Traffic safety researchers have developed many approaches for measuring safety at a given location such as a road segment or intersection. These approaches mainly focus on using the occurrence of crashes and their consequences. For example, the safeness of a location can be ranked by the number of observed crashes over some years. The more crashes observed at a location, the more dangerous this location would be. The underlying assumption of these approaches is that analyzing traffic crashes is a valid method for safety estimation which is intuitive. However, there are several well-known issues of using crash data for traffic safety analysis, which include [77]:

1. Crashes are rare and random events. The rarity and randomness of crash occurrences often make it insufficient to collect crash data for weeks or months. The typical period to be considered sufficient is as long as three years [59, 73]. Moreover, drivers observed in crashes may not be able to represent the entire driving population since riskier drivers will be overrepresented in crash data, and this could lead to false interpretations of model parameters [52, 53].

2. Using crash data for safety analysis is a reactive approach and suffers from an ethical dilemma that fatalities and injuries need to accrue over a long time periods. We have to wait for crashes to take place before the corrective countermeasures can be applied. This problem is particularly severe for a new traffic infrastructure where there is no historical crash data available.

3. The crash data usually does not include the pre-crash process. The pre-crash driver behaviors and conditions receive little attention in police crash reports. It is difficult to understand the dynamics of a crash and how a pre-crash event evolves into a crash event. More details of the pre-crash events can help us better understand crash failure mechanisms and the evasive actions of drivers.

Given the shortcoming of current practices for crash-based safety analysis, traffic safety researchers are trying to find alternative measures of safety that are not based on the occurrence of crashes but the presence and occurrence of traffic conflicts. Analyzing traffic conflicts can provide fast and pro-active safety analysis as they happen much more frequently and give insights on the pre-crash process and crash failure mechanism by studying near misses [83, 82]. A traffic conflict is usually defined as "an observable situation in which two or more road users approach each other in space and time to such an extent that there is a risk of collision if their movements remain unchanged" [2]. In this definition, a traffic conflict is considered a broad concept that can be precisely identified using a range of surrogate safety measures (SSM) based on proximity (vehicles approaching each other in space and time) metrics such as time-to-collision (TTC) and post-encroachment-time (PET) and/or evasive actions (movement changes to avoid a crash), as well as near-misses, near-crashes and safety critical events [88, 87, 66]. In order to quantify this definition, researchers often classify a traffic event as traffic conflict if a surrogate safety measure like TTC exceeds some pre-defined threshold. However, there is still no consensus on which surrogate safety measures should be used [25] and the conflict threshold for these measures may be different under different traffic environments. A more detailed review can be found in Section 2.2.

The theoretical foundation of safety analysis using traffic conflict and surrogate safety measures assumes that all traffic events are related to safety. These traffic events have different degree of severity (unsafety) and a relationship exists between the severity and the frequency of events shown as Figure 1.1.

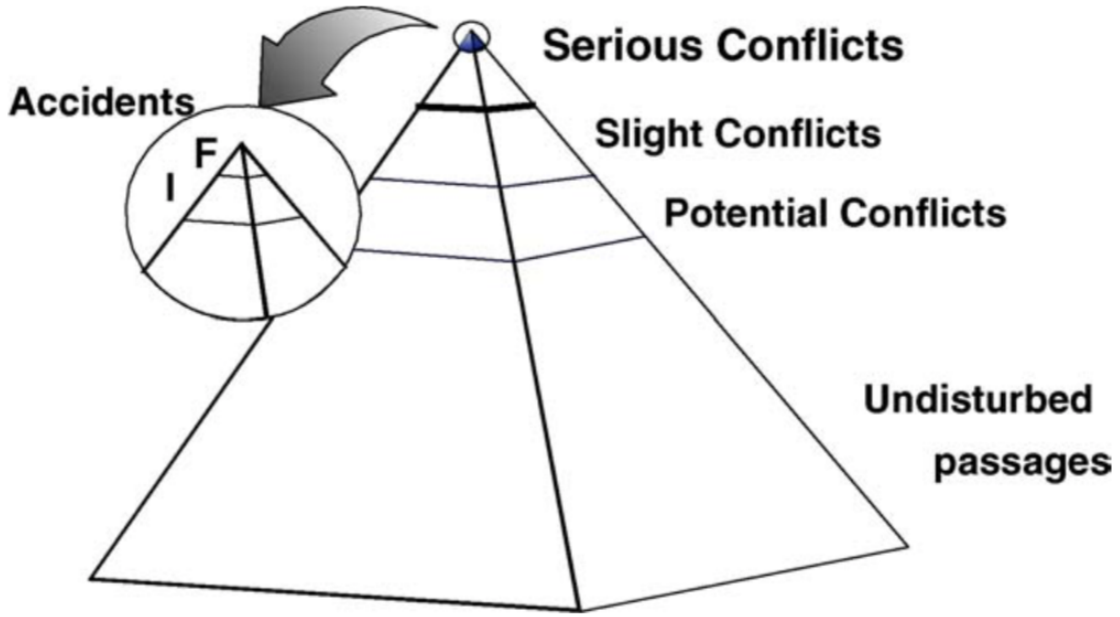


Figure 1.1: Safety Pyramid - the Interactions between Road Users as a Continuum of Events [36]

If the relationships between the layers of the safety pyramid are known, it is theoretically possible to calculate the frequency of the very severe but infrequent events (accidents) based on the known frequency of the less severe but more frequently occurring events [78]. However, connecting traffic conflicts to crashes is still an open question and several methods have been proposed [101]. Researchers proposed crash-based method like regression modelling [28] and non-crash-based method like causal model and extreme value theory (EVT) [13, 76]. However, these methods all have their limitations and a more detailed review can be found in Section 3.2.

With the rapid development of connected and autonomous vehicles (CAV) equipped with advanced sensing technologies, vast amounts of real-time vehicle data will become more available and these surrogate safety measures will play an important role in road safety analysis [61, 89, 85]. Moreover, deep learning, especially deep unsupervised learning, will become more applicable to surrogate safety analysis since lots of data can be fed to the model and these data are usually unlabeled. With these considerations, the aim of this dissertation is to explore the use of deep unsupervised learning on traffic conflict identification and validation problems.

1.2 Summary of Contributions

This dissertation focuses on solving two research questions for road safety analysis with traffic conflicts and surrogate safety measures: how to define and identify traffic conflicts, and how to validate the relation between traffic conflicts and crashes. For traffic conflict identification problem, our contributions can be summarized as the following:

1. We demonstrate the applicability of transformer encoder models with unsupervised pre-traning to real-world interaction data for traffic conflict identification.
2. The method unifies both proximity-based and evasive action-based surrogate safety measures and utilizes the entire time series of the interaction data.
3. The method eliminates the use of thresholds and interprets the similarities of traffic conflict clusters and non-conflict clusters.
4. We conduct eight case studies using real-world data that validate the usefulness of the proposed method and found general properties of the identified traffic conflicts.

For the problem of validating traffic conflicts, our contributions include the follow:

1. The method uses transformer-MAF to predict real time crash probability by estimating the probability density function of surrogate safety measures for every time step.
2. The method implements the dependency structure among condition, action and crash outcome from the causal model into the probability density functions using an autoregressive network.
3. The method overcomes the limitations of the causal model and uses all values of condition, action and crash outcome by treating them as continuous variables.
4. We estimates the model on real-world traffic data to compare the crash probability under traffic conflict and normal interaction scenarios and calculate the effectiveness of evasive action to avoid crashes.

1.3 Dissertation Outline

The dissertation is structured in the following manner:

1. Chapter 2 focuses on the first question of identifying traffic conflicts. We first discuss the motivation of this study and review the current methods for traffic conflict identification. Several limitations are pointed out and we proposed a deep unsupervised method by learning the representation of TTC and driver maneuver profiles. The concept of representation learning and pre-training strategy are introduced and the final structure of the model is proposed. The data information and several preprocess procedures are introduced. After training the model, the accuracy of the model is compared and the predicted sequences are presented. We then cluster the learned representations into several clusters and traffic conflict and non-conflict clusters are discovered through statistical comparisons and visualizing trajectory pairs of each cluster. Finally we discuss the general properties of traffic conflicts based on the identified traffic conflict clusters.

2. Chapter 3 studies the second question of traffic conflict validation. We first discuss the motivation of this study and review the current methods for traffic conflict validation. The core idea of the methods are summarized and several limitations are pointed out. We proposed deep unsupervised method to validate traffic conflicts by connecting surrogate safety measures to crash probability. The concept of the flow model for density estimation and the autoregressive structure for causal relationships are discussed. We further formulate the calculation of conditional action probability, crash probability and conditional crash probability using the probability density functions of condition, action and crash outcome. We then compare the accuracy of the trained model and the predicted samples. The aforementioned probabilities are computed under traffic conflict and normal interaction contexts and their validity is discussed. Finally, we conducted a counterfactual experiment to show the effectiveness of evasive action to avoid crashes.

3. Chapter 4 provides a comprehensive summary of our research motivation, objective, methodological frameworks, experimental results and corresponding findings. This chapter also identifies possible future research directions for traffic safety analysis using surrogate safety measures under the deep unsupervised learning framework.

Chapter 2

Traffic Conflict Identification by Representation Learning

2.1 Introduction

Traditional road safety analysis techniques which heavily rely on crash data have two major shortcomings. First of all, crashes are rare and thus safety researchers have to wait for years to collect enough crash data to draw conclusion for a given location. Moreover, it is reactive to improve traffic safety only after crashes happen [53]. Due to these limitations, traffic safety analysis using surrogate safety measures based on traffic conflicts have attracted more and more interest, since traffic conflicts are more frequent and it is proactive to analyze traffic conflicts instead of actual crashes [82]. The observation and analysis of the dynamics of traffic conflicts may provide insight into the failure mechanisms that lead to collision [37]. The analysis process usually involves observing interactions between road users, quantifying the severity of these interactions by surrogate safety measures, and identifying traffic conflicts if these interactions have safety implications [83].

There are two kinds of surrogate safety measures to quantify the severity of road user interactions. The first kind is based on the presence of evasive maneuver [64] as indicated, for example, by rapid deceleration and sharp turn. The second is based on the proximity in space and time of road users, as measured, for example, by time-to-collision (TTC) and post-encroachment time (PET) [3]. Given two interacting road users, a sequence of surrogate safety measures can be calculated and the interaction will be classified as a traffic conflict if the values of this sequence meet some pre-defined criteria. However, there is still no consensus on which surrogate safety measures should be used [25] and the conflict criteria for these measures may be different under

different traffic environment. Moreover, it has been repeatedly shown that the simple definitions such as TTC$<$ 1.5 seconds might capture the safety-relevant situation, but often also deliver up to 90% of completely irrelevant events [38]. Therefore, how to identify traffic conflict is still an open question [97].

With the help of computer vision techniques and in-vehicle devices to collect data, trajectories of road users can be extracted and driving maneuvers can be captured. As a result, the interactions between road users can be recorded and the entire time series of surrogate safety measures can be calculated. With the development of computing power and more traffic data being collected, data driven approaches such as deep neural networks (DNN) [46] have arisen as a prominent solution to many transportation problems as they are capable of mining information from messy and multi-dimensional traffic data sets with few modeling constraints. Convolutional neural network (CNN) is used to understand images of drivers to detect driving distraction [34] and Long Short-Term Memory (LSTM) is used to predict accidents [91]. However, most of the DNN applications in transportation are trained in a supervised manner, using data that includes labeled output. For traffic conflict identification problem, deep unsuperivsed learning is a more viable option, since there is no large dataset of interaction data with labels indicating whether an interaction is a traffic conflict.

In this paper, we propose a framework to identify traffic conflicts using time series of surrogate safety measures (SSM) and answer what is an ideal traffic conflict. First we train a transformer encoder to learn the latent representation of the sequences with unsupervised pre-training. Then the learned representations are clustered into several clusters with agglomerative clustering [57]. We then interpret some clusters as traffic conflicts by calculating their statistics and inspecting trajectory-pair plots and discuss the universal properties of the identified traffic conflicts. In summary, our main contributions are:

1. We demonstrate the applicability of transformer encoder models with unsupervised pre-training to real-world interaction data for traffic conflict identification.

2. The method unifies both proximity-based and evasive action-based surrogate safety measures and utilizes the entire time series of the interaction data.

3. The method eliminates the use of thresholds and interprets the similarities of clusters of traffic conflicts and non-conflicts.

4. We conduct eight case studies using real-world data that validate the usefulness of the proposed method and found the universal properties of traffic conflict from the identified results.

2.2 Literature Review

Traffic Conflict Identification

There are two definition of traffic conflict. One definition of traffic conflict is "an observable situation in which two or more road users approach each other in space and time to such an extent that there is a risk of collision if their movements remained unchanged" [3]. Based on this definition, proximity-based surrogate safety measures have been proposed to measure the severity of road user interactions such as time-to-collision (TTC) and post-encroachment time (PET). TTC is the time until a collision would have happened if two conflicting road users were to continue on their paths at current speeds [30]. An empirical approach to identify traffic conflict is to select a threshold and compare this pre-defined threshold with the calculated surrogate safety measures. For example, if the minimum TTC is less than 1.5 seconds, an interaction will be classified as traffic conflict. The theoretical assumption for this approach is that proximity is the surrogate for the severity of an interaction. For two vehicles on a collision course, the closer of the interaction, the more likely for it to become a crash. Many studies and research have been done to explore and validate the threshold to identify traffic conflict. [70] used a threshold of 1.5s for minimum TTC for signalized intersections, [74] used 2.0s for minimum TTC for un-signalized intersection and [25] used 1.5s for minimum TTC for urban roundabouts.

The second definition of traffic conflict is "a traffic event involving two or more road users, in which one user performs some atypical or unusual action, such as a change in direction or speed, that places another user in jeopardy of a collision unless an evasive maneuver is undertaken" [22]. Similarly, several evasive action-based surrogate safety measures like deceleration rate and jerk rate have been proposed and the empirical approach to identify traffic conflict is also to define a threshold and compare this pre-defined threshold with the surrogate safety measures. This approach is aligned with the evasive action-based definition since an evasive action undertaken to avoid crashes is usually a rapid maneuver like large deceleration and sharp turn. [4] used a threshold of $3.35m/s^2$ for deceleration rate at urban intersections to identify traffic conflicts, [25] used $3.35m/s^2$ for deceleration rate at urban roundabouts and [92] used $-8m/s^3$ for jerk rate at intersections.

More and more research has shown the importance of evasive action in identifying traffic conflicts. [80] showed that surrogate safety measures designed to detect evasive actions, such as deceleration, jerk and yaw rate are better able to identify traffic conflicts in less-organized traffic environments with highly-mixed road users. [79] compared the effectiveness of time-proximity and evasive action conflict measures in five cities and found that evasive action-based indicators are more effective in traffic

environments such as Shanghai and New Delhi where close interactions between road users are common and sudden evasive actions are frequently used to avoid crashes. [92] used jerk rate as a complementary indicator and the traffic conflict detection accuracy was raised from 84.2% when TTC was the only measure to 91% when the jerk measure was added.

A problem for the current approaches to identify traffic conflict using surrogate safety measures is that researchers only use one critical value of a sequence of the surrogate safety measure and compare it with a predefined threshold to identify traffic conflicts. Each published study has its more or less unique thresholds or indicators to identify traffic conflict which makes the results hard to compare. Therefore, some researchers have dropped the threshold-based approach and started to analyze the shape of the time series profile of surrogate safety measures. [58] categorized the interactions between vehicles and pedestrians into three groups, hard interaction, no interaction and soft interaction, according to the curve shapes of TTC and gap time (GT) and used a different combination of TTC and PET and thresholds for each group to identify traffic conflicts. [40] categorized pedestrian-vehicle interactions into two types of patterns based on the similarity in shape of TTC, GT and Speed profile. Pattern 1 is when either pedestrian or vehicle or both take an evasive action while in pattern 2 neither of them does so. Different indicators and thresholds were adopted to identify traffic conflicts in the different patterns.

Representation Learning

The ability to learn useful representations of data with little or no supervision is a key challenge in applying artificial intelligence to the vast amounts of unlabelled data collected in the world. Representation learning is a set of techniques that tackles this problem. The goals of these techniques is to learn reusable feature representations from large unlabeled datasets in order to make it easier to extract useful information when building classifiers or other predictors [7, 67]. The learned features are useful in many downstream tasks and they are superior to feature engineering based on human judgment [47].

Learning Strategies

Many strategies such as AutoEncoder, contrastive learning and unsupervised pre-training, for learning the representations have been proposed. AutoEncoder is a type of neural architecture that contains an encoder and a decoder [43]. The encoder will encode the input data into some latent space and the decoder will decode the latent representation and try to reconstruct the original input data as close as possible. [9]

trained a variational autoencoder with multidimensional road traffic datasets for road traffic forcasting. The model can impute the unobserved traffic data with missing values and using the learnt latent representation as input achieved better forecasting accuracy than using the original data. Contrastive learning like SimCLR [11] learns representations by maximizing agreement between representations from similar data while minimizing the agreement between representations from dissimilar data. BERT [14] used unsupervised pre-training to learn the representation of sentences. It simply masked some percentage of the input tokens of a sentence and then used the representations of the sentence to predict those masked tokens.

Neural models for time series data

Many neural models are suitable for handling time series data like audio, video and text. Recurrent neural network (RNN) can learn a hidden state of one time step and take this hidden state as additional information for generating the hidden state of next time step [56]. Therefore, the temporal relationship between each time step can be extracted. Long short term memory (LSTM) is a variant of RNN [33]. The core idea behind this network is that there is a cell state that bypasses each time step and therefore the network is able to get information from farther in the past. Several researchers [48, 96] have shown that LSTM can capture both long term and short term information from time series data and achieve better traffic forecasting accuracy than RNN.

However, the recurrent model can only process the next time step after the previous time step is processed and inhibiting parallelization. Dilated CNN [60] was used in WaveNet to learn the representation of audio data. The model mainly focuses on capturing the local dependency between time steps and it can be parallelized since the model sees the entire sequence of the data instead of one time step at a time. Transformer [84] uses only attention mechanism, dispensing with recurrence and convolutions entirely. The experiment showed that the transformer model produces better results, becomes more parallelizable, and requires significantly less time to train. However, most of the applications of transformer framework are built to solve problems in natural language processing (NLP) and recently [93] applied the transformer and unsupervised pre-training framework to learn the representation of six multivariate time series datasets and tested the performance of the learnt latent representation on regression and classification tasks. They showed that the method can learn a good representation of the time series data and the performance even exceeded the supervised methods.

2.3 Methodology

In this section, we propose a framework that implements a transformer encoder to learn the representation for road vehicle interaction data in an unsupervised manner, clusters the representations and interprets the clusters from three aspects. Given the trajectory data of road users, we first calculate sequences of TTC, longitudinal acceleration and lateral acceleration of each interaction between two road users. Then the time series data are used to train the transformer encoder model with an unsupervised pre-training strategy. After training the model, we feed the time series data to the model again to get the representation of the data. Next agglomerative clustering is used to cluster the latent representation. Finally, we interpret the clusters by calculating the statistical summaries and visualizing the trajectory pairs. A diagram of our methodology is shown in Figure 2.1.

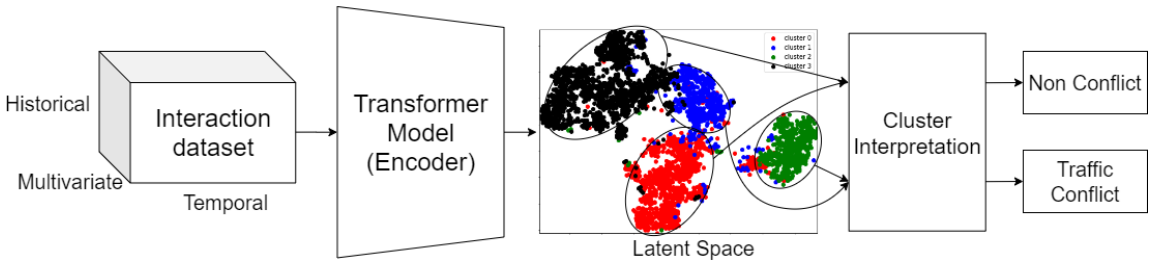
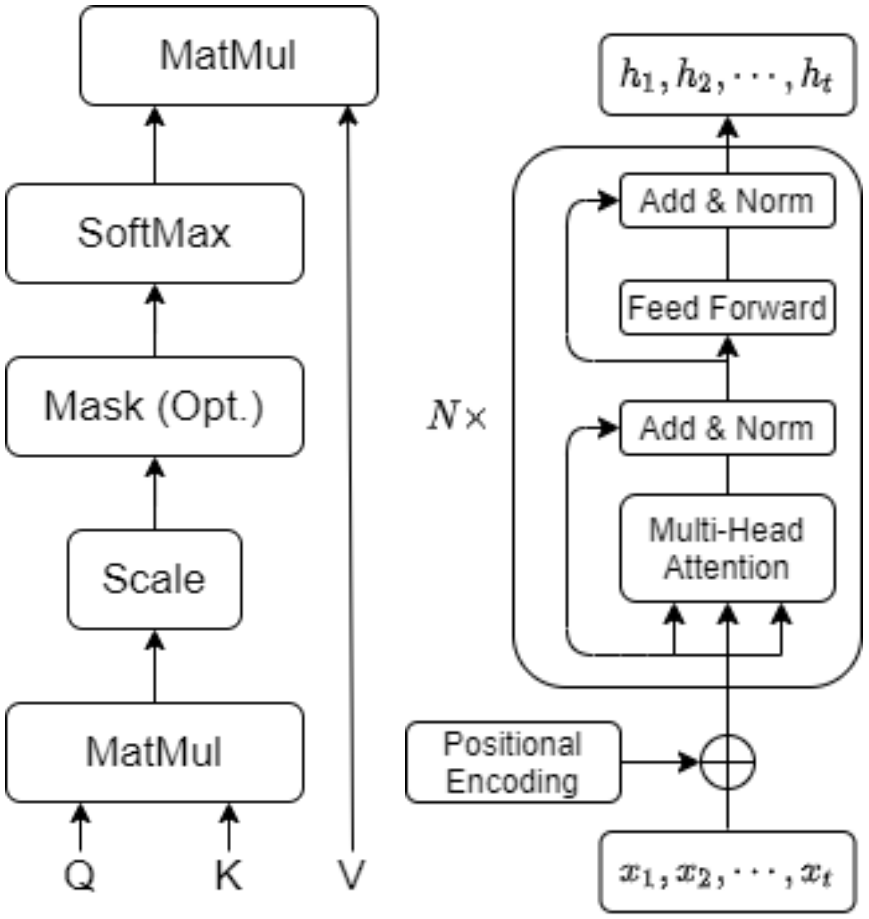


Figure 2.1: Diagram of the methodology

Transformer Encoder

Our model only consists of a transformer encoder which is different from the transformer model with both encoder and decoder in [84]. The reason is that our work purely focuses on learning a unified latent representation of the time series data that is suitable for any downstream task while the original paper focuses on sequence-to-sequence tasks like machine translation. Without the decoder part, our model contains only half of the layers as the original transformer model, and therefore our model is lighter and faster to train. The core layer for the transformer encoder is the attention layer. The idea of attention is that when the attention layer is looking at some time step named as query (Q) of a sequence, it will calculate how much attention it should pay to every other time step named as keys (K) in the sequence. Then the attention is normalized and becomes a weight matrix. Finally the weight matrix is multiplied by the current time step named as value (V) which is the same as query. The implementation of the attention layer is shown in Figure 2.2a. Moreover, we

implement the multi-head attention layer, the parallel version of the attention layer, to improve model performance. Figure 2.2b shows the architecture of the transformer encoder. First we need a positional encoding layer to encode the position information of the time series and add the encoded information to the time series data since the attention layer sees the entire time series at the same time and we want to make use of the order of the time series data. We use a learnable matrix instead of a fixed sine and cosine function in [84] for the positional encoding layer. We also use the residual connection [31] and layer normalization [6] techniques, which have been found to greatly improve the performance of deep neural networks.



(a) Scaled Dot-Product Attention

(b) Transformer Encoder

Figure 2.2: Architecture of Transformer Encoder
Latent representation at time t is $h_t \in \mathbb{R}^d$ and input sequence at time t is $x_t \in \mathbb{R}^f$, d is the dimension of the latent space and f is the dimension of the input features

Unsupervised Pre-training

We train our model using the unsupervised pre-training strategy. Specifically, we set part of the input to some masking value. Any value that is different from the time series data can be used and here we use -1.5 since the original data will be normalized to $[-1, 1]$. Then we ask the model to predict the masked value as shown in Figure 2.3. The masked input and a masking matrix M are generated with Algorithm 1.

Algorithm 1 Generating Masked Input $\tilde{\mathbf{X}}$

1: Set mask length to be $r\times$ seq length where r is the masking ratio
2: Split mask length into multiple small segments
3: Initialize $M \in \mathbb{R}^{T\times f}$ to be all zeros, where T is the length of the sequence and f is the number of features in $\mathbf{X}$
4: **for** each feature of $\mathbf{X}$ **do**
5: **for** each segment **do**
6: Randomly pick a starting point s
7: Set the value of the feature from position s to position $s+$ len(segment) to be -1.5
8: Set the same locations in M to be ones
9: **end for**
10: **end for**

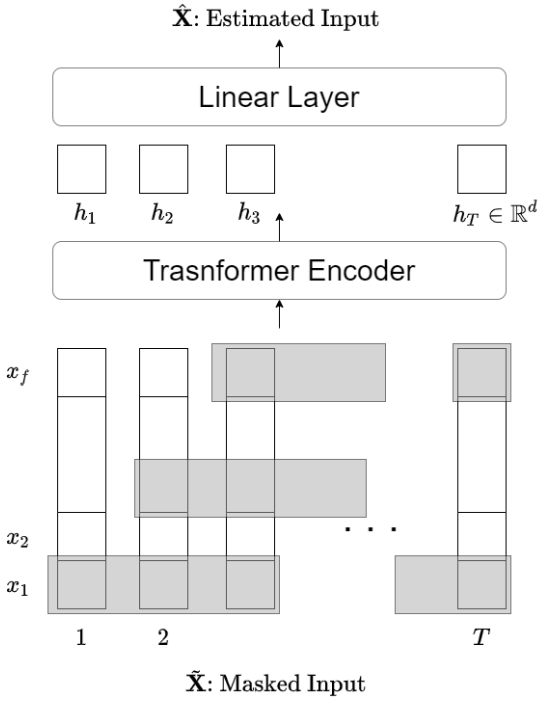


Figure 2.3: Training Strategy: Unsupervised Pre-training

A proportion $r = 0.15$ of the sequence is masked and multiple small segments

with l time steps (l randomly picks values from $[2-4]$) are created to mask the input sequence. We do not choose a larger r or mask a long consecutive segment of the input sequence since they will mask out too much information and make the prediction task unnecessarily challenging. Moreover, we also eliminate masking a very short segment (length of 1) since it is trivial to predict with good approximation by replicating the immediately preceding or succeeding values or the average of them [93]. Finally all the features from one data input are masked instead of one feature at a time is masked. It encourages the model to learn to attend not only to the previous and later pieces in each feature, but also to the values of other features at the same time step in the sequence, and therefore the model can learn the relationship between two variables at the same time step.

Second, we add a linear layer which takes latent representations $\mathrm{h_t}$ from the transformer encoder as input and output the estimated $\hat{\mathrm{x}}_\mathrm{t}$. And finally, we calculate the Mean Squared Error loss between $\hat{\mathrm{x}}_\mathrm{t}$ and $\mathrm{x_t}$ only over the masked location as in Equation 2.1:

$$\mathcal{L}_{\mathrm{MSE}} = \frac{1}{|M|}\|(\hat{x}_t - x_t) \odot M\|_2^2 \tag{2.1}$$

Clustering and Interpretation

After training the model and obtaining the representations of the interaction datasets, we utilize the learnt representation to solve our downstream task: traffic conflict identification. However, we do not have the ground-truth labels of the interaction datasets, we can only use clustering technique and interpret the clusters in a way that different clusters have different characteristics that align with domain knowledge of traffic safety. Hence, agglomerative clustering with Ward linkage [57] is used to cluster the latent representations.

The next step is to interpret the clusters and determine if some can be identified as conflicts. To find traffic conflict clusters and we calculate the average minimum TTC, average minimum speed and average minimum longitudinal acceleration for each cluster as the cluster properties since these values are surrogate safety measures often used in the literature to identify traffic conflicts. Moreover, visualizing trajectory pairs of interacting vehicles involved in an interaction can illuminate the intention of the drivers and why they do or do not perform evasive actions. Therefore, we interpret traffic conflict clusters by calculating the statistical summary and plotting the trajectory pairs.

2.4 Data

The dataset we use for the study is INTERACTION dataset [94] which is extracted from videos of three unsignalized intersections and five roundabouts taken by drones and traffic cameras. The dataset records the position (x, y), velocity (v_x, v_y) and yaw angle θ for each vehicle within videos at every time step of 0.1 second. We prepare the data as shown in Diagram 2.4

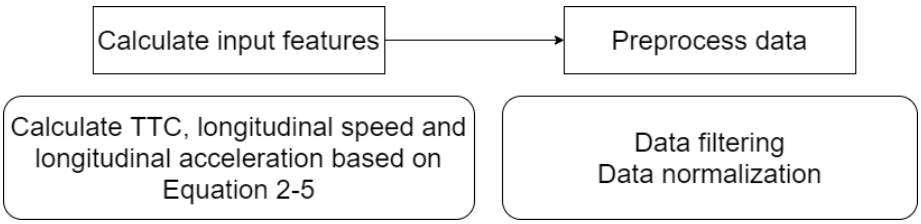


Figure 2.4: Data Processing Diagram

$$\mathrm{TTC}^t = \begin{cases} k, & \text{if } \sqrt{(x_i^t + v_{x_i}^t k - (x_j^t + v_{x_j}^t k))^2 + (y_i^t + v_{y_i}^t k - (y_j^t + v_{y_j}^t k))^2} \leq d \\ \infty, & \text{otherwise} \end{cases} \tag{2.2}$$

where subscripts i and j represent the first and the second vehicle, supperscript t represent the time step t and d is the minimum distance between the centers of two vehicles which is set to 2 meters. The resulting TTC is a sequence of $\{\mathrm{TTC}^t\}$ where each TTC^t is calculated by the speed and position at time step t. d only affects the value of TTC but not the presence of TTC which is determined by whether two vehicles are on a collision course. When two vehicles are on a collision course, a larger d will create a smaller TTC and vice versa. Since the average length and width of a vehicle are around 4 meters and 2 meters respectively, if we use $d = 4$ meters, it will introduce some non-crash scenarios like two vehicles driving parallel to each other. Using $d = 2$ meters eliminates those scenarios while increases the TTC value by a small amount for other types of conflicts. To give an idea of how much the TTC value is increased, we can consider an rear-end conflict situation where one vehicle is stopped and the other vehicle is approaching from behind with speed of $4m/s$ at some point. The conflict point for $d = 4m$ is $2m$ closer than that for $d = 2m$ so the TTC for $d = 4m$ is $0.5s$ shorter than that for $d = 2m$.

$$v_{\text{long}} = v_x \cos\theta + v_y \sin\theta \tag{2.3}$$

$$a_{\text{long}} = a_x \cos\theta + a_y \sin\theta \tag{2.4}$$

where a_x and a_y is the first derivatives of v_x and v_y. We use the average acceleration of two consecutive time steps to approximate the derivative:

$$a = \frac{v_1 - v_2}{t_1 - t_2} \text{ where } t_1 - t_2 = 0.1s \tag{2.5}$$

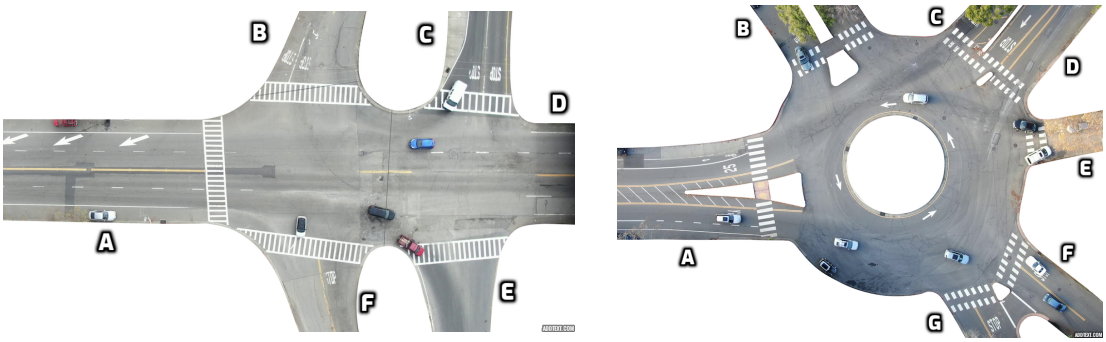
We also considered other commonly used evasive action-based surrogate safety measures, lateral acceleration [26] and jerk rate. However, we do not include them as features since in the context of intersections and roundabouts drivers perform large lateral acceleration mostly for turning movement, while the magnitude of jerk rate after normalization is too small compared with other features to convey information to the model.

After calculating all the input features, we filter out interactions with sequence length less than five time steps or minimum TTC greater than four seconds. Interactions with short length are most likely outliers and interactions with minimum TTC greater than four seconds entail very little collision risk. Then each feature i of the data is normalized to [-1, 1] through Equation 2.6 in order to make the training of the model stable.

$$x_i' = 2 * \frac{x_i - \min(x)}{\max(x) - \min(x)} - 1 \tag{2.6}$$

We represent an interaction between two vehicles as five sequences: TTC, v_{long} for vehicle i and j and a_{long} for vehicle i and j. After processing all the interactions from eight scenarios, we have 12037 interactions in total and the data for training has shape $[12037, T, 5]$ where T is the sequence length for each interaction. T is determined by the time window that two objects have non infinite TTC and T varies among different interactions since interactions when the rear vehicle is approaching the front vehicle would usually last longer than interactions when two vehicles cross paths.

We select the intersection labeled DR_USA_Intersection_GL (GL) and the roundabout labeled DR_USA_Roundabout_FT (FT) as representatives of four intersections and four roundabouts that are analyzed in our study. Bird's-eye views of GL and FT are shown in Figure 2.5 and the names for the legs are also defined. Legs B, C and F are stop sign control while legs A and D have the right of way for GL. Leg A has yield sign control while legs B-G have stop sign control in the case of FT.



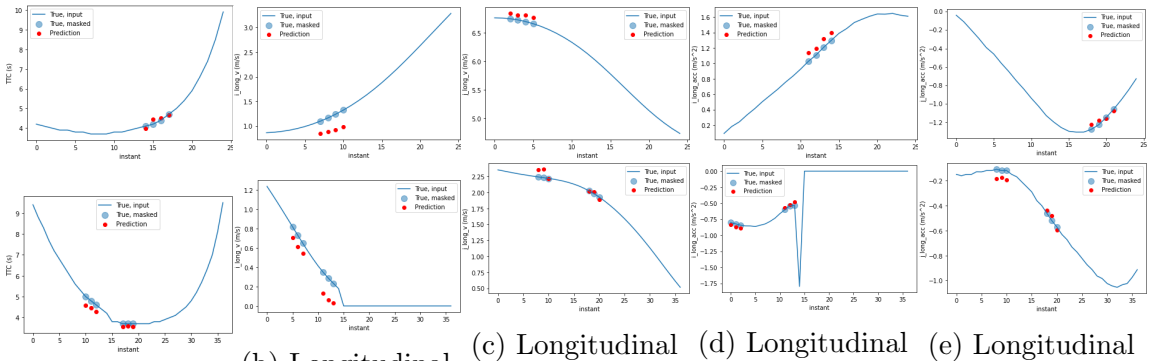
(a) GL (b) FT

Figure 2.5: Bird's-eye views of two locations

2.5 Experiments and Results

Training Transformer Encoder

We split the data into training, validation and testing set by 80/10/10 and train the model for 100 epochs with Adam optimizer and warm up scheduling. The hyperparameters for the transformer encoder model are shown in Table A.1 of the Appendix. The final testing mean square error is 9.2e-3 and some visualizations of the estimated input data are shown in Figure 2.6. The visualizations shows that the learned representation contains most of the information of the sequences since even with one simple fully connected layer, the masked segments can be reconstructed from the learned representation reasonably well. There is some deviation between the prediction and the ground truth but the goal of the model is to learn the representation of the sequences rather than predict the sequences. This deviation error will not go into the clustering part and affect the clustering results since we will use the learned representation instead of the reconstructed sequences for clustering.



(a) TTC (b) Longitudinal Speed for vehicle i (c) Longitudinal Speed for vehicle j (d) Longitudinal Acceleration for vehicle i (e) Longitudinal Acceleration for vehicle j

Figure 2.6: Examples of estimated input sequences

Clustering Latent Representations

The trained transformer encoder model is used to encode the scaled sequential data and outputs the latent representation $h_t \in \mathbb{R}^{128}$. Then we calculate the average and maximum for each latent dimension over the entire time series and have $(h_{\text{mean}}, h_{\text{max}})$ as the final representation of an interaction. The reason is that mean pooling captures general information and does not focus too much on specific values while max pooling captures the most dominant property of the time series [95]. Interactions with varying time step T are now transformed into latent representations with the same shape $(2, 128)$. Next agglomerative clustering is applied to cluster the latent representations of interactions. The clustering is performed individually for each location since interactions are liked to differ systematically for different locations due to geometry or other factors.

Intersection GL

The clustering results for GL are visualized through t-Distributed Stochastic Neighbor Embedding (t-SNE) [50] which is often used to visualize high-dimensional data. We find that using 5 clusters fits the data best as shown in Figure 2.7. We interpret the clusters using trajectory pair plots and summary statistics. On this basis we conclude that the interactions in cluster 1 are angle conflicts while those in cluster 4 are rear-end conflicts. Interactions in other clusters are not traffic conflicts. The trajectory pair plots and their main patterns are shown in Figure 2.8- 2.12.

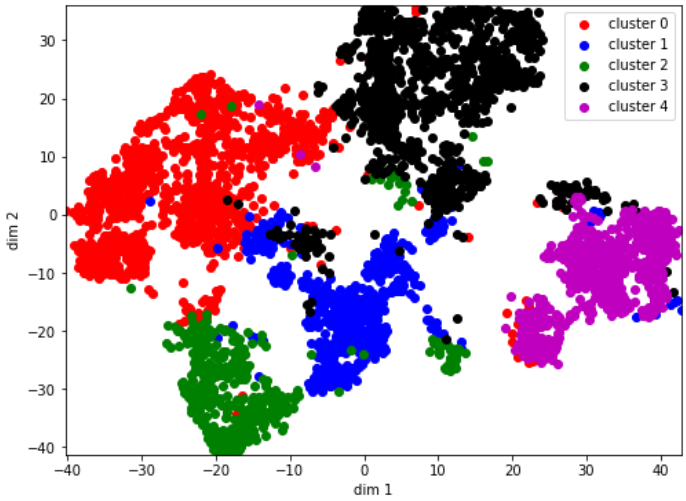
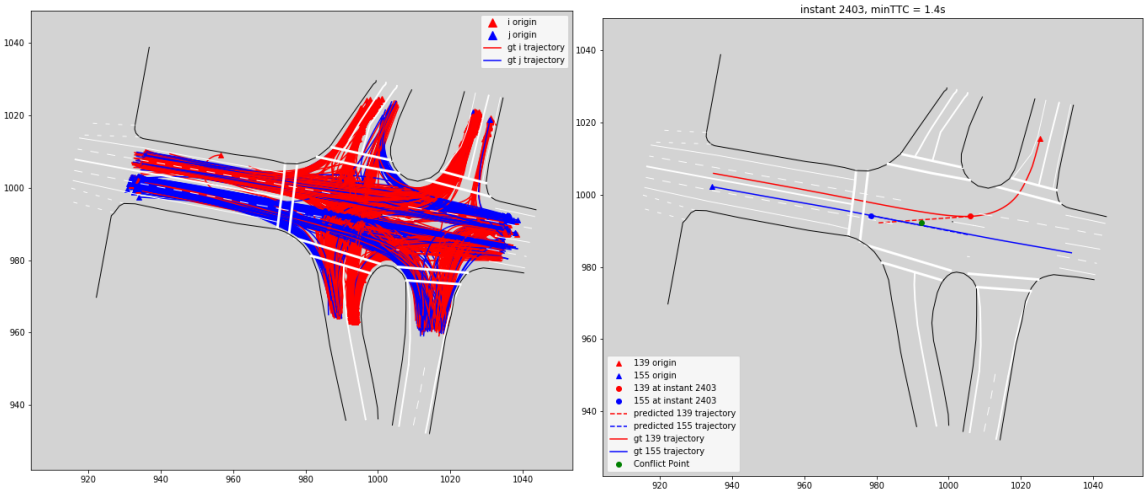
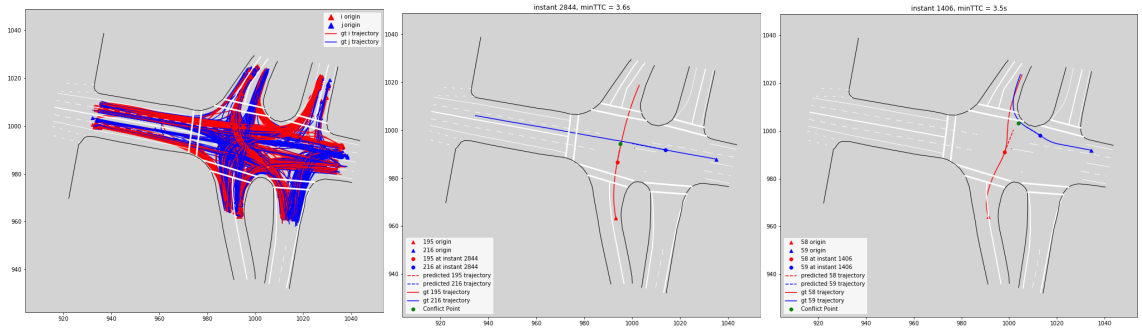


Figure 2.7: Clusters of Interactions at GL



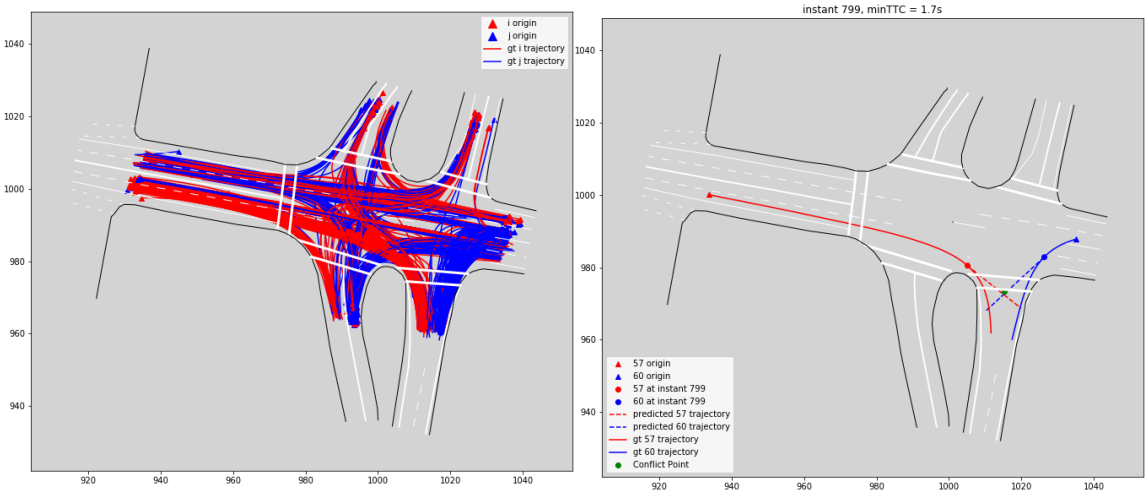
(a) overall (b) pattern 1

Figure 2.8: Trajectory Pairs for GL Cluster 0



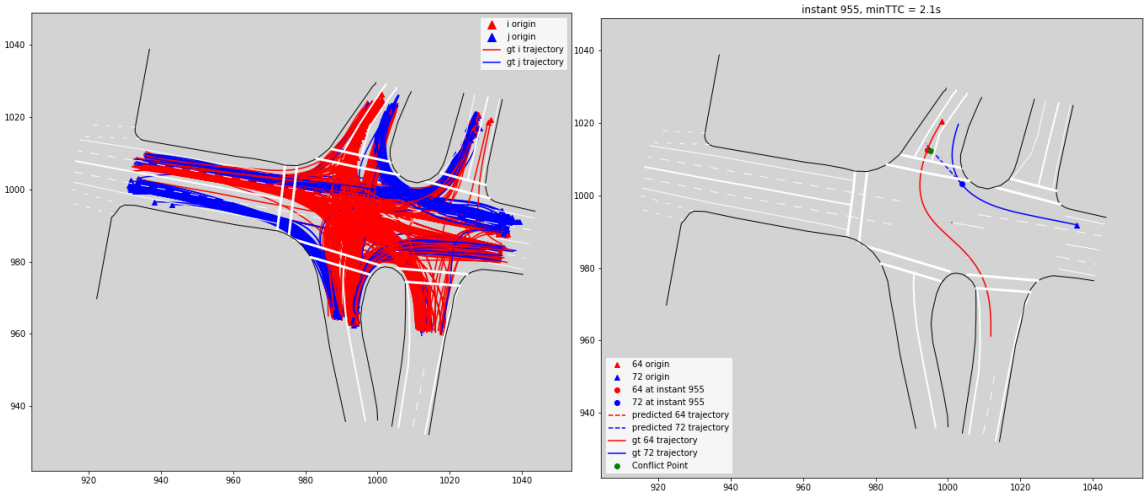
(a) overall (b) pattern 1 (c) pattern 2

Figure 2.9: Trajectory Pairs for GL Cluster 1



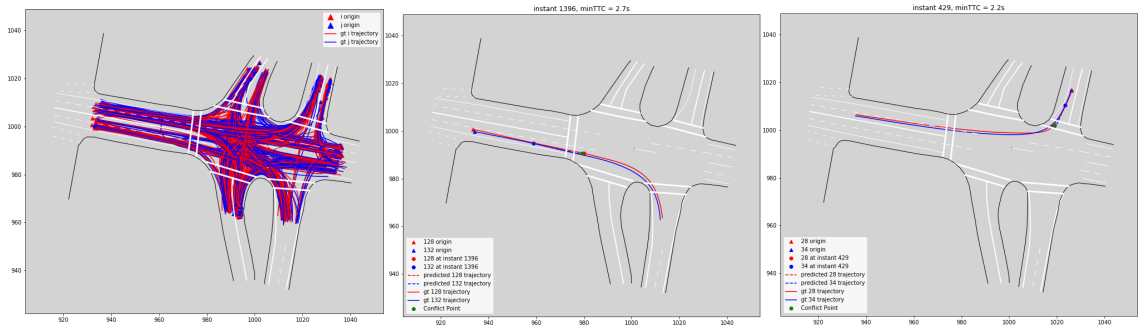
(a) overall (b) pattern 1

Figure 2.10: Trajectory Pairs for GL Cluster 2



(a) overall (b) pattern 1

Figure 2.11: Trajectory Pairs for GL Cluster 3



(a) overall (b) pattern 1 (c) pattern 2

Figure 2.12: Trajectory Pairs for GL Cluster 4

Table 2.1: Intersecting rate for trajectory pairs in GL clusters

C	Num. Pairs Intersected	Total Num. Pairs	Intersecting Rate
0	302	1026	0.29
1	492	688	**0.72**
2	271	882	0.31
3	78	963	0.08
4	243	318	**0.76**

Table 2.2: Mean of minimum of SSMs over interactions within each GL clusters

TTC (s)	i v (m/s)	j v (m/s)	i long acc (m/s^2)	j long acc (m/s^2)
2.87	3.15	9.64	0.1	-0.22
3.16	3.95	4.04	-0.57	**-1.45**
3.2	7.43	4.38	1.14	-0.63
2.33	0.3	5.9	-0.04	-0.36
3.24	0.22	1.15	**-1.35**	**-1.48**

We can see that most of the trajectory pairs in cluster 1 and 4 are intersecting or overlapping with each other while those in cluster 0, 2 and 3 are completely separated. In order to quantify this property, we calculate the intersecting rate of the trajectory pairs for each cluster which is defined in Equation 2.7 and the results are shown in Table 2.1.

$$\text{Intersecting Rate} = (\text{Num. Intersecting Traj Pairs})/(\text{Total Traj Pairs}) \tag{2.7}$$

We can see that the intersecting rates for cluster 1 and 4 are high while those for cluster 0, 2 and 3 are relatively low. Combining these numbers with the statistical summary in Table 2.2, we can reconstruct the driving scenarios in each cluster. The interactions in cluster 1 are angle conflicts. They happen when vehicle i is already in the intersection coming from the cross street and vehicle j is entering the intersection from the main street. However, vehicle i does not decelerate and continues with its movement (mean of minimum longitudinal acceleration $-0.57m/s^2$ and mean of minimum speed $3.95m/s$) since it has the right of way while vehicle j has to decelerate (mean of minimum longitudinal acceleration $-1.45m/s^2$) to avoid

crashing. For cluster 4, the interactions are rear end conflicts. They happen when vehicle i is decelerating (mean of minimum longitudinal acceleration $-1.35m/s^2$) and stops (mean of minimum speed $0.22m/s$) at a leg of intersection while vehicle j is approaching vehicle i and has to decelerate (mean of minimum longitudinal acceleration $-1.48m/s^2$) to avoid a collision.

For cluster 0, interactions are non-conflicts and that occur when both vehicles are in the intersection and traveling at high speed. Neither vehicle performs an evasive action (mean of minimum longitudinal accelerations are $0.1m/s^2$ and $-0.22m/s^2$ for vehicle i and j) since vehicle i intended to turn right and vehicle j intended to go straight at the critical moment (the instant at minTTC along the interaction sequence) and their future trajectories do not intersect. For cluster 2, non-conflict interactions happen when two vehicles are leaving the intersection. Unlike the merging conflict in Figure 2.9c, neither of the vehicles needs to yield since there are two lanes on this leg. For cluster 3, non-conflict interactions happen when vehicle i stops at the stop sign on one leg and vehicle j is leaving the intersection on the same leg. Even though the mean of minimum TTC in this cluster is the smallest among the five clusters, they are not traffic conflict because neither of the vehicles performs evasive action and their future trajectories never intersect.

Roundabout FT

We process the latent representations for FT with the same steps as for GL and find that using four clusters fits the data best as shown in Figure 2.13. Similarly, the trajectory pair plots and their main patterns are shown in Figure 2.14-2.17, and the intersecting rate calculated with Equation 2.7 and statistic summary are shown in Table 2.3 and 2.4. We conclude that cluster 1 are angle conflicts and cluster 2 are rear end conflicts while cluster 0 and cluster 3 are non-conflicts.

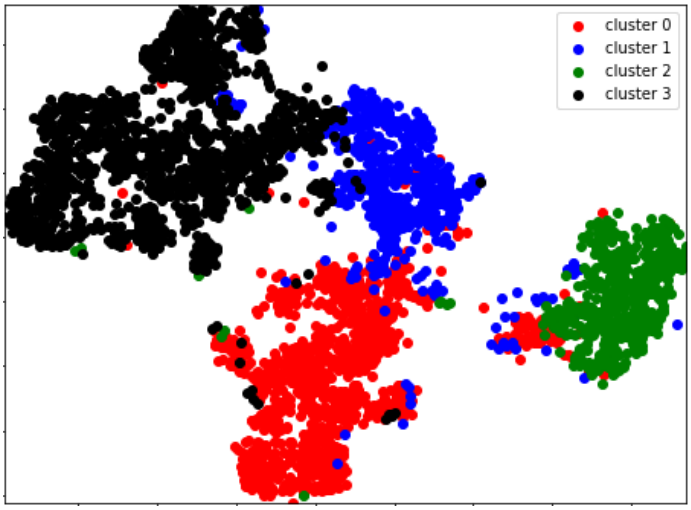
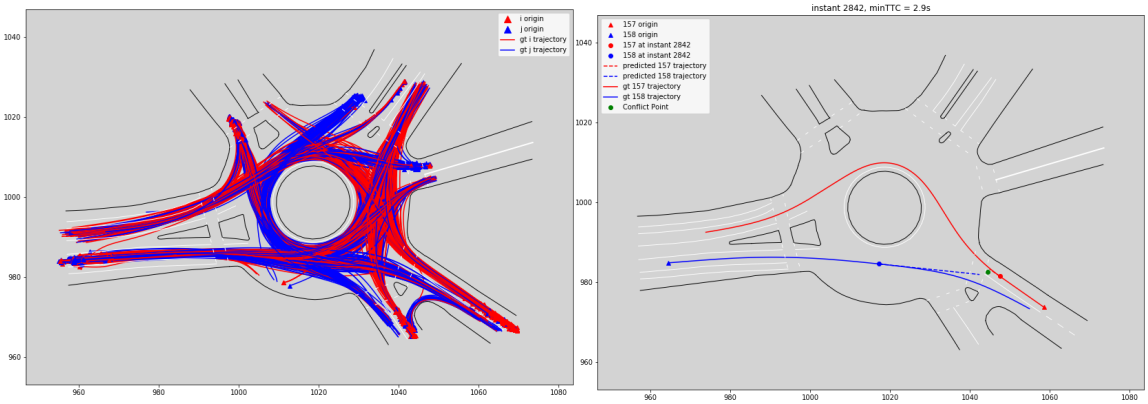
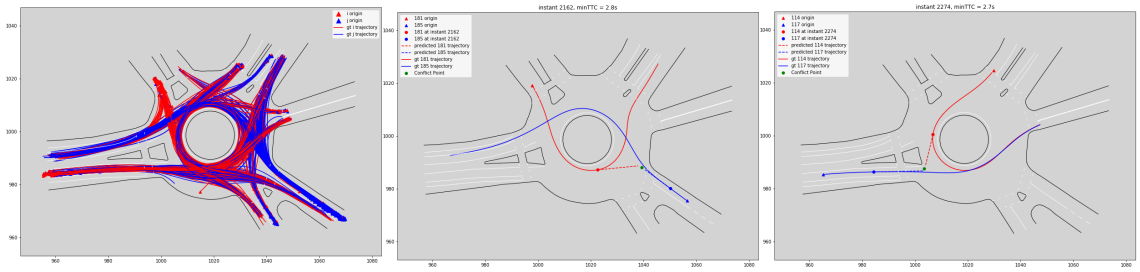


Figure 2.13: Clusters of Interactions at FT



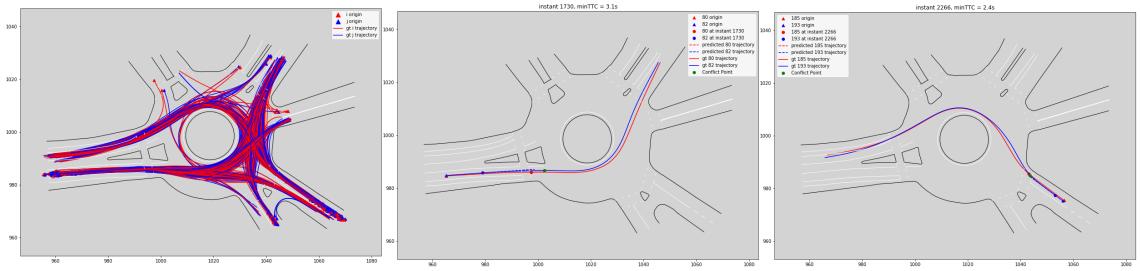
(a) overall (b) pattern 1

Figure 2.14: Trajectory Pairs for FT Cluster 0



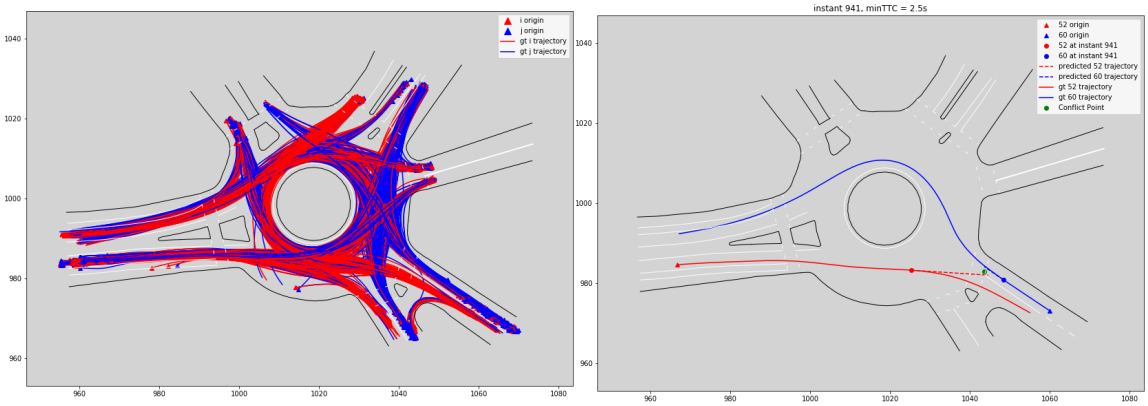
(a) overall (b) pattern 1 (c) pattern 2

Figure 2.15: Trajectory Pairs for FT Cluster 1



(a) overall (b) pattern 1 (c) pattern 2

Figure 2.16: Trajectory Pairs for GL Cluster 2



(a) overall (b) pattern 1

Figure 2.17: Trajectory Pairs for FT Cluster 3

Table 2.3: Intersecting rate for trajectory pairs in FT clusters

C	Num. Pairs Intersected	Total Num. Pairs	Intersecting Rate
0	701	1102	0.64
1	487	597	**0.82**
2	472	572	**0.83**
3	695	1498	0.46

Table 2.4: Mean of minimum of SSMs over interactions within each FT clusters

TTC (s)	i v (m/s)	j v (m/s)	i long acc (m/s^2)	j long acc (m/s^2)
2.95	1.03	5.76	-0.44	-0.28
3.15	5.61	5.29	-0.18	**-1.49**
3.26	0.54	1.6	**-1.39**	**-1.8**
2.67	6.25	1.31	0.12	-0.48

We can see that the intersecting rates for cluster 2 and 3 are substantially higher than those for cluster 0 and 3. The driving scenarios for each cluster can be reconstructed. The interactions in cluster 2 are angle conflicts. They happen when vehicle i is in the roundabout and vehicle j is entering the roundabout from the main street. Vehicle i has the right of way and it continues with its movement (mean of minimum longitudinal acceleration $-0.18m/s^2$ and mean of minimum speed $5.61m/s$) while vehicle j has to yield to vehicle i (mean of minimum longitudinal acceleration $-1.45m/s^2$) in order to avoid crash since their future trajectories intersect. For cluster 3, the interactions are rear end conflicts. They happen when vehicle i is decelerating (mean of minimum longitudinal acceleration $-1.39m/s^2$) and stops (mean of minimum speed $0.54m/s$) at a leg of intersection while vehicle j is approaching vehicle i and has to decelerate (mean of minimum longitudinal acceleration $-1.8m/s^2$) to avoid a collision

For clusters 0 and 3, interactions are not conflicts and they happen when one vehicle stops at stop sign on one leg and the other vehicle is leaving the intersection on the same leg. Both clusters have small mean minimum TTC but they are not traffic conflict because neither of the vehicle perform evasive action and their future trajectories never intersect.

Discussion

We perform the same process for all intersections and roundabouts and intersecting rate tables and statistic summary tables can be found in Appendix B.1. We identify similar interaction types from the remaining locations as we do in intersection GL and roundabout FT. We summarize the properties of the interaction types: rear end conflict, angle conflict and non conflict in Table 2.5. We also find that the average length of the interaction time series data from rear end conflict cluster is longer (3 seconds) than the average length of angle conflict clusters (1 second) and non-conflict clusters (1 seconds). This is because the two vehicles in a rear end conflict have overlapping trajectories and similar directions of movement so they have a longer time window to be exposed to a collision risk.

Table 2.5: Properties of Interaction types

Interaction Type	Trajectory Pairs	min TTC	i v	j v	i long acc	j long acc	Duration
Rear end conflict	Overlapped	low	low	low	Large deceleration	Large deceleration	Long
Angle conflict	Intersected	low	high	high	Near 0	Large deceleration	Short
Non conflict	Separated	lowest	low/ high	high	Near 0	Near 0	Short

The traffic conflict clusters identified from our approach is aligned with the two theoretical definition of traffic conflict since the clusters have both low proximity-based SSM (low TTC) as well as large evasive action-based SSM (large deceleration). Moreover, as we dive deeper into the interactions in the traffic conflict and non-conflict clusters and ask what contributes to the large deceleration in the traffic conflict clusters, we find that the traffic conflict interactions all have some critical conditions of vehicle i and j that are unexpected by vehicle j so he needs to take evasive action to avoid crashes while the non-conflict interactions have conditions of vehicle i and j expected by vehicle j so vehicle j does not need to take evasive action even though the conditions are critical.

For example, the pattern 1 of GL cluster 1 (conflict cluster) in Figure 2.9b shows that, vehicle j (blue line) is on the main street that has no signal control or stop sign and he expects himself to go through the intersection freely. However, an abnormal

situation happens where there is already vehicle i (red line) from the cross street in the intersection and crossing so vehicle j has to do large deceleration. The pattern 2 of GL cluster 1 in Figure 2.9c also shows that, vehicle j was on the main street and is leaving the intersection through leg B. He thought that he could leave the intersection freely but unexpectedly, vehicle i is already in the intersection and also leaving the intersection through leg B. Since leg B has only one lane, two vehicles need to merge and vehicle j has to yield for vehicle i. On the other hand, pattern 1 of GL cluster 0 (non-conflict cluster) in Figure 2.8b shows that everything goes normally as vehicle j is on the main street and can go through the intersection without yielding and vehicle i is making a right turn to leave the intersection. Similarly, pattern 1 of GL cluster 2 (non-conflict cluster) in Figure 2.10b shows that vehicle j predicts the behavior of vehicle i well as they will travel on two different lanes of leg E and leave the intersection smoothly.

The pattern 2 of FT cluster 1 (conflict cluster) in Figure 2.15c shows that vehicle j from leg A with only yield control expects himself to go through the roundabout smoothly but vehicle i who is already in the roundabout forces vehicle j to yield. The pattern 1 of FT cluster 2 (conflict cluster) in Figure 2.16b shows the similar situation where vehicle i is on leg A is blocking vehicle j approaching from behind which forces vehicle j to yield. On the contrary, pattern 1 of FT cluster 0 (non-conflict cluster) in Figure 2.14b shows that vehicle j already in the roundabout expects himself to travel smoothly and sees vehicle i stopping at the stop sign which matches his expectation.

Based on the above deep-dived examples, we can conclude that a traffic conflict consists of a series of events where there is a critical condition of vehicle i and j, vehicle j perceives this condition as abnormal and vehicle j takes evasive action. These are three events happening sequentially within an interaction and if we define the conditions of vehicle i and j as event A, the perception of vehicle j as event B and the action of vehicle j as event C, their relations and the resulting interaction types are shown in Figure 2.18. The traditional proximity-based approach only considers the critical condition of i and j (event A) while ignores the perception of vehicle j and the action of vehicle j. If vehicle j perceives the critical condition of i and j as normal, he will not take evasive action and the interaction will not be a traffic conflict. The traditional evasive action-based approach only considers the evasive action of vehicle j (event C) while ignores conditions of vehicle i and j. Since braking habits are subjective, there might be some precautionary braking from vehicle j [5] and the conditions of vehicle i and j are not critical.

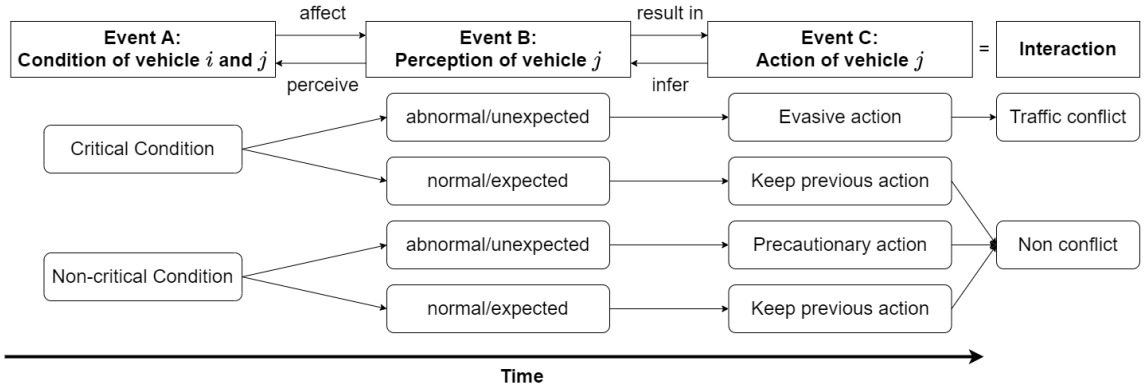


Figure 2.18: Relation of Event A, B, C and the Resulting Interaction Types

However, as a human way to observe an interaction, we look at the entire duration of the interaction and ask what is the conditions of vehicle i and j, are they abnormal to vehicle j and what is the corresponding action by vehicle j. Therefore, it is important to analyze the time series of surrogate safety measures and find the patterns from these time series data to identify traffic conflict. Fortunately, our deep unsupervised learning approach is trained with time series SSM data which mimics the human way of observing interactions. In our model, the conditions of vehicle i and j are represented by the time series of TTC, v_i and v_j, the action of vehicle j is represented by the time series of a_j. Even though we did not explicitly input the perception of vehicle j as a variable to the model, the model can learn it from the profile pattern of a_j sequence since a hard brake has a large jerk rate which infers the condition is unexpected while keeping previous action has 0 jerk rate which infers the condition is expected. To prove this, we calculated the mean of the minimum value of the jerk rate sequence for the GL and FT clusters and found that the jerk rate in traffic conflict clusters ($-4m/s^3$) are much larger than that in non-conflict clusters ($-0.5m/s^3$).

We also use the traditional threshold-based method to identify traffic conflicts in GL and FT locations where interactions that have minimum TTC less than two seconds are identified as traffic conflicts. To check the agreement of the threshold-based method and the proposed method, Table 2.6 and 2.7 show the distribution of traffic conflicts using threshold-based method in the clusters from the proposed method. The clusters with bold numbers are interpreted as traffic conflict clusters from the previous section. We can see that most of the traffic conflicts identified by the threshold-based method are in the non-conflict clusters (90% for GL and 75% for FT) and thus lots of false positive traffic conflicts are detected.

Table 2.6: Distribution of Traffic conflicts using threshold-based method in GL clusters

c	Total Interations	Interactions with min TTC ≤ 2
0	1026	160
1	688	19
2	882	21
3	963	326
4	318	10

Table 2.7: Distribution of Traffic conflicts using threshold-based method in FT clusters

c	Total Interations	Interactions with min TTC ≤ 2
0	1102	235
1	597	58
2	572	112
3	1498	273

Finally, after identifying the informative clusters for each location, we are ready to identify whether an interaction is a traffic conflict in the future without re-doing the interpretation process. We can simply encode the interaction as a latent representation by the transformer encoder model and assign to the cluster that has the closest distance between the cluster mean and the latent representation as the interaction type for the new interaction.

2.6 Conclusion

This paper proposed a deep unsupervised learning approach to identify traffic conflicts and found the universal properties of traffic conflict from the identified results. First we built a transformer encoder model and trained it with the unsupervised pre-training strategy. The model then learned the representations of multiple sequences of surrogate safety measures which consist of sequences of TTC, speed and longitudinal acceleration. The latent representations were clustered through agglomerative clustering and the clusters were interpreted as rear end conflict, angle conflict and non-conflict by plotting the trajectory pairs and calculating the statistical summaries. The method was validated on data from 8 real-world locations and the results were similar across all locations. We found that rear end conflict had overlapping trajectory pairs, low min TTC, low speed and large deceleration and angle conflict had intersecting trajectory pairs, low min TTC, high speed and large deceleration while non-conflict had lowest min TTC but separated trajectory pairs and no evasive action which is aligned with two definitions of traffic conflicts. Moreover, the identified traffic conflicts contain critical conditions from the two vehicles in an interaction and one vehicle perceives them as abnormal and takes evasive action to avoid crashes.

The advantages of this method are that it utilizes both proximity and evasive action-based surrogate safety measures to identify traffic conflicts instead of only one of them. Moreover, this approach allows conflicts to be detected based on sequences of several surrogate safety measures, rather than simply applying a single threshold to a specific measure. By exploiting these data, we can more reliably classify vehicle interactions as conflicts and non-conflicts, as well as categorize conflicts into different subtypes. This paper shows clustering as one of the downstream tasks that uses the latent representation of sequences of SSM, and moreover, the latent representation can be used in other downstream tasks like prediction and classification.

Chapter 3

Traffic Conflict Validation by Probabilistic Time Series Prediction

3.1 Introduction

Traditional crash-based safety analysis has many limitations since crash data has small sample size which leads to unobserved heterogeneity [49, 51], lacks detailed information describing the crash process, and can improve traffic safety only after crashes happen [45, 39]. On the other hand, traffic safety analysis using surrogate safety measures (SSM) have attracted more and more interest, since it can provide fast, pro-active safety analysis, while also yielding insights on the pre-crash process and crash failure mechanism by studying near misses [83, 82].

The theoretical foundation of surrogate safety indicators assumes that all traffic events are related to safety. These traffic events have different degree of severity (unsafety) and a relationship exists between the severity and the frequency of events shown as Figure 3.1 [36]. The severity of an event is often measured by the proximity in space or time two road users. Time to collision (TTC) [29] and post-encroachment time (PET) [2] are two most often used indicators. TTC is the time remaining before the collision if the involved road users continue with their respective speeds and trajectories and can be calculated as long when vehicles are on a collision course. The minimum TTC during an interaction is compared to a pre-defined threshold ($1.5s$ [71]) to determine whether this event is a traffic conflict or a normal interaction.

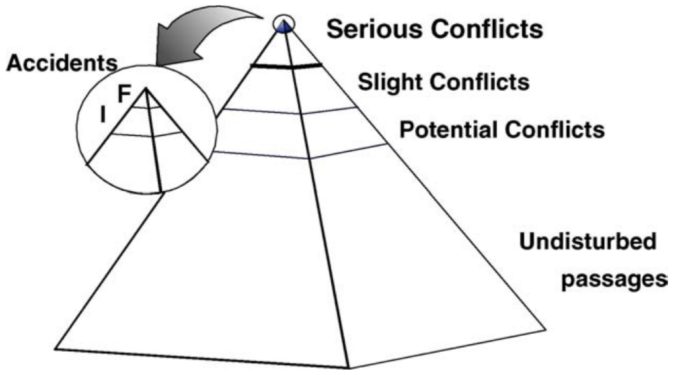


Figure 3.1: Safety Pyramid - the Interaction between Road Users as a Continuum of Events [36]

If the relationships between the layers of the safety pyramid is known, it is theoretically possible to calculate the frequency of the very severe but infrequent events (accidents) based on the known frequency of the less severe but more frequently occurring events [78]. However, connecting traffic conflicts to crashes is still an open question and several methods have been proposed [101]. [28] proposed a regression model to relate conflicts and crashes. [13] used a structural equation to model causal relationships among initial condition, action and crash outcome to estimate the crash probability. [76] used the extreme value theory (EVT) to model the distribution of TTC and calculated the probability of TTC reaching the extreme level (TTC = 0) as the crash probability. A more detailed review of these methods is in the next section.

With the development of deep unsupervised learning in computer science field, generative models [42] can learn the distribution of the data and generate new samples that are similar to the original data. Some of the models have been applied in the transportation field. [12] used generative adversarial network (GAN) to generate traffic accident and [15] used probability graphic model to generate safety-critical scenarios. By incorporating neural networks, such as Long short-term memory (LSTM) and transformer, that can deal with time series data, probabilistic time series prediction models [72] can predict the distribution of the data every time step. Therefore, the distribution of time series of TTC can be estimated and the crash probability can be calculated with probabilistic time series prediction.

In this paper, we propose a non-crash-based method to relate surrogate safety measures to crashes based on the causal model. Our main contributions are:

1. The method uses transformer-MAF to predict real time crash probability by estimating the probability density function of surrogate safety measures for every

time step.

2. The method implements the dependency structure among condition, action and crash outcome from the causal model into the probability density functions with an autoregressive network.

3. The method overcomes the limitations of the causal model and uses all values of condition, action and crash outcome by treating them as continuous variables .

4. We estimates the model on real-world traffic data to compare the crash probability under traffic conflict and normal interaction scenarios and calculate the effectiveness of evasive action.

3.2 Literature Review

Connecting SSMs to Crashes

The regression-based method can directly model the relationship between traffic conflict and crashes given the count of both data. [27] used linear regression model with form as $\lambda = \pi \cdot c$ where λ is the number of crashes on an entity during a certain period of time, c is the number of traffic conflicts on the same entity of the same time and π is the crash-to-conflict ratio. [17] estimated the counts of traffic conflict from traffic volume with Poisson-lognormal model and incorporated it in a safety performance function with negative binomial model. The regression-based model are easy to understand and apply but this approach still requires crash data which suffers from the same issues of the traditional road safety analysis and the crash-to-conflict ratios may vary for different road entities and time periods [97].

The EVT method can extrapolate the distribution of the observed traffic conflicts to the unobserved crashes to calculate the crash probability as shown in Figure 3.2a. Traffic conflicts are measured by SSMs like TTC and if TTC reaches the extreme level (TTC = 0), traffic conflicts would become crashes. The risk of crash can be calculated as Equation 3.1

$$R = \Pr(Z \geq 0) = 1 - G(0) \tag{3.1}$$

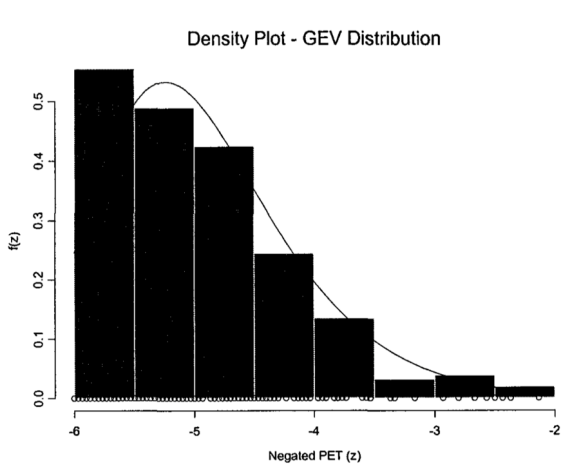
where R is the risk of crash, Z is the negated TTC, and $G(\cdot)$ is the generalized extreme value distribution or the generalized Pareto distribution. There is growing interest in using EVT for traffic conflict-based safety estimation through application of advanced statistical methods. [102, 100] used bivariate generalized Pareto distribution to estimate crashes with several different SSMs. With the combined use of different indicators, the model provides a more holistic approach to measure the severity of

an event. [98, 99] developed Bayesian hierarchical extreme value models to combine traffic conflicts from different sites for crash estimation in order to overcome the problem that severe traffic conflicts are rare for each individual site. However, the traffic conflict indicators are mainly proximity metrics such as TTC and PET while evasive action-based indicators are overlooked. Moreover, the statistical models have their inherent model assumptions like the parameters of GEV distribution are linearly related to the site properties [19] that sometimes the data do not follow. Additionally, indicators used in EVT are the extreme of a sequence of SSMs so the temporal correlations in the conflicts are not explored, and the EVT method can only be applied to a site-level safety analysis instead of an individual real-time crash estimation.

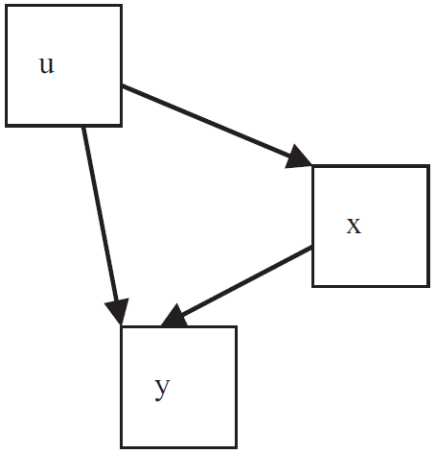
The causal model is another non-crash-based method where the crash outcome y of an event depends on its initial condition u and action x shown as Figure 3.2b. The probability distribution of crash outcome is given by Equation 3.2:

$$p(y, x, u) = (y|x, u)p(x|u)p(u) \tag{3.2}$$

where $p(u)$ is the probability distribution of the initial condition and $p(x|u)$ is the conditional probability distribution of action under the initial condition. The crash probability is by summing the probabilities of all the actions that could lead to a crash [13]. The model can also lead to a natural interpretation of the counterfactual element in the definition of conflict and [90] combined the causal model and the potential outcome model [65] to create a traffic conflict measure that can quantify the effectiveness of a given evasive action taken by a driver to avoid crashes. However, there are lots of assumptions for this causal model such as defining a set of initial conditions U and a set of evasive actions X. It is complicated to estimate the probability distribution for all possible evasive actions and initial condition and the studies that employ this definition usually focus on a small subset of possible interactions and participants [5].



(a) Distribution of TTC estimated from EVT model [75]



(b) Dependency structure among initial condition u, action x and crash outcome y [13]

Figure 3.2: Illustration of the EVT model and the causal model

We can conclude that the core idea behind both non-crash-based methods, EVT and causal model, are probability density estimation. EVT tries to estimate the distribution of the extreme values of surrogate safety measures like TTC while causal model tries to estimate the distribution of initial condition u, the conditional distribution of action x as well as the condtional distribution of crash outcome y. Both methods require many constraints and assumptions.

Probabilistic Time Series Prediction

Deep learning has been proven to be powerful in many tasks like prediction and sample generation when there is sufficient training data and computing resources [46]. In the field of deep unsupervised learning, many generative models like the flow model [32], variational auto-encoder (VAE) [41] and GAN [24] have been used for density estimation and sample generation. Their main applications are in image generation and text generation. Combined with neural network structures like LSTM and transformer that can deal with time series data, these generative models can estimate density functions for each time step. Many different types of model structures [69, 68, 81] have been tested on several multivariate time series datasets from the UCI benchmark [8]. Many of these generative models use the autoregressive network [20] to

improve their model performance since the network imposes a dependency relationship between the previous estimated variable and the current estimated variable.

3.3 Methodology

In this section, we propose a framework to train the transformer-MAF model and sample sequence and calculate action and crash probability from the trained model. The diagram of the framework is summarized and visualized in Figure 3.3. The data we use are high dimensional time series data containing v_i, the speed for vehicle i, v_j, the speed for vehicle j, a_i, the longitudinal acceleration for vehicle i,a_j the longitudinal acceleration for vehicle j, and TTC. We define the condition $u = (v_i, v_j)$, the action $x = (a_i, a_j)$ and crash outcome $y = TTC$. Therefore, each data point D contains a sequence of (u_t, x_t, y_t) $\forall t = 1, \cdots, T$ where t represents the current time step and T is the sequence length. We did not use the distance between two vehicles in the condition u because the previous TTC and speed in the observed sequence could imply the distance information.

During training, the high dimensional time series data D is divided into an observed sequence and a target sequence. The observed sequence is fed into the transformer encoder-decoder model and the model outputs the latent representation k which will be called as the context vector in this paper. We then pass the context vector k and the current time step data (u_t, x_t, y_t) together into the MAF model. The parameters of the density function for time t are estimated for each data and each time step and used to calculate the log-likelihood of the data which is the loss function.

During prediction, the trained transformer-MAF is used. Only the observed sequence is fed into the model and the context vector k is computed as in the training phase. Using the context vector k, the MAF model estimates the parameters for the density function at time t and samples the current time step data $(\hat{u}_t, \hat{x}_t, \hat{y}_t)$ autoregressively. Finally the density functions are used to computes the conditional action probability, the crash probability and conditional crash probability.

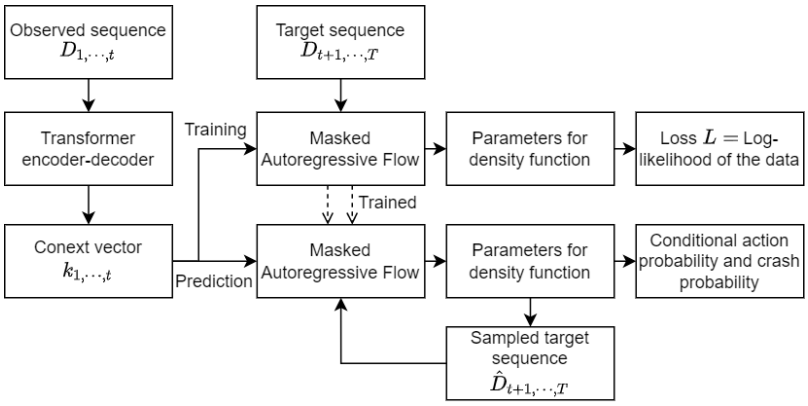


Figure 3.3: Workflow of the model

Masked Autoregressive Flow Model

In this section, we will first introduce the 1-dimensional flow model and then extend it into a high dimensional flow model where masked autoregressive flow is one of its variants. First, we will explain why flow model is better in our use case compared to other deep unsupervised model like VAE and GAN. The goal of this research is to do density estimation for the SSM data $a = (u, x, y)$ and calculate the action and crash probability. Therefore, we need to explicitly estimate a density function $p(u, x, y)$ for the SSM data while VAE and GAN model do not meet this requirement. Another advantage of the flow model is that it can deal with continuous variables easily. One common objective for density estimation is to maximize the log likelihood of the data as Equation 3.3, however, condition u, action x and TTC y are all continuous variables and it is difficult to ensure two conditions in Equation 3.4 for a proper distribution since a continuous variable can take any value in a range.

$$\max_{\theta} \sum_{i} \log p\left(a^{(i)}\right) \tag{3.3}$$

$$\int_{-\infty}^{+\infty} p(a)\, da = 1 \quad \text{and} \quad p(a) > 0 \quad \forall a \tag{3.4}$$

However, flow model can transform these unknown and complex continuous random variables (u, x, y) into other random variables (z_1, z_2, z_3) with known and simple probability density functions p_{Z_1}, p_{Z_2} and p_{Z_3}. A common choice for them is standard Gaussian distribution [16]. We will use $z_1 = f_\theta(u)$ as an example where f_θ is the transformation. With the change of variable theorem, we have Equation 3.5-3.7 and therefore the loss function becomes easy to compute given that p_{Z_1} is

standard Gaussian. There are two requirements for the transformation f_θ. One is differentiability since we need to calculate the derivative of f_θ and the other one is invertibility since we need to calculate $\hat{u}$ from f_θ^{-1}. Fortunately, using a deep neural network as the transformation f_θ satisfies these two requirements with suitable activation functions [63]. During sampling, we can sample $z_1 \sim N(0,1)$ and then calculate $\hat{u} = f_\theta^{-1}(z_1)$.

$$\int p(u)du = \int p_{Z_1}(z_1)dz_1 = 1 \tag{3.5}$$

$$p(u) = p_{Z_1}(z_1)|\frac{dz_1}{du}| = p_{Z_1}\left(f_\theta(u)\right)\left|\frac{\partial f_\theta(u)}{\partial u}\right| \tag{3.6}$$

$$\max_\theta \sum_i \log p\left(u^{(i)}\right) = \max_\theta \sum_i \log p_{Z_1}\left(f_\theta\left(u^{(i)}\right)\right) + \log\left|\frac{\partial f_\theta}{\partial u}\left(u^{(i)}\right)\right| \tag{3.7}$$

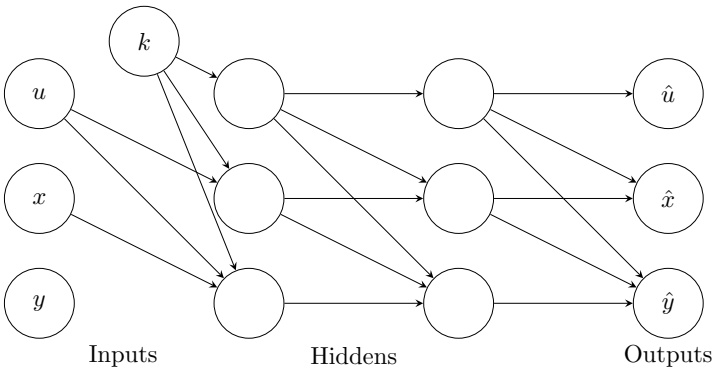
For high dimensional data, the training process is the same and researchers usually add an autoregressive structure between each dimension to increase the performance of the neural network[62, 35]. According to the causal model from [13], there is a dependency relationship among condition u, action x and crash outcome y (in our case TTC) shown in Figure 3.2b. Therefore, adding this autoregressive structure can not only improve the model performance but also impose some physical meanings on the model and increase its interpretability. With this structure, we can calculate the conditional action probability, crash probability and conditional crash probability which are shown in the next sub-section. The transformation between (u, x, y) and (z_1, z_2, z_3) are shown in Equation 3.8-3.10. We will first sample z_1 to get $\hat{u}$ and then sample z_2 and combine with $\hat{u}$ to get $\hat{x}$ and lastly sample z_3 and combine with $\hat{u}$ and $\hat{x}$ to get $\hat{y}$. The sampling process has to been done step by step and it becomes slow for much higher dimensional data.

Training: $z_1 = f_\theta(u)$ Sampling: $\hat{u} = f_\theta^{-1}(z_1)$ (3.8)

$z_2 = f_\theta(x; u)$ $\hat{x} = f_\theta^{-1}(z_2; \hat{u})$ (3.9)

$z_3 = f_\theta(y; x, u)$ $\hat{y} = f_\theta^{-1}(z_3; \hat{x}, \hat{u})$ (3.10)

Based on [20], we design an autoregressive neural network shown as Figure 3.4 to ensure the dependency structure among (u, x, y). k is the context vector from the Transformer encoder-decoder and all output neurons should have its information. Mask A is applied on the fully connected network between inputs and the first hidden layers and mask B is applied on the two hidden layers and the outputs. They guarantee the information from the inputs flows properly into the outputs.



(a) Connections between neurons under autoregressive structure

$$\mathrm{A} = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 1 & 0 \end{bmatrix}$$

$$\mathrm{B} = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

(b) Masks applied on a fully connected network to ensure dependency structure

Figure 3.4: Design autoregressive neural network for the SSM data

Action Prediction and Crash Probability Calculation

The transformer encoder-decoder and the autoregressive flow model allow us to explicitly estimate three probability density function $p_t(u|k)$, $p_t(x|u,k)$ and $p_t(y|x,u,k)$ for each future time step t, we can calculate the conditional action probability $P_t(X|U,k)$, crash probability $P_t(y \leq 0|k)$ and conditional crash probability $P_t(y \leq 0|X_t, U_t, k)$ where X and U represent some events of the random variables x and u and k is the observed context vector. Since all the density functions contain time step t and the context vector k, we will omit them in the following calculation for simplicity but keep in mind that different context vectors can drastically change the density functions which will be shown in the next section.

The conditional action probability is the conditional probability of some action X can happen under some condition U which is shown in Equation 3.11. We can evaluate this probability over any interval of action and under any interval of condition. For example, we can define evasive action as acceleration within range $[-6, -3]m/s^2$ and no action as acceleration within range $[-0.5, 0.5]m/s^2$ and the probability of the driver doing evasive action and the probability of the driver doing no action under some condition U can be calculated by replacing the corresponding intervals.

$$P(x \in X|u \in U) = \frac{\int_X \int_U p(x|u)p(u)\,dx\,du}{\int_U p(u)\,du} \tag{3.11}$$

The crash probability is the marginal probability of the crash outcome exceeding its critical point, in our case $TTC \leq 0$. This marginal probability can be written as the joint probability of a crash happens and all the possible actions x and conditions u are considered since $P(x \in \mathbb{R}) = 1$ and $P(u \in \mathbb{R}) = 1$. Introducing x and u can help us calculate the crash probability because our transformer-MAF model does not directly estimate the probability density of crash outcome $p(y)$ but the conditional probability density of crash outcome given action and condition $p(y|x, u)$. The calculation is shown in Equation 3.12. However, in practice, the values of actions like acceleration can not span the entire real number set and we shrink the integration interval to the minimum and maximum values in the empirical data. And the condition u like speed can only change slightly between each time step ($0.1s$ for our data) so we replace the real number set with the range from the 25th-percentile value to the 75-th percentile value $[U_{25\text{th}}, U_{75\text{th}}]$ as Equation 3.13.

$$P(y \leq 0) = P(y \leq 0, x \in \mathbb{R}, u \in \mathbb{R}) = \int_{-\infty}^{0} \int_{R} \int_{R} p(y|x, u)p(x|u)p(u)\, dy\, dx\, du \tag{3.12}$$

$$\approx \int_{-\infty}^{0} \int_{X_{\min}}^{X_{\max}} \int_{U_{25\text{th}}}^{U_{75\text{th}}} p(y|x, u)p(x|u)p(u)\, dy\, dx\, du \tag{3.13}$$

The conditional crash probability is the conditional probability of a crash can happen given some action X is taken and under some condition U shown in Equation 3.14. For a traffic conflict event, we can calculate the conditional crash probability if the no action was taken and compare it to the conditional crash probability given evasive action was taken by replacing the corresponding intervals. With this counterfactual experiment, we can explore the causal effect of the action on the crash outcome of an event.

$$P(y \leq 0|x \in X, u \in U) = \frac{\int_{-\infty}^{0} \int_{X} \int_{U} p(y|x, u)p(x|u)p(u)\, dy\, dx\, du}{\int_{X} \int_{U} p(x|u)p(u)\, dx\, du} \tag{3.14}$$

Since we only know the probability density functions $p(u)$, $p(x|u)$ and $p(y|x, u)$, we can use the Monte Carlo method [21] to calculate these integrals. However, the traditional Monte Carlo integration method can only work with finite intervals and can not handle the interval for y $(-\infty, 0]$ which contains infinite limit. We use the change of variable method to map this interval to some finite interval and multiply the derivative to the integrand so that the traditional Monte Carlo method works properly [55]. The derivation of Equation 3.11-3.14 can be found in Appendix B.1.

3.4 Experiments and Results

Model training and evaluation

The dataset we use for the study is INTERACTION dataset [94] which is extracted from videos of three unsignalized intersections and five roundabouts taken by drones and traffic cameras. The dataset records the position (x, y), velocity (v_x, v_y) and yaw angle θ for each vehicle within videos at every time step of 0.1 second. Using the raw data, we calculate the longitudinal speed and acceleration and TTC with the constant speed assumption. We also fix the sequence length for each interaction to be 20 time steps (2 sec) where the first 10 (1 sec) is the observed sequence and the last 10 (1 sec) is the target sequence. The total shape of the entire dataset after processing is $(55055, 20, 5)$ which means there are 55055 interactions as data points and each data point contains 5 sequences with length 20.

The entire dataset is split into training, validation and testing set by 80/10/10 and the model is trained on the training dataset with Adam optimizer and evaluated on validation set after every epoch. The hyperparameters for the model are shown in Table A.1 and the training and validation loss plot is shown in Figure 3.5. We select the model based on the best validation loss and use mean square error (MSE) and continuous ranked probability score (CRPS) to measure the accuracy of the trained model on the test set. For each test data, we sample 1000 sequences from the model. For MSE, we calculate the median value of the 1000 samples and compute the mean square error between the median and the ground truth sequence. For CRPS, it is defined as Equation 3.15 where F is the cumulative function of random variable X and x is the observation [54] and it is often used to measure the difference between the predicted cumulative distribution function (CDF) and the empirical CDF of the observation. We use the weighted quantile loss to approximate CRPS similar to other implementations in Python package gluonts [1] and properscoring [23]. The final testing MSE is 0.17 and CRPS is 0.1.

$$\mathrm{CRPS}(F, x) = \int_{-\infty}^{\infty} (F(y) - \mathbb{1}(y \geq x)^2)\, dy \tag{3.15}$$

Table 3.1: Hyperparameters for transformer MAF model

Parameter	Value
Activation Function	Gelu
Model Dimension	40
Feedforward Dimension	160
Dropout Rate	0.1
Num. Attention Heads	8
Num. Encoder Blocks	3
Num. Decoder Blocks	3
Positional Encoding	Learnable
Num. Layers in MAF	2
Learning Rate	1e-3

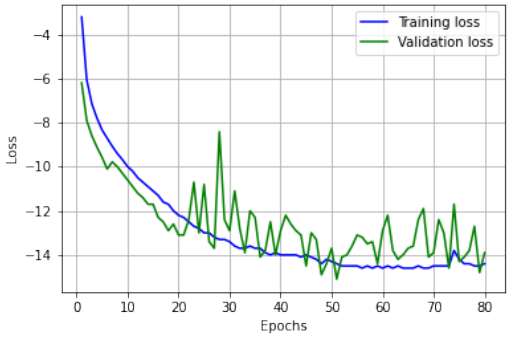
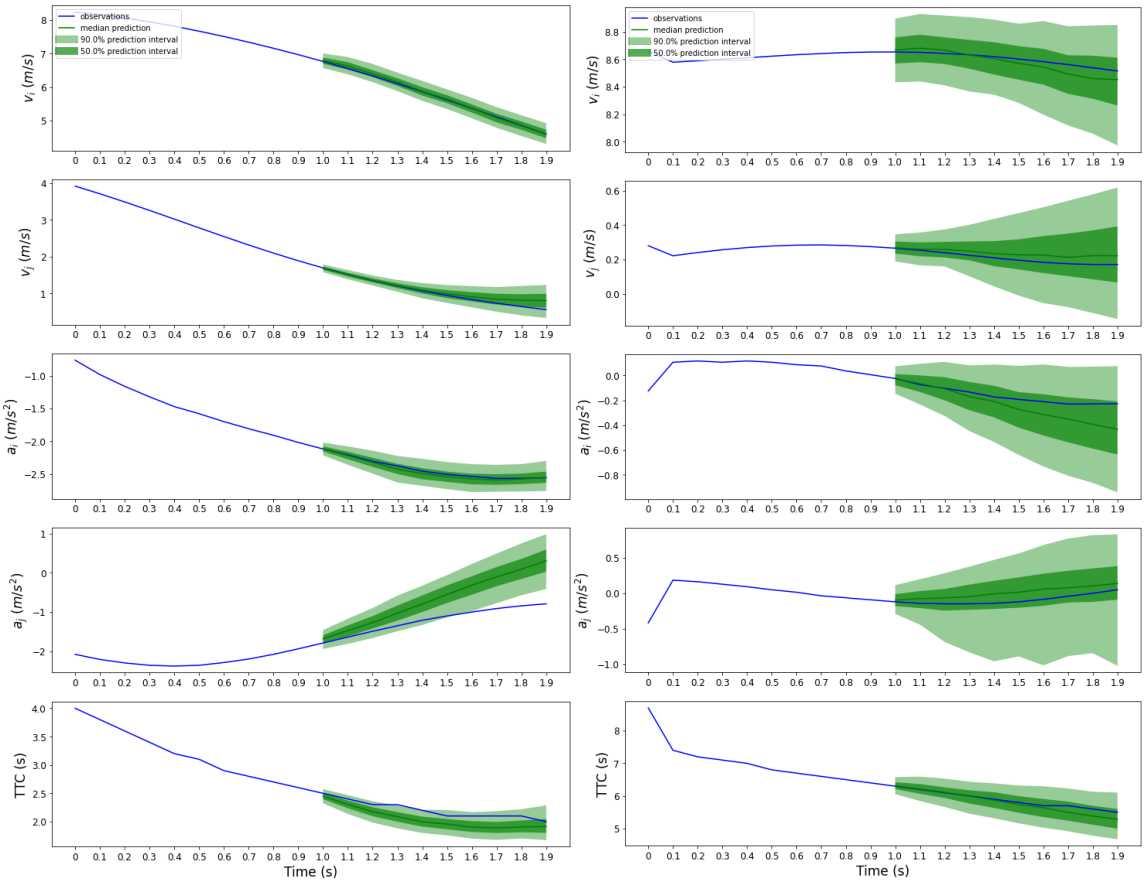


Figure 3.5: Training & Validation Loss

Some visualizations of ground truth vs sampled data are shown in Figure 3.6. Since our data is multidimensional time series data, each visualization contains 5 sub-figures. X axis is time and Y axis represents a feature. The blue line is the ground truth data while the dark green line is the median of the prediction. The dark green area is 50% prediction interval which is bound by the 25-percentile and 75-percentile predictions and the light green area is 90% prediction interval which is bound by the 5-percentile and 95-percentile predictions. The bound area increases as time increases since the next prediction is based on the observed sequence and the pass predictions and the uncertainty accumulates through time. The visualizations show that the model can predict the future SSMs reasonably well and more visualizations can be found in the Appendix B.2.



(a) Interaction with low TTC and large deceleration
(Traffic conflict)

(b) Interaction with high TTC and no deceleration
(Normal Interaction)

Figure 3.6: Example of probabilistic SSM prediction

We also implemented a non-autoregressive structure and compare the MSE and CRPS on the test set of two models. The non-autoregressive structure is shown in Figure 3.7. Compared to the autoregressive structure in Figure 3.4a, this structure removes the connections from the input u and x to the hidden neurons so that there is no information in u and x passed to the outputs. The hidden layers and the output layers are fully connected since there is no constraint on the context information and it can also increase the model performance. In Table 3.2, the result shows that the autoregressive structure can fit the data better which validates the dependency structure between u and x as well as x and y.

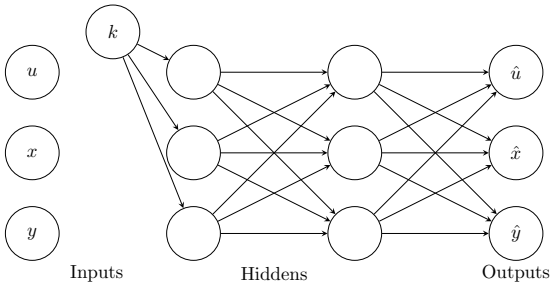


Figure 3.7: Connections between neurons under non-autoregressive structure

Table 3.2: MSE and CRPS of the autoregressive and non-autoregressive structure (the lower the better)

	Autoregressive	Non-autoregressive
MSE	0.17	0.24
CRPS	0.1	0.19

Probability Prediction

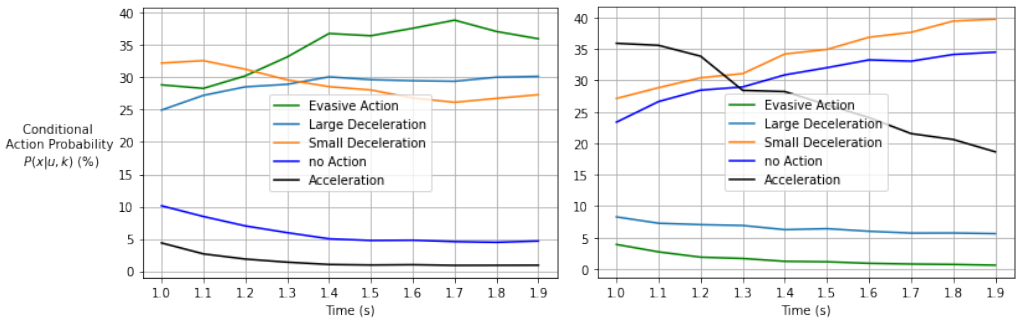
First we calculate the conditional action probability under the conditional $u \in [U_{25\text{th}}, U_{75\text{th}}]$ and two different contexts: traffic conflict and normal interaction. We define 5 actions with their names and value ranges shown in Table 3.3. X_{eva} represents an event when vehicle i is doing evasive action and vehicle j is doing all possible actions and similarly X_{no} represents an event when vehicle i is not taking any action and vehicle j is doing all possible actions. We only consider vehicle j doing all possible actions because there will be too many combinations with vehicle j taking 5 different actions. More importantly, from vehicle i's perspective, it does not know what action vehicle j will take so it has to optimize its action with all possible actions from vehicle j considered.

Table 3.3: Notations and Definitions for Different Actions

Action	Range
Evasive Action X_{eva}	$(a_i \in [-6, -3], a_j \in [-6, 6])$
Large Deceleration	$(a_i \in [-3, -2], a_j \in [-6, 6])$
Small Deceleration	$(a_i \in [-2, -0.5], a_j \in [-6, 6])$
No Action X_{no}	$(a_i \in [-0.5, 0.5], a_j \in [-6, 6])$
Acceleration	$(a_i \in [0.5, 6], a_j \in [-6, 6])$

The probability of 5 different actions are evaluated in Figure 3.8. We can validate these probabilities by summing them at each time step and the sum should be close to 100% since these 5 actions cover the entire range of action values. Under the traffic

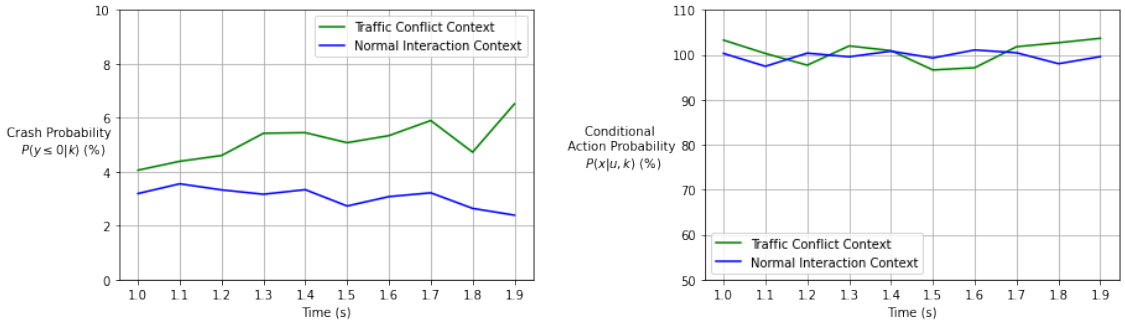
conflict context in Figure 3.6a, the model predicts that there will be high probability for evasive action and large deceleration (over 30% and 25%) and low probability for no action and acceleration (below 10% and 5%) to happen in the future 10 time steps. The probability for evasive action gradually grows as time increases. This aligns with the ground truth and the sampled sequence of a_i in Figure 3.6a since the actual and predicted speed values are decreasing over time. On the other hand, under the context of normal interaction in Figure 3.6b, the model predicts that there will be high probability for no action and small deceleration (over 25% and 30%) and low probability for evasive action and large deceleration (below 5% and 10%) to happen in the future 10 time steps. The probability of acceleration decreases drastically (from 40% to 20%) and the mass is shifted toward small acceleration and no action whose probabilities both increase around 10%. This aligns with the sampled sequence of a_i in Figure 3.6b because 90% of the predicted values are within $[-0.2, 0.1]$ at time 1.0 and 75% of the predicted values are within $[-0.9, -0.2]$ at time 1.9.



(a) Traffic conflict context (b) Normal Interaction context

Figure 3.8: Conditional action probability vs. time under different contexts

We also calculate the crash probability under different contexts using Equation 3.13. In Figure 3.9a, the model predicts the crash probability under traffic conflict context increases from 4% to 6% over time since the predicted TTC values in Figure 3.6a are decreasing and getting closer and closer to 0. On the other hand, the crash probability for normal interaction context is around 2.5% over the entire future 10 steps since the predicted TTC values in Figure 3.6b are high and far away from 0. We also calculate the conditional action probability with $x \in X_{\text{all}} = (a_i \in [-6, 6], a_j \in [-6, 6])$ in Figure 3.9b and they are around 100% for both contexts which also validates the correctness of the Monte Carlo integration process since theoretically $P(x \in \mathbb{R}|U, k) = 1$.



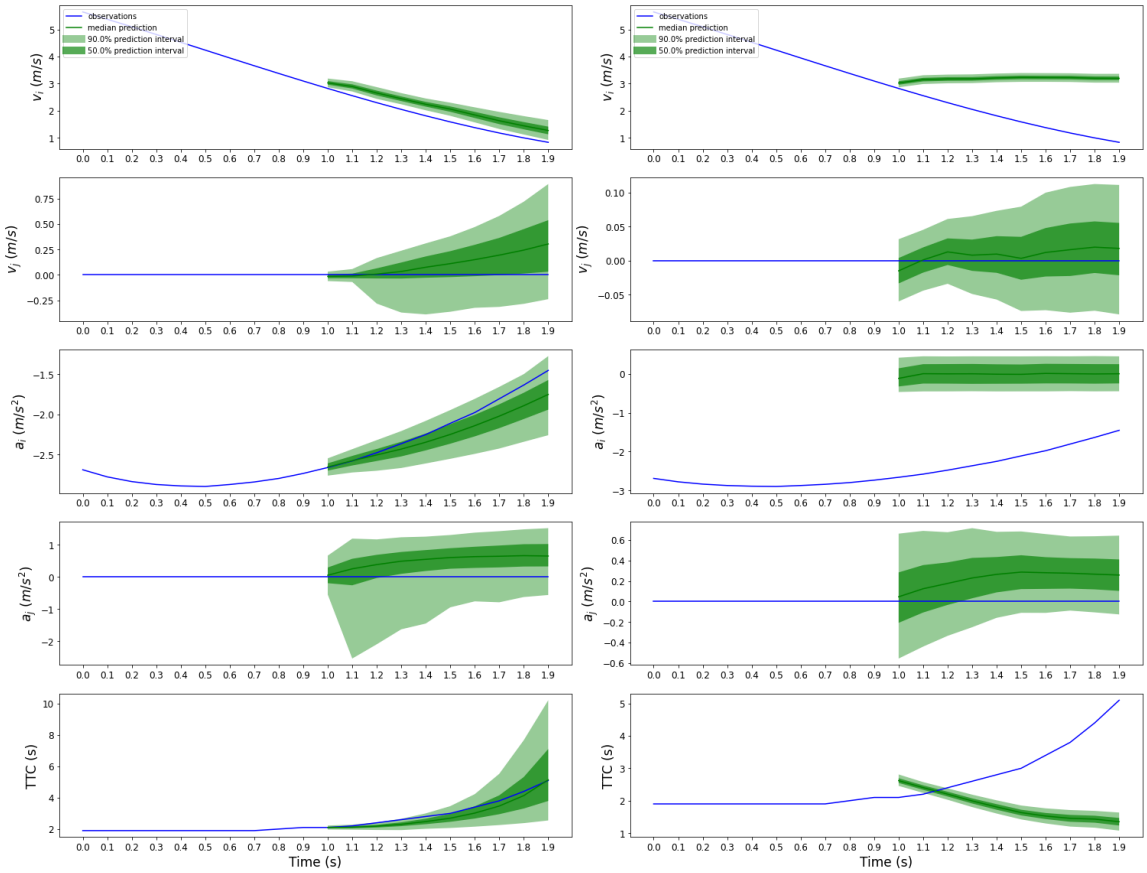
(a) Crash probability vs. time

(b) Conditional action probability vs. time

Figure 3.9: Considering all possible actions under different scenarios

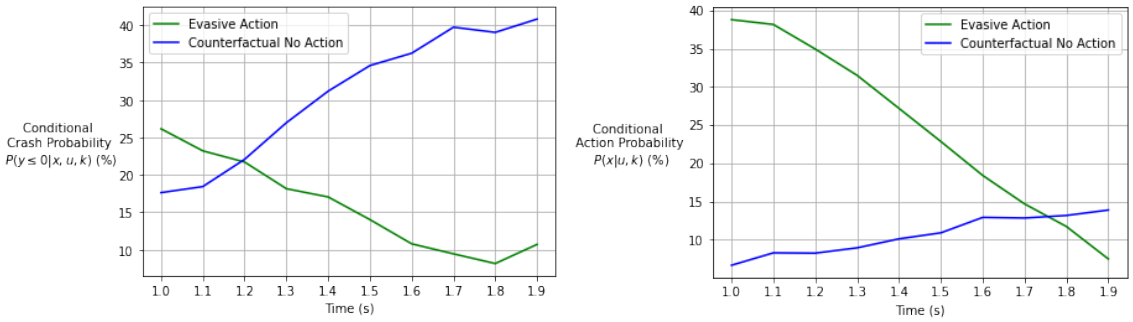
We select a rear-end conflict in Figure 3.10a where vehicle i is approaching to vehicle j from behind and vehicle j is already stopped as our counterfactual experiment. In the counterfactual scenario, vehicle i didn't perform evasive action but instead take no action from time 1.0 to time 1.9 (flat green zone in a_i of Figure 3.10b), therefore the model is forced to take this "fake" no action as input to predict the future condition and future TTC as Figure 3.10b. The predicted v_i becomes a flat green zone with small variation since the counterfactual a_i is small and can only take values in $[-0.5, 0.5]$. The model also predicts that the future TTC will continue to decrease if vehicle i takes no action. The conditional crash probabilities $P(y \leq 0|x \in X, u \in [U_{25\text{th}}, U_{75\text{th}}])$ for evasive action X_{eva} and no action X_{no} are shown in Figure 3.11a. The result shows that evasive action is effective to avoid crashes because when the driver is doing evasive action, the conditional crash probability decreases over time while if the driver was doing no action, the conditional crash probability would increase over time. Moreover, in this counterfactual experiment, only the action taken by the driver is changed but not any other variables like condition u or context k.

The conditional crash probability is a lot higher than the crash probability and this is because in Equation 3.14, the conditional crash probability is the ratio of the crash probability and the action probability which are both small. Shown as the green curve in Figure 3.11b, the model predicts the needs for evasive action drops over time which matches the a_i curve in Figure 3.10a. On the other hand, the blue curve in Figure 3.11b shows the model initially predicts that counterfactual no action should not be taken (around 5% probability) but slowly increases this probability to 15% because a series of the counterfactual no actions are taken by the driver and those actions become a new context which affects the future prediction.



(a) Correct prediction: evasive action (b) Counterfactual prediction: no action

Figure 3.10: Correct prediction vs. counterfactual prediction on a traffic conflict context



(a) Conditional crash probability vs. time (b) Conditional action probability vs. time

Figure 3.11: Counterfactual example

Discussion

Our trained transformer-MAF model can sample accurate future sequences and estimate the conditional action probability, crash probability and conditional crash probability reasonably. Under the traffic conflict context, the model estimates a high probability (40%) of evasive action while under the normal interaction context, the model estimates a high probability (35%) of no action. This conditional action probability is achieved by masking the last 5 time steps of the observed action sequence ($x_{6,\cdots,10}$). Without this masking mechanism, the conditional action probability can reach 90% for X_{eva} under traffic conflict context which is problematic because predicting x_{11} given $x_{1,\cdots,10}$ is trivial and the context vector k would be dominated by the information of $x_{1,\cdots,10}$ [93]. The current action should not completely depend on its previous actions since in the context of traffic conflict, evasive action is often taken abruptly [39] and it should also depend on the perceived safety by the driver which means the driver knows that he is driving fast (condition u) and perceives that he might crash soon (TTC y). Therefore, we mask out the $x_{6,\cdots,10}$ to reduce the information from action sequence so that the model can learn a context with more information extracted from the observed condition and TTC sequence. Similar to the observed action sequence, we also mask the last 5 time steps of the observed TTC sequence. Without the mask, the context vector k contains too much information of the previous TTC and the model will create a density function $p(y|x, u, k)$ where x and u are ignored and it becomes $p(y|k)$. This means that no matter what current action x_t and current u_t are, the density function of $p(y|x, u, k)$ will not change and it fails our purpose to build the causal model among u, x and y as well as estimate the counterfactual probability. However, if we mask out the entire observed action

sequence or TTC sequence, the predicted density function and the sampled sequence will be off compared to the ground truth. Therefore we learn that there is a trade-off between the accuracy and variety of the predicted density function and we need to find the balance point so that the prediction has enough variation and high accuracy.

The crash probability under traffic conflict context is around 5% and under normal interaction context is around 3%. These values are a little bit higher than those reported in [99, 10] (around 3%). The reason is that we sub-sampled the original processed data. The entire processed data contains all the interactions that have a possible collision course and a lot of them (50%) have relatively high minimum TTC (greater than $4s$) over the possible collision course. Therefore, we removed those interactions with minimum TTC greater than $4s$ from the data and created a smaller dataset. This is good from a model training standpoint because the portion of interactions with low TTC ($< 2.5s$) is changed from 10% to 20% after the sub-sampling and it becomes easier for the model to capture the pattern of these minority interactions. On the other hand, the subsampling process also changed the underlying distribution of the entire dataset and it can result in a biased estimation of the density function. One way to ease this inflated crash probability problem could be multiplying the crash probability by 50% to counter the sub-sampling effect. We can also create a larger model that can directly work with the entire dataset or use other methods like re-weighting the loss to handle imbalanced class problem.

The results of the time series conditional action prediction as well as crash prediction can be applied in many cases. In a real time driving scenario, vehicles with advanced driver assistance sytem (ADAS) [44] or connected and autonomous vehicles (CAV) [18] can capture sequences of conditions u like speed and location and actions x like acceleration and steering of themselves and other surrounding objects and then calculate the crash outcome y like TTC. These data can be directly fed into our transformer-MAF model to predict a sequence of future conditional action probability and crash probability. The predicted action and crash probability can be incorporated into the collision avoidance system. In a scenario of safety assessment, we can select the max value of the predicted crash probability sequence as the crash probability of an interaction. For each site, we can calculate the average crash probability of the interactions from this site as the crash probability of the site and prioritize sites based on this value. Moreover, the average conditional evasive action probability can indicate whether large deceleration is often used in the interactions from a site which can give some insight for the safety investigators. It is important to note that the actions and the corresponding ranges defined in Table 3.3 can be changed under different driving scenarios as well as the crash interval $[-\infty, 0]$. For example, the crash interval can be changed to $[-\infty, 1]$ if the system needs a safer buffer when calculating the probability. This can be done without retraining the model because

because the model directly estimates a continuous probability density function of action x which provides flexibility on probability inference. If the model was trained on a discrete variable with pre-defined thresholds for different actions, the entire model needs to be retrained if the thresholds change.

The conditional crash probability allows us to test the counterfactual situation since it is defined as the crash probability given action X is taken and under condition U. With different counterfactual action given as input, we can calculate different conditional crash probability. Specifically, we can calculate the conditional crash probability $P_{\text{crash|no}} = P(y \leq 0|X_{\text{no}}, U, k)$ if the driver had taken no action and the conditional crash probability $P_{\text{crash|eva}} = P(y \leq 0|X_{\text{eva}}, U, k)$ if the driver had taken an evasive action under traffic conflict context. According to the potential outcome model, We define $E = P_{\text{crash|no}} - P_{\text{crash|eva}}$ to quantify the effectiveness of the evasive action to avoid crashing. The effectiveness under traffic conflict context in Figure 3.10 defined as E_{TC} is a lot higher than the effectiveness under normal interaction context in Figure B.2 defined as E_{NI} which is shown in Figure 3.12. This is logical because we would expect a high effect of evasive action in a traffic conflict context and no effect in a normal interaction context. The counterfactual prediction plot of the normal interaction and all probability tables can be found in Appendix B.3.

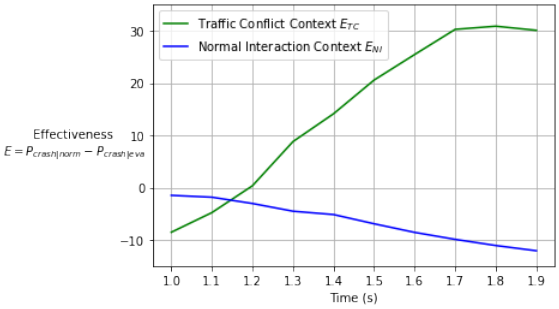


Figure 3.12: Effectiveness of evasive action under different context

3.5 Conclusion

This paper proposed a method to connect surrogate safety measures to crash probability via causal probabilistic time series prediction. The density functions of condition, action and crash outcome are estimated for each time step using transformer-MAF. The causal relationship among those variables are implemented in the neural network with autoregressive structure. The conditional action probability, crash probability

and conditional crash probability are calculated based on the estimated density functions. The results show that the sampled SSM sequences are accurate by measuring the MSE and CRPS. The estimated probabilities are comparable to those reported in the literature. Moreover, the effectiveness of evasive action to avoid crashes is evaluated with the potential outcome model.

Our method overcomes the limitations of the causal model with the help of deep learning and traffic data collection techniques. There are lots of improvements can be done to the current method. More variables such as relative distance between two vehicles can be added to condition u, steering can be added to action x and other types of crash outcome like PET can be incorporated as well. The current method can only predict the density function up to future 10 time steps and a larger model can be explored to estimate longer or variable-length sequences.

Chapter 4

Conclusion

Traffic safety analysis using surrogate safety measures has become more and more popular since traffic conflicts happen much more frequently than crashes and it contains the information about pre-crash event and crash avoidance mechanism. It is also pro-active to analyze conflicts than reactively analyze crashes. With the rapid development of connected and automated vehicles (CAV), surrogate safety analysis will become more prominent as more real-time in vehicle data will become more accessible.

There are two fundamental questions need to be answered. The first one is how to identify traffic conflicts and the second one is how to validate traffic conflicts. Most of the research uses one surrogate safety measure to classify a traffic event as traffic conflict based on a pre-defined threshold. However, there are limitations like each surrogate safety measure can only capture one aspect of the event and the pre-defined thresholds could change under different traffic environment. Researchers also validate surrogate safety measures by connecting them to crashes using crash-based method like regression modeling and non-crash-based method like causal model and extreme value theory (EVT). However, some of these methods still rely on crash data, some of the methods have too many model assumptions and some can only analyze the safeness of a site instead of a vehicle at micro level.

With these concerns, in this dissertation, we explored the use of deep unsuperivsed learning for these two problems. In particular, we incorporated the framework of transformer encoder with unsupervised pre-training and transformer masked autoregressive flow (MAF) to solve these two problems and the main results and findings are shown as follows.

4.1 Findings and Contributions

In Chapter 2, we studied the problem of traffic conflict identification. We proposed the transformer encoder with unsupervised pre-training technique to learn the representations of traffic interactions. Each traffic interaction containing speed profiles and acceleration profiles of two vehicles and time-to-collision is encoded into some latent space. In this method, both evasive action-based and proximity-based surrogate safety measures are used to identify traffic conflicts. Moreover, contrary to the traditional method that uses a pre-defined threshold for the minimum value of a sequence of surrogate safety measures to classify a traffic event, our method analyzes the pattern of the entire sequence of the surrogate safety measures by encoding them into latent representation. We clustered these latent representations and found out the similar characteristics within each cluster. For rear end conflicts, they have overlapping trajectory pairs, low minimum TTC, low speed and large deceleration while angle conflicts have intersecting trajectory pairs, low minimum TTC, high speed and large deceleration. For non-conflict clusters, they have the lowest minimum TTC but separated trajectory pairs and no evasive action. Moreover, the identified traffic conflicts contain abnormal conditions of one vehicle which is the precursor and contributes to the evasive action of the other vehicle in an interaction. Finally, we compared the clusters from our methods to the results from the traditional threshold-based method and found that there are lots of disagreements for two methods and around 90% of the traffic conflicts identified from the traditional method are in the non-conflict clusters identified from our method which means that the traditional threshold-based method creates lots of false positive in traffic conflict identification.

In Chapter 3, we studied the problem of traffic conflict validation. We proposed the transformer-MAF model to predict real time crash probability by estimating the probability density function of surrogate safety measures for every time step. Compared to the methods in the literature to connect traffic conflicts and crashes, our method does not require crash data and has few model assumptions. Moreover, Our method implements the dependency structure among condition, action and crash outcome from the causal model into the probability density functions with autoregressive network. The estimated probability density functions are used to sample future sequences and calculate conditional action probability, crash probability as well as conditional crash probability. The sampled future sequences are aligned with the future ground truth and the model predicts a high probability for the driver to do evasive action in the future 10 time steps under the traffic conflict context and a high probability to do no action under the normal interaction context. The predicted crash probability is 6% and 3% under traffic conflict context and normal interaction context respectively. It is a little bit higher than those reported in the literature because we

subsampled the data and the underlying distribution for conflicts is changed. Finally, the conditional crash probability is used to show the effectiveness of evasive action to avoid crashes in a counterfactual experiment and it shows that evasive action is a lot more effective under traffic conflict context than normal interaction context. Our method can be applied to advanced driver assistance sytem (ADAS) or connected and automated vehicles (CAV) to do real time evasive action prediction and crash prediction

4.2 Future Directions

There are lots of improvements can be done on our current approaches for traffic conflict identification and validation. More variables such as relative distance between two vehicles and steering can be added and other types of proximity-based surrogate safety measures like PET can be incorporated as well. The information of the infrastructure like traffic control and main/cross street information can be used to better capture the perception of the vehicles. Therefore, a more complex model or a different model structure like variational auto-encoder can be considered since masked autoregressive flow models need to sample variables sequentially and become slow when the number of variables increases. The current transformer-MAF model can only predict the density function up to future 10 time steps and a larger model can be explored to estimate longer or variable-length sequences.

There are also a lot of future research directions that the work in this dissertation can be further extended. The traditional TTC calculation assumes that the drivers will continue with their speed and direction but TTC can be also calculated by other motion prediction models. We can compare the distribution of TTC calculated by constant speed and direction model, LSTM/transformer trajectory prediction model and probabilistic trajectory prediction model using transformer flow model. Second, we can further extend the dependency structure in the transformer-MAF model to incorporate crash severity. Therefore, the model can predict the conditional crash severity given a crash happens. Moreover, the current methods are only applied to vehicle-to-vehicle interactions while missing the application on vulnerable road users (VRU). We can apply our methods on the vehicle-to-pedestrian interactions or vehicle-to-bicycle interactions by incorporating other surrogate safety measures which are more suitable for VRU. Another direction is that we can develop a system approach for action prediction and crash probability estimation of the ego vehicle under interactions with multiple vehicles. The current method only predicts action and crash probability for interactions separately and sometimes the optimal action for this interaction might result into a crash for other interactions. A global optimal

action and crash probability can be found by considering multiple vehicle interactions at the same time. Lastly, deep unsuperivsed learning can also be applied to many traffic safety problems like crash generation and sampling and crash report text mining.

Bibliography

[1] Alexander Alexandrov et al. "GluonTS: Probabilistic and Neural Time Series Modeling in Python". In: *Journal of Machine Learning Research* 21.116 (2020), pp. 1–6. URL: http://jmlr.org/papers/v21/19-820.html.

[2] Brian L Allen, B Tom Shin, and Peter J Cooper. *Analysis of traffic conflicts and collisions*. Tech. rep. 1978.

[3] FH Amundsen and C Hyden. "Proceedings of first workshop on traffic conflicts". In: *Oslo, TTI, Oslo, Norway and LTH Lund, Sweden* (1977).

[4] Jeffery Archer. "Indicators for traffic safety assessment and prediction and their application in micro-simulation modelling: A study of urban and suburban intersections". PhD thesis. KTH, 2005.

[5] Ashutosh Arun et al. "A systematic mapping review of surrogate safety assessment using traffic conflict techniques". In: *Accident Analysis & Prevention* 153 (2021), p. 106016.

[6] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. "Layer normalization". In: *arXiv preprint arXiv:1607.06450* (2016).

[7] Yoshua Bengio, Aaron Courville, and Pascal Vincent. "Representation learning: A review and new perspectives". In: *IEEE transactions on pattern analysis and machine intelligence* 35.8 (2013), pp. 1798–1828.

[8] Catherine Blake. "UCI repository of machine learning databases". In: *http://www. ics. uci. edu/˜ mlearn/MLRepository. html* (1998).

[9] Guillem Boquet et al. "A variational autoencoder solution for road traffic forecasting systems: Missing data imputation, dimension reduction, model selection and anomaly detection". In: *Transportation Research Part C: Emerging Technologies* 115 (2020), p. 102622.

[10] Attila Borsos et al. "Are collision and crossing course surrogate safety indicators transferable? A probability based approach using extreme value theory". In: *Accident Analysis & Prevention* 143 (2020), p. 105517.

[11] Ting Chen et al. "A simple framework for contrastive learning of visual representations". In: *arXiv preprint arXiv:2002.05709* (2020).

[12] Zhijun Chen et al. "Traffic Accident Data Generation Based on Improved Generative Adversarial Networks". In: *Sensors* 21.17 (2021), p. 5767.

[13] Gary A Davis et al. "Outline for a causal model of traffic conflicts and crashes". In: *Accident Analysis & Prevention* 43.6 (2011), pp. 1907–1919.

[14] Jacob Devlin et al. "Bert: Pre-training of deep bidirectional transformers for language understanding". In: *arXiv preprint arXiv:1810.04805* (2018).

[15] Wenhao Ding et al. "Learning to collide: An adaptive safety-critical scenarios generating method". In: *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2020, pp. 2243–2250.

[16] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. "Density estimation using real nvp". In: *arXiv preprint arXiv:1605.08803* (2016).

[17] Karim El-Basyouny and Tarek Sayed. "Safety performance functions using traffic conflicts". In: *Safety science* 51.1 (2013), pp. 160–164.

[18] David Elliott, Walter Keen, and Lei Miao. "Recent advances in connected and automated vehicles". In: *journal of traffic and transportation engineering (English edition)* 6.2 (2019), pp. 109–131.

[19] Chuanyun Fu and Tarek Sayed. "Random parameters Bayesian hierarchical modeling of traffic conflict extremes for crash estimation". In: *Accident Analysis & Prevention* 157 (2021), p. 106159.

[20] Mathieu Germain et al. "Made: Masked autoencoder for distribution estimation". In: *International Conference on Machine Learning*. PMLR. 2015, pp. 881–889.

[21] John Geweke. "Bayesian inference in econometric models using Monte Carlo integration". In: *Econometrica: Journal of the Econometric Society* (1989), pp. 1317–1339.

[22] William D Glauz and Donald J Migletz. *Application of traffic conflict analysis at intersections*. Tech. rep. 1980.

[23] Tilmann Gneiting and Adrian E Raftery. "Strictly proper scoring rules, prediction, and estimation". In: *Journal of the American statistical Association* 102.477 (2007), pp. 359–378.

[24] Ian Goodfellow et al. "Generative adversarial nets". In: *Advances in neural information processing systems* 27 (2014).

[25] Giuseppe Guido et al. "Comparing safety performance measures obtained from video capture data". In: *Journal of Transportation Engineering* 137.7 (2011), pp. 481–491.

[26] Yanyong Guo, Tarek Sayed, and Mohamed H Zaki. "Exploring evasive action–based indicators for PTW conflicts in shared traffic facility environments". In: *Journal of Transportation Engineering, Part A: Systems* 144.11 (2018), p. 04018065.

[27] Ezra Hauer. "Traffic conflicts and exposure". In: *Accident Analysis & Prevention* 14.5 (1982), pp. 359–364.

[28] Ezra Hauer and Per Garder. "Research into the validity of the traffic conflicts technique". In: *Accident Analysis & Prevention* 18.6 (1986), pp. 471–481.

[29] J Hayward. *Near misses as a measure of safety at urban intersections*. Pennsylvania Transportation and Traffic Safety Center, 1971.

[30] John C Hayward. "Near miss determination through use of a scale of danger". In: (1972).

[31] Kaiming He et al. "Deep residual learning for image recognition". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 770–778.

[32] Jonathan Ho et al. "Flow++: Improving flow-based generative models with variational dequantization and architecture design". In: *International Conference on Machine Learning*. PMLR. 2019, pp. 2722–2730.

[33] Sepp Hochreiter and Jürgen Schmidhuber. "Long short-term memory". In: *Neural computation* 9.8 (1997), pp. 1735–1780.

[34] Murtadha D Hssayeni et al. "Distracted driver detection: Deep learning vs handcrafted features". In: *Electronic Imaging* 2017.10 (2017), pp. 20–26.

[35] Chin-Wei Huang et al. "Neural autoregressive flows". In: *International Conference on Machine Learning*. PMLR. 2018, pp. 2078–2087.

[36] Christer Hydén. "The development of a method for traffic safety evaluation: The Swedish Traffic Conflicts Technique". In: *Bulletin Lund Institute of Technology, Department* 70 (1987).

[37] Karim Ismail, Tarek Sayed, and Nicolas Saunier. "Methodologies for aggregating indicators of traffic conflict". In: *Transportation research record* 2237.1 (2011), pp. 10–19.

[38] Carl Johnsson, Aliaksei Laureshyn, and Carmelo Dágostino. "Validation of surrogate measures of safety with a focus on bicyclist–motor vehicle interactions". In: *Accident Analysis & Prevention* 153 (2021), p. 106037.

[39] Carl Johnsson, Aliaksei Laureshyn, and Tim De Ceunynck. "In search of surrogate safety indicators for vulnerable road users: a review of surrogate safety indicators". In: *Transport Reviews* 38.6 (2018), pp. 765–785.

[40] Ankit Kathuria and Perumal Vedagiri. "Evaluating pedestrian vehicle interaction dynamics at un-signalized intersections: a proactive approach for safety analysis". In: *Accident Analysis & Prevention* 134 (2020), p. 105316.

[41] Diederik P Kingma and Max Welling. "Auto-encoding variational bayes". In: *arXiv preprint arXiv:1312.6114* (2013).

[42] Durk P Kingma et al. "Semi-supervised learning with deep generative models". In: *Advances in neural information processing systems* 27 (2014).

[43] Mark A Kramer. "Nonlinear principal component analysis using autoassociative neural networks". In: *AIChE journal* 37.2 (1991), pp. 233–243.

[44] Vipin Kumar Kukkala et al. "Advanced driver-assistance systems: A path toward autonomous vehicles". In: *IEEE Consumer Electronics Magazine* 7.5 (2018), pp. 18–25.

[45] Aliaksei Laureshyn et al. "Review of current study methods for VRU safety. Appendix 6–Scoping review: surrogate measures of safety in site-based road traffic observations: Deliverable 2.1–part 4." In: (2016).

[46] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. "Deep learning". In: *nature* 521.7553 (2015), pp. 436–444.

[47] Honglak Lee et al. "Unsupervised feature learning for audio classification using convolutional deep belief networks". In: *Advances in neural information processing systems* 22 (2009), pp. 1096–1104.

[48] Yuebiao Li, Zhiheng Li, and Li Li. "Missing traffic data: comparison of imputation methods". In: *IET Intelligent Transport Systems* 8.1 (2014), pp. 51–57.

[49] Dominique Lord and Fred Mannering. "The statistical analysis of crash-frequency data: A review and assessment of methodological alternatives". In: *Transportation research part A: policy and practice* 44.5 (2010), pp. 291–305.

[50] Laurens van der Maaten and Geoffrey Hinton. "Visualizing data using t-SNE". In: *Journal of machine learning research* 9.Nov (2008), pp. 2579–2605.

[51] Fred L Mannering and Chandra R Bhat. "Analytic methods in accident research: Methodological frontier and future directions". In: *Analytic methods in accident research* 1 (2014), pp. 1–22.

[52] Fred Mannering. "Temporal instability and the analysis of highway accident data". In: *Analytic methods in accident research* 17 (2018), pp. 1–13.

[53] Fred Mannering et al. "Big data, traditional data and the tradeoffs between prediction and causality in highway-safety analysis". In: *Analytic methods in accident research* 25 (2020), p. 100113.

[54] James E Matheson and Robert L Winkler. "Scoring rules for continuous probability distributions". In: *Management science* 22.10 (1976), pp. 1087–1096.

[55] Cameron Mcelfresh. "Monte Carlo Integration: Improper Integrals". In: (2021). Available at `https://cameron-mcelfresh.medium.com/monte-carlo-integration-improper-integrals-b73d1af03011` (accessed 4.15.2022).

[56] Tomáš Mikolov et al. "Recurrent neural network based language model". In: *Eleventh annual conference of the international speech communication association*. 2010.

[57] Daniel Müllner. "Modern hierarchical, agglomerative clustering algorithms". In: *arXiv preprint arXiv:1109.2378* (2011).

[58] Ying Ni et al. "Evaluation of pedestrian safety at intersections: A theoretical framework based on pedestrian-vehicle interaction patterns". In: *Accident Analysis & Prevention* 96 (2016), pp. 118–129.

[59] A.J. Nicholson. "The variability of accident counts". In: *Accident Analysis & Prevention* 17.1 (1985), pp. 47 –56. ISSN: 0001-4575. DOI: `https://doi.org/10.1016/0001-4575(85)90007-7`. URL: `http://www.sciencedirect.com/science/article/pii/0001457585900077`.

[60] Aaron van den Oord et al. "Wavenet: A generative model for raw audio". In: *arXiv preprint arXiv:1609.03499* (2016).

[61] Alkis Papadoulis, Mohammed Quddus, and Marianna Imprialou. "Evaluating the safety impact of connected and autonomous vehicles on motorways". In: *Accident Analysis & Prevention* 124 (2019), pp. 12–22.

[62] George Papamakarios, Theo Pavlakou, and Iain Murray. "Masked autoregressive flow for density estimation". In: *Advances in neural information processing systems* 30 (2017).

[63] George Papamakarios et al. "Normalizing flows for probabilistic modeling and inference". In: *Journal of Machine Learning Research* 22.57 (2021), pp. 1–64.

[64] MR Parker Jr and Charles V Zegeer. *Traffic conflict techniques for safety and operations: Observers manual*. Tech. rep. United States. Federal Highway Administration, 1989.

[65] Judea Pearl. "Causal inference in statistics: An overview". In: *Statistics surveys* 3 (2009), pp. 96–146.

[66] Victor Marin Puchades et al. "The role of perceived competence and risk perception in cycling near misses". In: *Safety science* 105 (2018), pp. 167–177.

[67] Alec Radford, Luke Metz, and Soumith Chintala. "Unsupervised representation learning with deep convolutional generative adversarial networks". In: *arXiv preprint arXiv:1511.06434* (2015).

[68] Kashif Rasul et al. "Autoregressive denoising diffusion models for multivariate probabilistic time series forecasting". In: *International Conference on Machine Learning*. PMLR. 2021, pp. 8857–8868.

[69] Kashif Rasul et al. "Multivariate probabilistic time series forecasting via conditioned normalizing flows". In: *arXiv preprint arXiv:2002.06103* (2020).

[70] Emanuele Sacchi and Tarek Sayed. "Conflict-based safety performance functions for predicting traffic collisions by type". In: *Transportation Research Record* 2583.1 (2016), pp. 50–55.

[71] Emanuele Sacchi, Tarek Sayed, and Paul Deleur. "A comparison of collision-based and conflict-based safety evaluations: The case of right-turn smart channels". In: *Accident Analysis & Prevention* 59 (2013), pp. 260–266.

[72] David Salinas et al. "DeepAR: Probabilistic forecasting with autoregressive recurrent networks". In: *International Journal of Forecasting* 36.3 (2020), pp. 1181–1191.

[73] Peter T Savolainen et al. "The statistical analysis of highway crash-injury severities: a review and assessment of methodological alternatives". In: *Accident Analysis & Prevention* 43.5 (2011), pp. 1666–1676.

[74] Tarek Sayed, Gerald Brown, and Francis Navin. "Simulation of traffic conflicts at unsignalized intersections with TSC-Sim". In: *Accident Analysis & Prevention* 26.5 (1994), pp. 593–607.

[75] Praprut Songchitruksa. "Innovative non-crash-based safety estimation: An extreme value theory approach". PhD thesis. Purdue University, 2004.

[76] Praprut Songchitruksa and Andrew P Tarko. "The extreme value theory approach to safety estimation". In: *Accident Analysis & Prevention* 38.4 (2006), pp. 811–822.

[77] Paul St-Aubin. "Driver behaviour and road safety analysis using computer vision and applications in roundabout safety". PhD thesis. Ecole Polytechnique, Montreal (Canada), 2016.

[78] Åse Svensson and Christer Hydén. "Estimating the severity of safety related behaviour". In: *Accident Analysis & Prevention* 38.2 (2006), pp. 379–385.

[79] Ahmed Tageldin, Tarek Sayed, and Khaled Shaaban. "Comparison of time-proximity and evasive action conflict measures: case studies from five cities". In: *Transportation research record* 2661.1 (2017), pp. 19–29.

[80] Ahmed Tageldin, Tarek Sayed, and Xuesong Wang. "Can time proximity measures be used as safety indicators in all driving cultures? Case study of motorcycle safety in China". In: *Transportation Research Record* 2520.1 (2015), pp. 165–174.

[81] Binh Tang and David Matteson. "Probabilistic Transformer For Time Series Analysis". In: *Advances in Neural Information Processing Systems* 34 (2021).

[82] Andrew Tarko. *Measuring road safety with surrogate events*. Elsevier, 2019.

[83] Andrew Tarko et al. "Surrogate measures of safety. White paper". In: *Transportation Research Board, Washington, DC* (2009).

[84] Ashish Vaswani et al. "Attention is all you need". In: *Advances in neural information processing systems*. 2017, pp. 5998–6008.

[85] Navreet Virdi et al. "A safety assessment of mixed fleets with connected and autonomous vehicles using the surrogate safety assessment module". In: *Accident Analysis & Prevention* 131 (2019), pp. 95–111.

[86] WHO. *Road traffic injuries*. Available at `https://www.who.int/news-room/fact-sheets/detail/road-traffic-injuries` (accessed 4.21.2022). 2021.

[87] Julia Werneke, Marco Dozza, and MariAnne Karlsson. "Safety–critical events in everyday cycling–Interviews with bicyclists and video annotation of safety–critical events in a naturalistic cycling study". In: *Transportation Research Part F: Traffic Psychology and Behaviour* 35 (2015), pp. 199–212.

[88] Kun-Feng Wu and Paul P Jovanis. "Defining and screening crash surrogate events using naturalistic driving data". In: *Accident Analysis & Prevention* 61 (2013), pp. 10–22.

[89] Kun Xie et al. "Use of real-world connected vehicle data in identifying high-risk locations based on a new surrogate safety measure". In: *Accident Analysis & Prevention* 125 (2019), pp. 311–319.

[90] Kentaro Yamada and Manabu Kuroki. "New traffic conflict measure based on a potential outcome model". In: *Journal of Causal Inference* 7.1 (2019).

[91] Zhuoning Yuan, Xun Zhou, and Tianbao Yang. "Hetero-convlstm: A deep learning approach to traffic accident prediction on heterogeneous spatio-temporal data". In: *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2018, pp. 984–992.

[92] Mohamed H Zaki, Tarek Sayed, and Khaled Shaaban. "Use of drivers' jerk profiles in computer vision–based traffic safety evaluations". In: *Transportation Research Record* 2434.1 (2014), pp. 103–112.

[93] George Zerveas et al. "A Transformer-based Framework for Multivariate Time Series Representation Learning". In: *arXiv preprint arXiv:2010.02803* (2020).

[94] Wei Zhan et al. "Interaction dataset: An international, adversarial and cooperative motion dataset in interactive driving scenarios with semantic maps". In: *arXiv preprint arXiv:1910.03088* (2019).

[95] Minghua Zhang et al. "Learning universal sentence representations with mean-max attention autoencoder". In: *arXiv preprint arXiv:1809.06590* (2018).

[96] Zheng Zhao et al. "LSTM network: a deep learning approach for short-term traffic forecast". In: *IET Intelligent Transport Systems* 11.2 (2017), pp. 68–75.

[97] Lai Zheng, Karim Ismail, and Xianghai Meng. "Traffic conflict techniques for road safety analysis: open questions and some insights". In: *Canadian journal of civil engineering* 41.7 (2014), pp. 633–641.

[98] Lai Zheng and Tarek Sayed. "Application of extreme value theory for before-after road safety analysis". In: *Transportation research record* 2673.4 (2019), pp. 1001–1010.

[99] Lai Zheng and Tarek Sayed. "Bayesian hierarchical modeling of traffic conflict extremes for crash estimation: a non-stationary peak over threshold approach". In: *Analytic methods in accident research* 24 (2019), p. 100106.

[100] Lai Zheng and Tarek Sayed. "From univariate to bivariate extreme value models: approaches to integrate traffic conflict indicators for crash estimation". In: *Transportation research part C: emerging technologies* 103 (2019), pp. 211–225.

[101] Lai Zheng, Tarek Sayed, and Fred Mannering. "Modeling traffic conflicts for use in road safety analysis: A review of analytic methods and future directions". In: *Analytic methods in accident research* 29 (2021), p. 100142.

[102] Lai Zheng et al. "Bivariate extreme value modeling for road safety estimation". In: *Accident Analysis & Prevention* 120 (2018), pp. 83–91.

Appendix A

Traffic Conflict Identification by Representation Learning

A.1 Model Hyperparameters

Table A.1: Hyperparameters for transformer encoder model and training

Parameter	Value
Activation Function	Gelu
Model Dimension	128
Feedforward Dimension	256
Dropout Rate	0.2
Num. Attention Heads	16
Num. Encoder Blocks	3
Positional Encoding	Learnable
Warm up Step	1000
Learning Rate	1e-4

A.2 Clustering Results for Other Intersections

We labeled DR_USA_Intersection_EP0 as EP0, DR_USA_Intersection_EP1 as EP1, DR_USA_Intersection_MA as MA, DR_USA_Roundabout_EP as EP, DR_DEU_Roundabout_OF as OF and DR_USA_Roundabout_SR as SR.

Table A.2: Intersecting rate for trajectory pairs in EP0 clusters

C	Num. Pairs Intersected	Total Num. Pairs	Intersecting Rate
0	50	227	0.22
1	131	187	**0.70**
2	25	40	**0.63**

Table A.3: Mean of minimum of SSMs over interactions within each EP0 clusters

C	TTC (s)	i v (m/s)	j v (m/s)	i long acc (m/s^2)	j long acc (m/s^2)
0	2.82	2.51	1.93	0.05	-0.3
1	3.3	0.46	2.09	**-1.37**	**-1.63**
2	3.16	3.59	6.64	0.24	**-1.52**

Table A.4: Intersecting rate for trajectory pairs in EP1 clusters

C	Num. Pairs Intersected	Total Num. Pairs	Intersecting Rate
0	54	241	0.22
1	109	140	**0.78**
2	42	82	**0.51**

Table A.5: Mean of minimum of SSMs over interactions within each EP1 clusters

C	TTC (s)	i v (m/s)	j v (m/s)	i long acc (m/s^2)	j long acc (m/s^2)
0	2.82	2.54	1.81	-0.19	-0.18
1	3.2	0.3	1.57	**-1.51**	**-1.63**
2	3.08	2.22	6.44	-0.09	**-1.43**

Table A.6: Intersecting rate for trajectory pairs in MA clusters

C	Num. Pairs Intersected	Total Num. Pairs	Intersecting Rate
0	350	653	**0.54**
1	699	957	**0.73**
2	36	186	0.19

Table A.7: Mean of minimum of SSMs over interactions within each MA clusters

C	TTC (s)	i v (m/s)	j v (m/s)	i long acc (m/s^2)	j long acc (m/s^2)
0	3.1	4.13	4.78	0.87	**-1.35**
1	3.05	0.21	1.48	**-1.13**	**-1.87**
2	2.94	1.02	3.5	-0.08	0.73

Table A.8: Intersecting rate for trajectory pairs in EP clusters

C	Num. Pairs Intersected	Total Num. Pairs	Intersecting Rate
0	131	263	0.50
1	168	212	**0.79**
2	111	282	0.39

Table A.9: Mean of minimum of SSMs over interactions within each EP clusters

C	TTC (s)	i v (m/s)	j v (m/s)	i long acc (m/s^2)	j long acc (m/s^2)
0	2.69	2.19	2.56	-0.26	-0.38
1	3.25	0.38	1.94	**-1.35**	**-1.64**
2	2.69	4.61	4.76	-0.15	-0.41

Table A.10: Intersecting rate for trajectory pairs in OF clusters

Table A.11: Mean of minimum of SSMs over interactions within each OF clusters

C	Num. Pairs Intersected	Total Num. Pairs	Intersecting Rate	TTC (s)	i v (m/s)	j v (m/s)	i long acc (m/s^2)	j long acc (m/s^2)
0	140	302	0.46	2.63	4.68	5.89	-0.14	-0.56
1	58	82	**0.71**	2.18	5.42	4.9	0.04	**-2.27**
2	53	60	**0.88**	3.13	0.69	2.25	**-1.33**	**-2.1**

Table A.12: Intersecting rate for trajectory pairs in SR clusters

Table A.13: Mean of minimum of SSMs over interactions within each SR clusters

C	Num. Pairs Intersected	Total Num. Pairs	Intersecting Rate	TTC (s)	i v (m/s)	j v (m/s)	i long acc (m/s^2)	j long acc (m/s^2)
0	109	143	**0.76**	3.14	4.96	3.43	-0.35	**-1.09**
1	188	212	**0.89**	3.23	0.35	1.71	**-1.24**	**-1.81**
2	51	122	0.42	3.25	0.4	6.24	-0.25	0.0

Appendix B

Traffic Conflict Validation by Probabilistic Time Series Prediction

B.1 Equation Derivation

Conditional action probability in Equation 3.11:

$$\begin{aligned} P(x \in X | u \in U) &= \frac{P(x \in X, u \in U)}{P(u \in U)} \\ &= \frac{\int_X \int_U p(x,u)\, dx\, du}{\int_U p(u)\, du} \\ &= \frac{\int_X \int_U p(x|u)p(u)\, dx\, du}{\int_U p(u)\, du} \end{aligned}$$

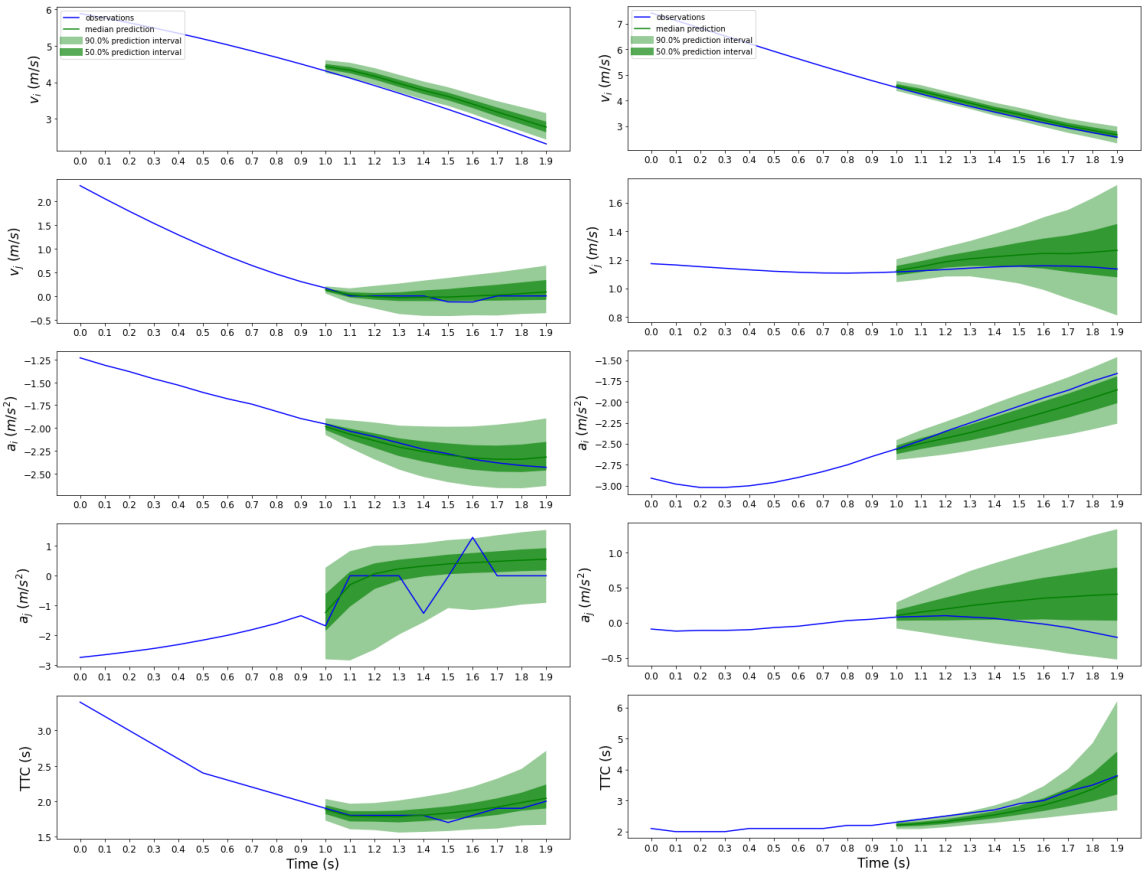
Crash probability in Equation 3.12:

$$\begin{aligned} P(y \leq 0, x \in R, u \in R) &= \int_{-\infty}^{0} \int_R \int_R p(y,x,u)\, dy\, dx\, du \\ &= \int_{-\infty}^{0} \int_R \int_R p(y|x,u)p(x,u)\, dy\, dx\, du \\ &= \int_{-\infty}^{0} \int_R \int_R p(y|x,u)p(x|u)p(u)\, dy\, dx\, du \end{aligned}$$

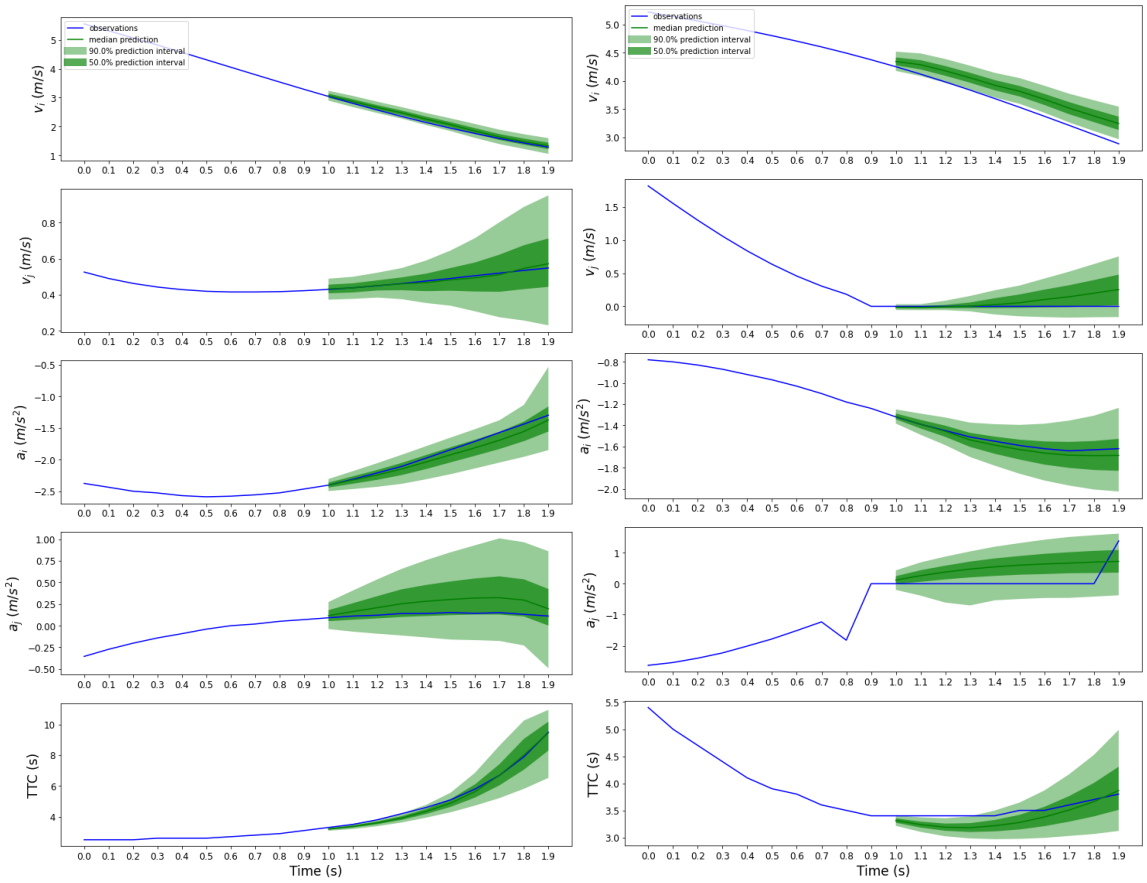
Conditional crash probability in Equation 3.14:

$$P(y \leq 0 | x \in X, u \in U) = \frac{P(y \leq 0, x \in X, u \in U)}{P(x \in X, u \in U)}$$
$$= \frac{\int_{-\infty}^{0} \int_X \int_U p(y|x,u)p(x|u)p(u)\, dy\, dx\, du}{\int_X \int_U p(x|u)p(u)\, dx\, du}$$

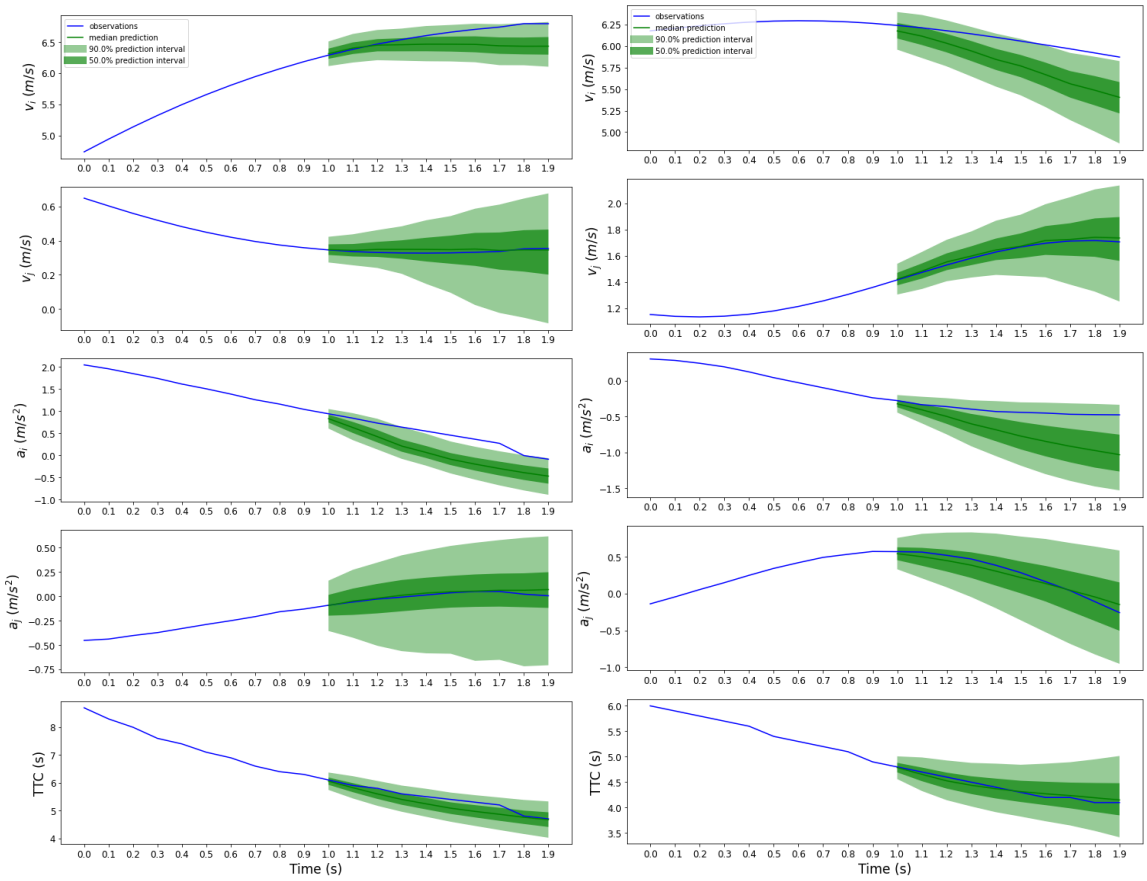
B.2 More SSM Probabilistic Prediction Plots



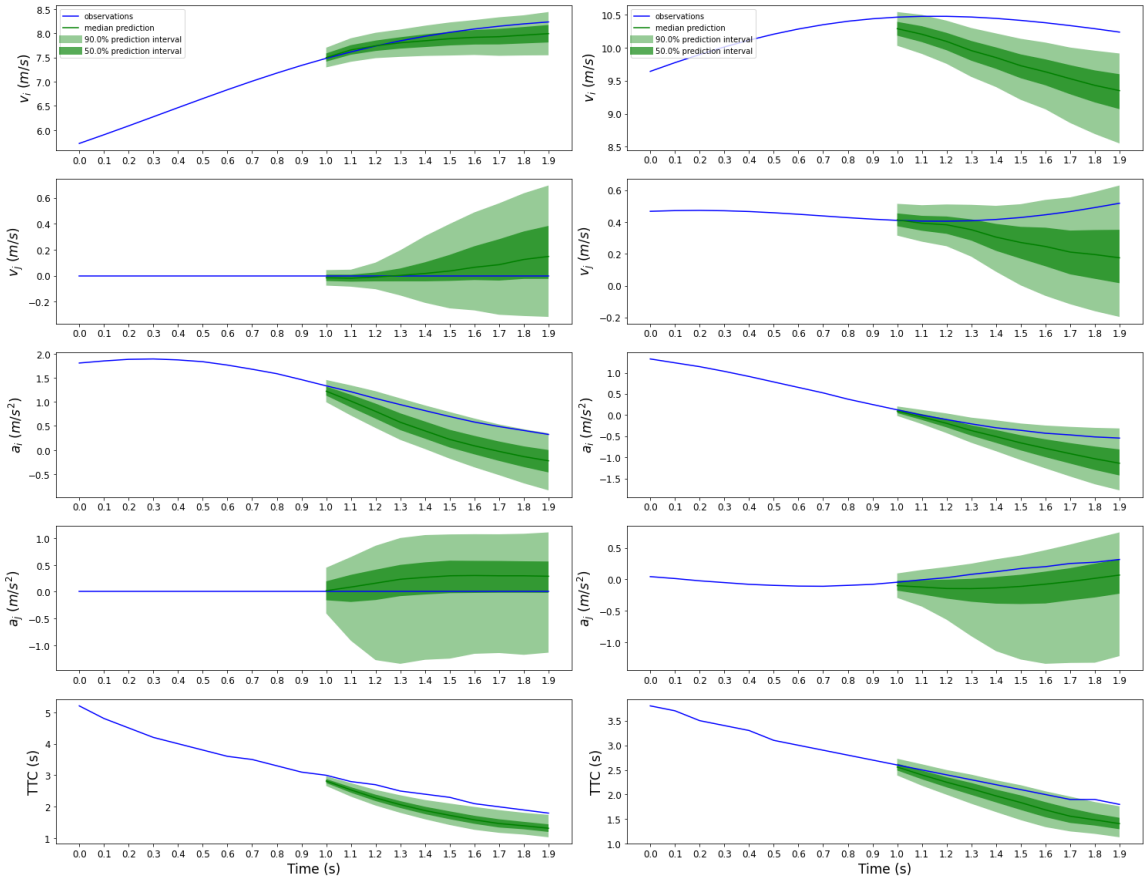
(a) (b)


observations
median prediction
90.0% prediction interval
50.0% prediction interval
v_i (m/s)
v_j (m/s)
a_i (m/s^2)
a_j (m/s^2)
TTC (s)
Time (s)

(c) (d)


observations
median prediction
90.0% prediction interval
50.0% prediction interval
v_i (m/s)
v_j (m/s)
a_i (m/s^2)
a_j (m/s^2)
TTC (s)
Time (s)

(e) (f)



(g) (h)

Figure B.1: More Examples of probabilistic SSM prediction

B.3 Values for the Estimated Probability

Table B.1 and B.2 are used to generate Figure 3.8.

Table B.1: Conditional Action Probability (%) under context k = traffic conflict at Figure 3.6a and $U = \{u \in U_{50\%}\}$

	1.0	1.1	1.2	1.3	1.4	1.5	1.6	1.7	1.8	1.9
Evasive Action	28.85	28.31	30.25	33.17	36.81	36.46	37.62	38.88	37.12	35.99
Large Deceleration	24.95	27.23	28.54	28.94	30.09	29.66	29.50	29.41	30.04	30.17
Small Deceleration	32.24	32.61	31.28	29.63	28.58	28.06	26.82	26.15	26.76	27.33
No Action	10.12	8.46	7.00	5.96	5.01	4.75	4.79	4.56	4.46	4.65
Acceleration	4.36	2.67	1.87	1.38	1.03	0.95	0.99	0.89	0.90	0.92

Table B.2: Conditional Action Probability (%) under context k = normal interaction at Figure 3.6b and $U = \{u \in U_{50\%}\}$

	1.0	1.1	1.2	1.3	1.4	1.5	1.6	1.7	1.8	1.9
Evasive Action	3.90	2.71	1.87	1.66	1.21	1.13	0.90	0.78	0.72	0.60
Large Deceleration	8.28	7.27	7.05	6.91	6.25	6.41	5.99	5.71	5.72	5.62
Small Deceleration	27.12	28.84	30.42	31.08	34.20	34.93	36.87	37.65	39.44	39.74
No Action	23.36	26.64	28.45	28.96	30.88	32.02	33.26	33.06	34.12	34.49
Acceleration	35.90	35.58	33.85	28.41	28.22	26.12	24.09	21.54	20.60	18.64

The first two rows of Table B.3 and B.4 are used to generate Figure 3.9.

Table B.3: Probability (%) for context k = traffic conflict at Figure 3.6a and $Y = \{y \leq 0\}, X = X_{\text{all}} = \{a_1 \in [-6, 6], a_2 \in [-6, 6]\}, U = \{U_{25\%}, U_{75\%}\}$

	1	1.1	1.2	1.3	1.4	1.5	1.6	1.7	1.8	1.9
$P(X\|U)$	103.3	100.3	97.70	102	100.9	96.63	97.13	101.8	102.7	103.7
$P(Y, X, U)$	4.06	4.39	4.60	5.43	5.45	5.07	5.34	5.90	4.72	6.51
$P(X, U)$	22.78	23.38	24.51	27.22	28.08	24.52	24.99	25.45	22.01	28.49
$P(Y\|X, U)$	17.80	18.76	18.77	19.93	19.40	20.70	21.36	23.17	21.46	22.86

Table B.4: Probability (%) for context k = normal interaction at Figure 3.6b and $Y = \{y \leq 0\}, X = X_{\text{all}} = \{a_1 \in [-6, 6], a_2 \in [-6, 6]\}, U = \{u \in [U_{25\%}, U_{75\%}\}]$

	1	1.1	1.2	1.3	1.4	1.5	1.6	1.7	1.8	1.9
$P(X\|U)$	100.3	97.45	100.4	99.56	100.8	99.30	101.1	100.5	98.02	99.59
$P(Y, X, U)$	3.19	3.55	3.33	3.16	3.33	2.73	3.08	3.22	2.64	2.39
$P(X, U)$	24.42	25.83	27.03	24.45	26.47	22.62	26.30	28.43	24.61	22.99
$P(Y\|X, U)$	13.06	13.75	12.30	12.94	12.59	12.06	11.70	11.32	10.73	10.39

The first and last rows of Table B.5 and B.6 are used to generate Figure 3.11.

Table B.5: Probability (%) for context k = traffic conflict at Figure 3.10a and $Y = \{y \leq 0\}, X = X_{\text{eva}} = \{a_1 \in [-6, -3], a_2 \in [-6, 6]\}, U = \{u \in U_{50\%}\}$

	1	1.1	1.2	1.3	1.4	1.5	1.6	1.7	1.8	1.9
$P(X\|U)$	38.78	38.15	34.94	31.49	27.17	22.84	18.40	14.67	11.71	7.47
$P(Y, X, U)$	2.42	1.97	1.69	1.62	1.08	0.74	0.51	0.35	0.26	0.18
$P(X, U)$	9.25	8.47	7.78	8.92	6.32	5.28	4.72	3.66	3.13	1.65
$P(Y\|X, U)$	26.15	23.21	21.71	18.15	17.05	14.02	10.79	9.45	8.16	10.71

Table B.6: Counterfactual probability (%) for context k = traffic conflict at Figure 3.10a and $Y = \{y \leq 0\}, X = X_{\text{no}} = \{a_1 \in [-0.5, 0.5], a_2 \in [-6, 6]\}, U = \{u \in U_{50\%}\}$

	1	1.1	1.2	1.3	1.4	1.5	1.6	1.7	1.8	1.9
$P(X\|U)$	6.64	8.27	8.23	8.92	10.10	10.89	12.91	12.83	13.16	13.87
$P(Y, X, U)$	0.34	0.34	0.48	0.67	0.86	0.85	1.08	1.28	0.97	1.43
$P(X, U)$	1.91	1.84	2.18	2.49	2.75	2.46	2.98	3.22	2.50	3.50
$P(Y\|X, U)$	17.63	18.43	22.07	26.96	31.18	34.60	36.25	39.72	39.03	40.81

Moreover, E_{TC} is the results of the last row of Table B.6 minus the last row of Table B.5 and E_{no} is the results of the last row of Table B.7 minus the last row of Table B.8. And Table B.9 generates Figure 3.12.

Table B.7: Probability (%) for context k = normal interaction at Figure 3.6b and $Y = \{y \leq 0\}, X = X_{\text{no}}\{a_1 \in [-0.5, -0.5], a_2 \in [-6, 6]\}, U = \{u \in U_{50\%}\}$

	1	1.1	1.2	1.3	1.4	1.5	1.6	1.7	1.8	1.9
$P(X\|U)$	23.36	26.64	28.45	28.96	30.88	32.02	33.26	33.06	34.12	34.49
$P(Y, X, U)$	0.66	0.78	0.80	1.01	1.15	0.99	0.93	0.87	0.87	1.03
$P(X, U)$	5.31	6.24	6.77	7.96	8.92	8.43	7.79	7.67	8.24	9.95
$P(Y\|X, U)$	12.35	12.50	11.74	12.66	12.89	11.80	11.90	11.30	10.54	10.32

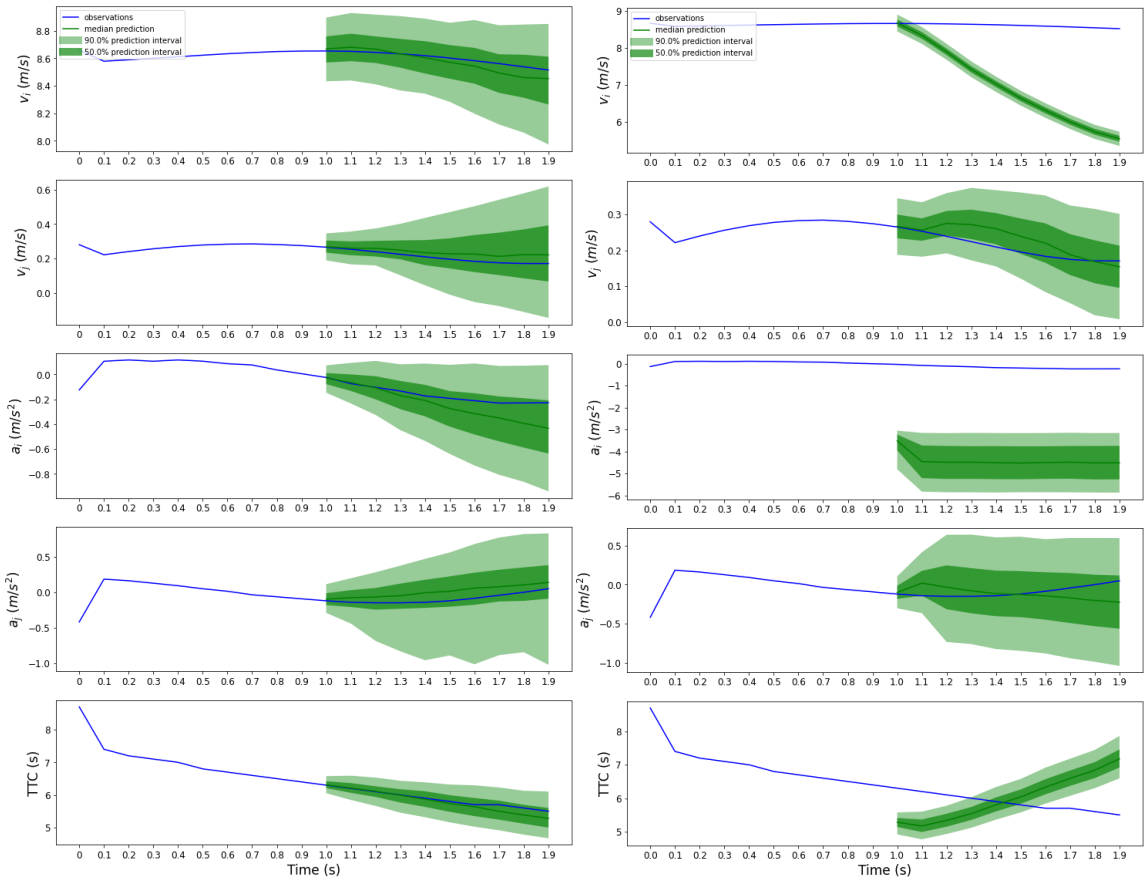
Table B.8: Counterfactual probability (%) for context k = normal interaction at Figure 3.6b and $Y = \{y \leq 0\}, X = X_{\text{eva}} = \{a_1 \in [-6, -3], a_2 \in [-6, 6]\}, U = \{u \in U_{50\%}\}$

	1	1.1	1.2	1.3	1.4	1.5	1.6	1.7	1.8	1.9
$P(X\|U)$	3.76	23.98	12.36	11.90	11.94	11.62	11.48	10.38	10.00	9.63
$P(Y, X, U)$	0.12	0.85	0.48	0.57	0.51	0.57	0.53	0.50	0.61	0.54
$P(X, U)$	0.90	5.91	3.28	3.34	2.83	3.03	2.60	2.35	2.82	2.39
$P(Y\|X, U)$	13.82	14.33	14.77	17.17	18.04	18.71	20.46	21.19	21.61	22.39

Table B.9: Effectiveness of evasive action under traffic conflict and normal interaction contexts

	1	1.1	1.2	1.3	1.4	1.5	1.6	1.7	1.8	1.9
E_{TC}	3.76	23.98	12.36	11.90	11.94	11.62	11.48	10.38	10.00	9.63
E_{no}	0.12	0.85	0.48	0.57	0.51	0.57	0.53	0.50	0.61	0.54

Additionally, the counterfactual plot for normal interaction is shown in Figure B.2b.



(a) Correct prediction: driver is taking no action

(b) Counterfactual prediction: if the driver had taken evasive action

Figure B.2: Correct prediction vs. counterfactual prediction under normal interaction context