LSTM Model and Data Segmentation Collaboration

A Deep Learning Strategy for Traffic Flow Anomaly Detection Using SUMO Simulated Data

Lai Hu
School Of Computing and Physical Science
University Of Leeds
Leeds, United Kingdom
sc22l2h@leeds.ac.uk

***Abstract*—To address the escalating challenge of traffic congestion and the associated difficulties in obtaining specific congestion data sets, this paper explores the use of deep learning strategies to determine traffic flow conditions using traffic flow data generated by urban traffic simulation software (SUMO). This paper describes an approach that combines LSTM and data segmentation to convert SUMO data into a general model format. This conversion is compatible with LSTM-based models originally trained on SUMO traffic flows. The paper confirms through rigorous quantitative analyses that the use of SUMO-generated traffic flows in the field of traffic flow prediction has the advantages of data purity, ease of collection and ubiquity over the use of real traffic flows and verifies that there can be the same correspondence mapping between the real-world dataset and the SUMO dataset. We have also investigated the robustness and generality of the model, demonstrating its ability to adapt to different traffic scenarios, and its ability to identify anomalous traffic flows by 89.5% under experimental conditions, which is 8.1% higher than the model based on the NGSIM dataset (81.4%).This work highlights the promise of using simulation software to generate datasets, combined with deep learning approaches, to provide practical and scalable solutions to support data-driven traffic management decisions. In general, this paper propose the use of simulation software with data segmentation mapping to provide a solution for the research and analysis of regional traffic in circumstances where unavailability and realistic pristine datasets are not obtainable.**

Keywords-component; Deep learning, Data processing, LSTM, SUMO software, Synthetic dataset creation, Data Mapping.

I. INTRODUCTION

Rapid changes in traffic flow have become a pervasive problem in today's urban landscapes, resulting in longer travel times, poorer air quality and reduced quality of life. Despite significant advances in infrastructure planning and traffic management systems, urban areas still struggle with chronic traffic congestion [1]. The increasing complexity of urban transport systems, including dynamic variables such as traffic volume, route choice, pedestrian movements and traffic light synchronization, is central to this problem. It is therefore necessary to monitor traffic flows in order to provide decision support for related problems.

In 2020, an AI-based traffic control system has been implemented to reduce the burden on human operators responsible for monitoring traffic events. This one particularly striking example shows that AI systems have gradually been deployed in practice in recent years and demonstrates the potential of using AI systems to significantly reduce the workload of these operators [2].

The use of AI in traffic detection is highly relevant as a tool to assist managers in minimizing the need for human resources and improving the ability of managers to deal with traffic.

However, a common limitation of many studies is the reliance on realistic data, and the simulation results obtained are often not applicable to real-world scenarios. The aim of this research is therefore to use AI to determine whether traffic flow is normal or not, and its viability as a platform for training deep learning models to predict traffic flow status. The bionic AI algorithm will use data inputs to determine the operational state of a traffic flow. The paper will use different datasets to compare the effectiveness of these models in determining real-world traffic flow conditions.

To achieve this, we plan to use Long Short Term Memory Networks (LSTM), a class of AI algorithms that learn to make decisions by interacting with their environment. LSTM is a special type of recurrent neural network (RNN) that captures long-term dependencies in time-series data. This could be very useful for traffic flow prediction, as the state of traffic flow can be influenced by past states [3].

A major challenge in training traffic models for most practical applications is obtaining pure training data sets. It is often difficult to obtain a dataset that meets real-world requirements. The ideas presented in this paper, if validated, will also provide a viable approach to providing the necessary datasets.

The expected outcome of this paper is an artificial intelligence system capable of effectively detecting traffic flow anomalies through the analysis of road traffic data. If successful, this AI and the ideas presented in this paper will provide a new source of traffic flow data training and a method of data generation for related research.

II. REALATED WORK

This section will review the relevant literature in the field of traffic flow prediction and anomaly detection. This includes studies that have utilized traditional statistical methods, machine learning techniques, and deep learning models. The limitations of these studies and how the proposed approach addresses these limitations will also be discussed.

A. Traffic Flow Prediction Studies

Traffic simulation is an important tool for studying and managing urban traffic. It uses virtual reality to reconstruct road traffic and simulate complex processes involving drivers, vehicles, roads and the traffic environment. The increasing number of vehicles in urban environments poses new challenges for traffic management, making traffic analysis and modelling relevant and important for intelligent traffic management.

When trying to use real-world traffic datasets, one usually needs to spend a lot of time processing the dataset, even at the national level, the error that occurs in processing the data can only be controlled within 15% of the total data, and this leads to a lot of money consumption and collection time spent in the United Kingdom[4], the standard traffic sampling length unit for Northern Ireland is feet, whereas in England it is meters and in Scotland it varies according to policy[5].

Traditionally, traffic control research has used two main approaches to selecting datasets: mixing software datasets with real-world datasets, or directly using processed real-world datasets. When researchers justify traffic prediction scenarios, they process real-world extracted datasets or combine them with self-constructed software-generated datasets to the topic [6]. Despite the high quality of these carefully processed datasets, the use of these datasets by others in the field requires additional processing or format correction, resulting in many wasted research processes.

However, at the same time, research indicates that the problem of congestion in urban traffic involves the flow of data in addition to planning problems and management problems. In highways, the government and related institutions can easily get the data of traffic flows but in cities and countryside, the result is controversial [7].

There are two main types of traffic flow conditions: normal and abnormal. Abnormal traffic flow is usually caused by excess vehicles, accidents, bad weather and special events. Non-recurring traffic events account for about 50% of all traffic congestion, and the accurate prediction of non-recurring traffic events is a challenging area that requires more research to analyze[8].The features present in the traffic flow are very easy to extract, so it can be considered as a binary classification problem, i.e. whether the traffic flow state is normal or abnormal, which is a more applicable topic for AI to determine.

This actually means a lot of traffic data is not systematically analyzed, uploaded and fed back to the projects, and no improvements can be made to the existing congestion points since they are not being noticed.

B. Deep Learning in Traffic Management

Deep learning has been increasingly applied to the detection, prediction and mitigation of traffic congestion, with the aim of improving the level of service of transport networks. The relevance of deep learning to these tasks has become apparent with the growth of high-resolution and larger datasets. Traffic congestion affects the level of service (LOS) of road networks, which in turn generates direct and indirect social costs. For example, in the US alone, approximately 8.8 billion hours of work are lost each year due to traffic congestion. In addition, congestion encourages aggressive driver behavior, increases the likelihood of accidents, and leads to higher greenhouse gas emissions [9].

In terms of forecasting, short-term traffic predictions are usually made seconds to hours in advance using historical and current traffic information. For short-term traffic flow prediction, even in the case of missing data sets, traditional algorithms are still statistically used to make correct predictions in the short term, as mentioned in Offor et al [10]. using optimized traditional algorithms was able to improve the accuracy of short-term traffic flow prediction by 23% to 36.34% compared to AI models trained on incomplete data, however, in the case of complete data, the algorithms still perform worse than the AI models used for comparison in short-term predictions.

While the exact boundary between short-term and long-term forecasting is unclear, and it is difficult to be optimistic about the long-term effectiveness of artificial algorithms given their input limitations, it is clear that intelligent, reliable methods of traffic forecasting are essential for reducing congestion. Despite being highly sensitive to training datasets, AI models are still widely used for traffic prediction, for example Amazon's AWS model is widely used for traffic prediction and race prediction (e.g., F1, MotoGP and other highly diverse competitive sports).

Improved traffic data collection infrastructure and increased computational resources have facilitated the use of deep learning for congestion detection, prediction and mitigation [7]. In supervised learning tasks using deep learning, such as congestion prediction, the goal is to train a deep learning model that learns a mapping from input data to output data. Several techniques have been explored to better generalize deep learning models, including dropout and other methods that exploit knowledge of the traffic domain [9]. There are also those who argue that AI trained on neural networks is feasible in this context.

It is well known that traffic data varies over space and time, and two families of neural network architectures - convolutional neural networks (CNNs) and recurrent neural networks (RNNs) - have been shown to be particularly effective at capturing this interdependence. CNNs have been widely used for image classification problems due to their ability to capture correlations between nearby pixels in an image. When applied to traffic data, CNN models can capture spatial dependencies, making them suitable for tasks such as predicting congestion levels based on two-dimensional images of traffic, while RNNs, particularly long short-term memory (LSTM) networks, can capture dependencies between data points that are far apart in time [11].

C. Use of SUMO in Traffic Studies

Traffic simulation has been a long-standing academic focus in the field of urban intelligent transport, and many theoretical and technological innovations and applications have emerged over the past century. These include advanced traffic signal control systems such as SCOOT and SCATS, and a variety of excellent traffic control algorithms such as model-driven algorithms, data-driven algorithms and artificial intelligence-based algorithms [12]. These are powerful tools for the rapid development of urban traffic control, but these algorithms require the support of a large number of data sets.

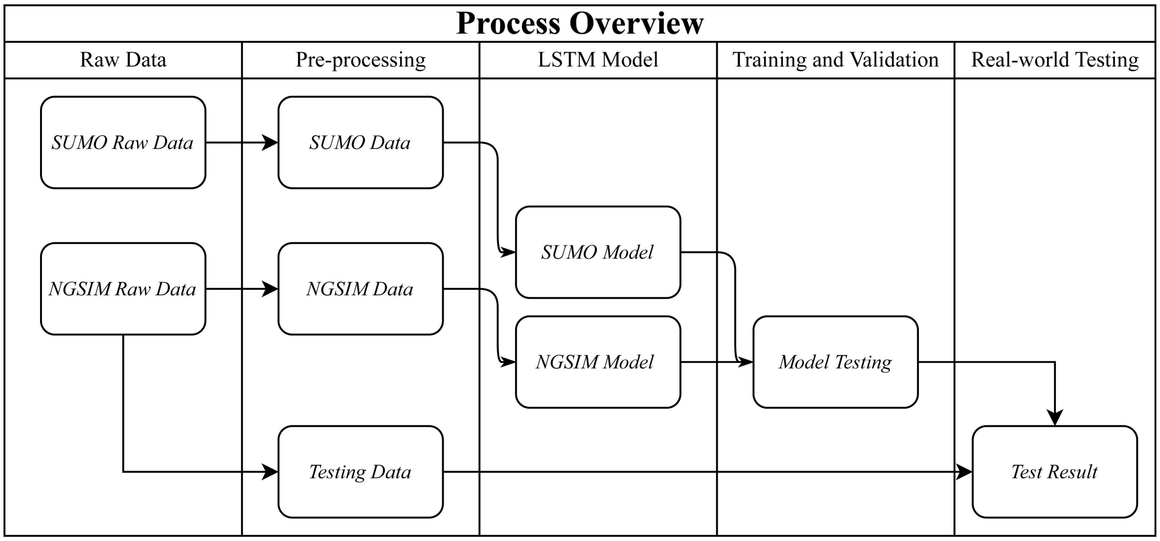

Figure 19. Process Overview

The research conducted by Ma et al. has illuminated the substantial role that the SUMO simulator plays within the domain of traffic flow studies [13]. Their paper provides robust evidence supporting the reliability of SUMO. Despite the enormous sample size of 683,500 instances, the predictions generated from traffic flow data simulated by SUMO retained a high degree of accuracy. This finding underscores the robustness of SUMO in generating and simulating traffic flows. It suggests that for traffic flow studies, SUMO is not merely a tool that provides convenience for researchers, but more importantly, the traffic flow data it generates can manage large-scale samples while maintaining a high level of precision. Consequently, this further validates the importance and reliability of SUMO within traffic flow studies, offering a powerful tool for researchers in related fields.

III. Overview

Our work covers the following two parts.

- Develop a deep learning model that uses traffic flow data generated by the SUMO simulator to accurately predict the operational state of traffic flows.
- Validate the predictive capabilities of the deep learning models trained on both SUMO simulated data and real-world data by assessing their performance in predicting the state of real traffic flows.

In this section, we first introduce the relevant techniques involved in this work and then describe the structure of model, whereas the evaluation and measurements of results will be listed at the end of this section.

A. Techniques

*1) SUMO (**S**imulation of **U**rban **Mo**bility):* SUMO is an open-source, highly portable, microscopic, and continuous road traffic simulation package designed to handle large road networks. It is capable of intermodal simulations including road vehicles, public transport, and pedestrians. SUMO's comprehensive Python API, which allows for external control of the simulation, is leveraged in this paper.

*2) LSTM (**L**ong **S**hort-**T**erm **M**emory):* Long Short-Term Memory (LSTM) is a type of recurrent neural network (RNN) capable of learning and remembering over long sequences of data. Unlike traditional feedforward neural networks, LSTM has feedback connections, making it a "general purpose computer". It is not only capable of processing single data points, but also entire sequences of data, making it suitable for time-series prediction tasks, such as traffic flow prediction. Therefore, as the research does not involve reading traffic images, LSTM models will be used to build artificial intelligence and compare research results, as there is a need for correlation over time slices and there are dependencies between data in learning.

B. Structure of Overview

The workflow of the project is listed in figure 19.

At the start of the process, we will prepare and pre-process two sets of data. This is to split them into the format used by the model. The processed data from these two datasets will be formatted in the same way and split into three separate datasets. These datasets will be used to train the two models separately and validate them in a loop on the final dataset.

During the training and validation phases, we will use independent data to train the models; the model trained on the

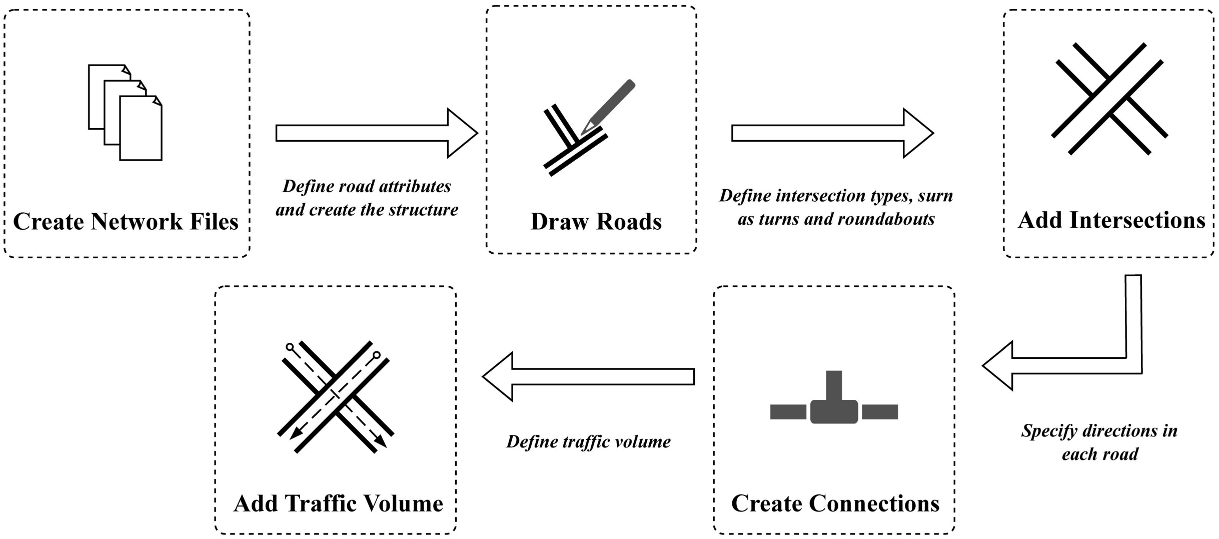
Create Network Files
Define road attributes and create the structure
Draw Roads
Define intersection types, surn as turns and roundabouts
Add Intersections
Specify directions in each road
Create Connections
Define traffic volume
Add Traffic Volume

Figure 20. Data Segmentation Process

SUMO data will have no access at all to the actual test dataset during both the training and validation phases, and the model trained on the NGSIM data will also have no direct access to the relevant test dataset. This ensures the comparability and accuracy of the final results.

This process can be seen as consisting of three phases: data preparation, model training and model testing. In the data preparation phase, we split the original data set into a training set, a validation set and a test set. In the model training phase, we use the training and validation sets to train and optimize our model. In the model testing phase, we use the test set to evaluate the performance of the model.

In this process, the data used in each step is independent and does not interfere with each other. In this way, we can gradually improve the accuracy and robustness of the model, while eliminating the interference caused by the problem of inaccurate data sets.

C. *Model Architecture*

As Shown in figure 18, we use a bidirectional long short-term memory (Bi-LSTM) model, which includes two LSTM layers and a dense (fully connected) layer. The bidirectional LSTM model facilitates feature extraction from sequential data by simultaneously considering forward and backward temporal dependencies [14].

Each LSTM layer contains 50 units, with a 'return_sequences' parameter set to True to pass the output sequence to the next layer. To mitigate overfitting, the model uses L2 regularization with a coefficient of 0.0001, and each dropout layer omits 20% of the input. The dense layer produces a singular output for the binary classification task, using a sigmoid activation function to constrain the output to the range 0 to 1, indicating the binary classification probability. An RMSprop optimizer is used with a learning rate of 0.001. The model uses a binary cross-entropy loss function, which is suitable for binary classification tasks. In addition to the loss function, the model's performance metrics also calculate accuracy during training and evaluation.

The model also includes a monitor that specifies the indicators to watch, which helps to determine when to stop training. In this case, the training status is tracked based on the loss ('val_loss') on the validation set.

D. *Evaluation and Measurement of results*

The core objective of this paper is to use models trained on traffic flow data generated by the SUMO software as a basis for judging realistic anomalous traffic flows, therefore the measurement factors depend on the two objectives mentioned above, which will be achieved experimentally, therefore parameters such as confusion matrices, which measure the effectiveness of the deep model, will be used in conjunction to

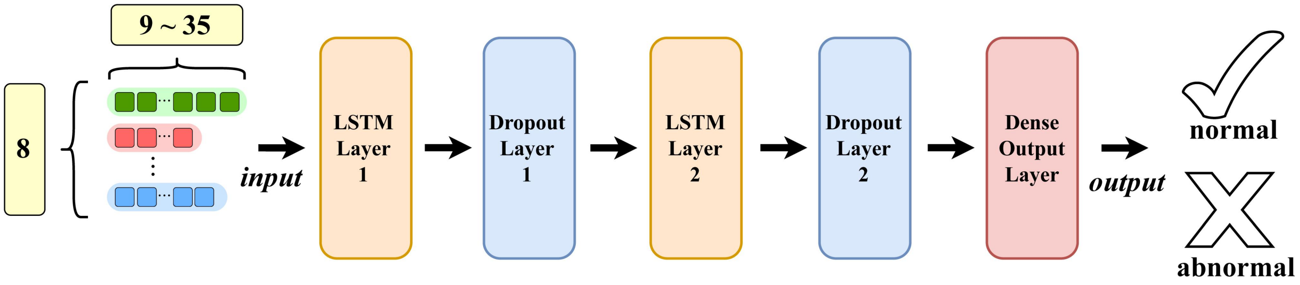

Figure 18. LTSM Model Design

measure whether the objectives have been achieved as well as the overall efficiency of the model.

The evaluation phase will present the results of the model trained on the real dataset and the model trained using SUMO, along with comparisons and discussions based on the results and related data.

By analyzing the results, It is reasonable to compare the performance of the model and the advantage or disadvantage of different dataset.

IV. METHODOLOGY

A. Method Overview

This section presents the training and validation results of the model and the process of the total workflow.

In order to fully assess the validity of the model and the criticality of the paper, model was trained on both simulated and real datasets and demonstrate its training results.

B. Data Segmentation Process

In this paper, Long Short-Term Memory (LSTM), a deep learning model extensively employed for time series prediction, is utilized as a fundamental component of the artificial intelligence algorithm. The LSTM model is selected as the base model due to its proven efficacy in a multitude of time series prediction tasks [15].

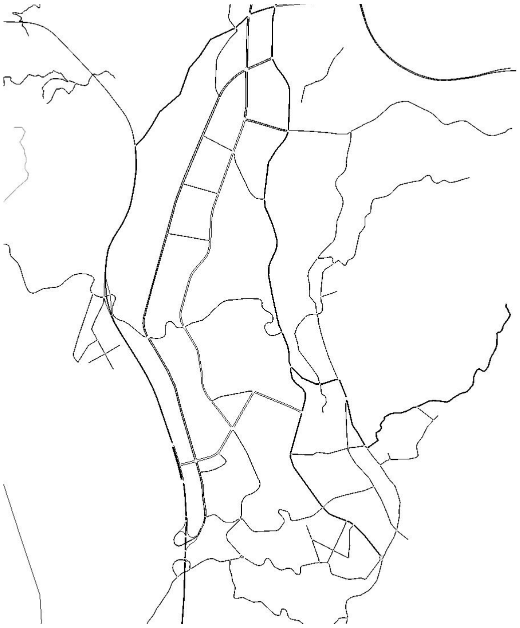

Figure 1. Simulation map.

The selection of LSTM offers the advantage of effectively capturing long-term dependencies in time series data. Given that the task at hand involves traffic flow prediction, this advantage facilitates the model in maintaining superior performance when learning deeper, more abstract features. Moreover, the robust performance of the LSTM model across various time series forecasting tasks substantiates its selection as the base model for this paper.

As shown in figure 20, during the data pre-processing phase, several processes are applied to the training data. These processes not only augment the diversity of the training data but also enhance the robustness of the model. This enables the model to generalize and make improved predictions when confronted with a range of complex scenarios in the real world. Throughout the model training process, the model is trained and tuned with different datasets, and fine-tuning strategies are implemented.

The benefit of this approach is that the model performance can be rapidly brought to a relatively reasonable state, leading to recommendations on how to train the model on a large dataset, and subsequently fine-tuning the dataset or the model as a whole. Such a two-step training strategy can further enhance the performance of the model while maintaining the speed of training.

C. SUMO Raw Data Generating and Processing

The simulated area for this paper is a matrix range of urban traffic lanes on GPS: from longitude 105.5570 (left boundary) to longitude 105.6188 (right boundary) and from latitude 27.2211 (upper boundary) to dimension 27.1264 (lower boundary). By using the Netedit that comes with Sumo, after planning the plotting we obtained the following image, and it largely matches the real-life satellite image [16].

On this map, the paths out of the city are randomly set where the vehicles are generated and where they leave, while all major traffic points are retained, eliminating the rail system and the bypass motorway system to achieve maximum simulation.

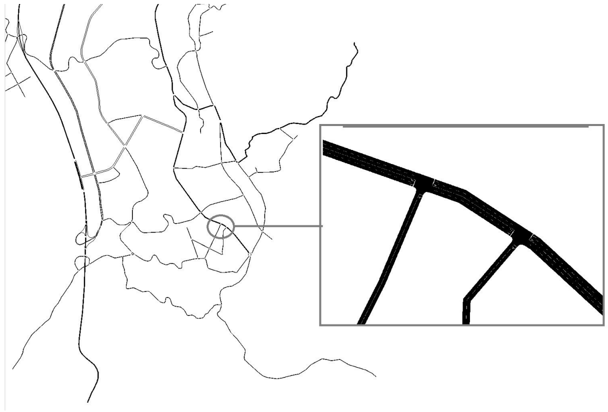

Figure 2. Simulation Map Overview.

D. Traffic Flow Generating

The traffic flow itself consists only of the travel time, type, speed and route of the vehicle, but is influenced at a macro level by a number of factors, such as the traffic light settings on the road network, the length of the road configuration and the routes of other vehicles.

The sample of a part of traffic flow was listed below, and the Figure below also indicated a typical crossroad in SUMO.

In this paper, different traffic flow simulations were planned for the city to train and test AI models to achieve accurate predictions of future traffic flows. We will collect and analyze various traffic data to improve the accuracy and efficiency of our prediction models.

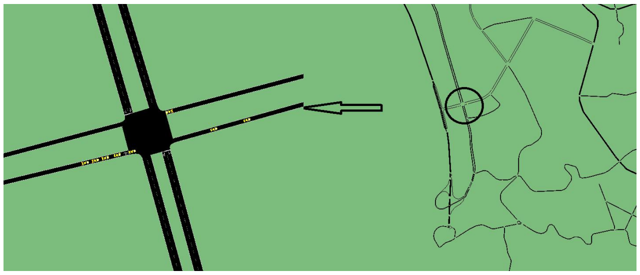

Figure 3. Traffic Flow In Sumo.

When the simulation is complete, the detectors generate a separate file that counts the vehicles that have passed during each manually set detection period, and by comparing this data we can work out which sections of the road are congested.

E. Data Pre-Processing and Overview

The reason for this is that irregular traffic flows in the city are mainly caused by inconvenient traffic at junctions (Russo et al, 2023), so by detecting traffic flows at junctions it is possible to effectively detect traffic flows at a macro level across the city and it means that by detecting traffic flow at junctions.

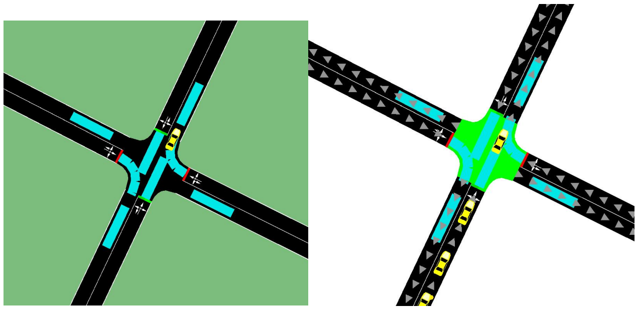

Figure 4. Sensor in Crossroads.

With a total of 182 sensors, a single sensor detection period of 45 time units (SUMO units/second) and a length of 100 (SUMO units/meter), the summary statistics of the overall sensor layout are as follows, and the sensor layout in the intersections is summarized below, with a total of 182 sensors, a single sensor detection period of 45 time units (SUMO units/second) and a length of 100 (SUMO units/meter), the bar chart below shows the distribution of the number of sensors at each of the 34 junctions involved in the road network test at that time. It should be noted that in the final generated dataset the traffic flow is completely random and only the roads remain fixed, so the final generated training traffic flow data is still purely random.

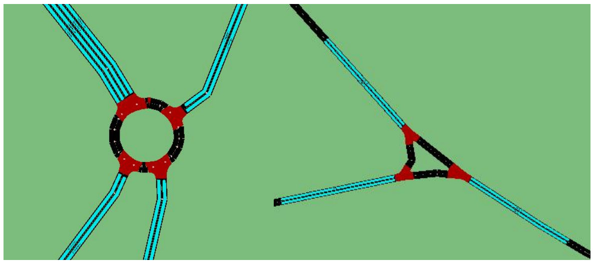

Figure 5. Intersection Sensors Example.

F. SUMO Sensor Data Collecting

During processing, the processor iterates over each sensor file. For each file, a function is instantiated that is designed to manage the data from a single sensor.

For each time interval, the traffic flow index C is calculated. We then generate a distribution plot of the C values, arrange the C values in order, and generate a line graph of the C values. This graph also includes a baseline, which is the mean of all the C values.

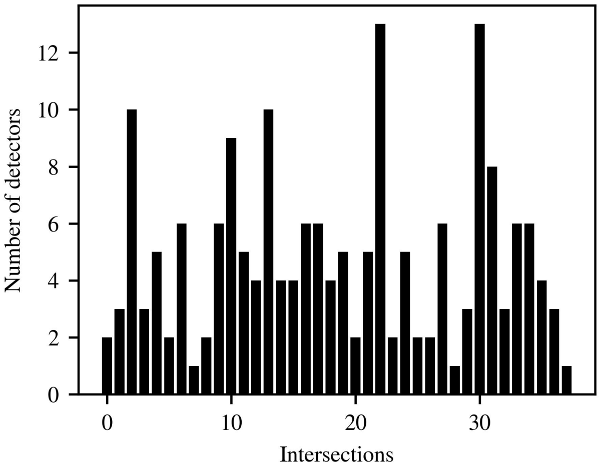

Figure 6. Sensors in Different Intersections.

Each time interval is given a label depending on whether its C value exceeds the baseline. If the C value exceeds the baseline, the label is 1, indicating an abnormal traffic flow. If the C-value is less than or equal to the baseline, the label is 0, indicating normal traffic flow. The data and labels are then split into a training set and a test set using the train_test_split function from the sklearn.model_selection library, setting the size of the test set to 20% of the total data and defining a random seed.

The data is reshaped to meet the input shape requirements of the LSTM model: (samples, time steps, features). In the original data, each sample could represent a time interval, each time interval could have one time step, and each time step could have multiple features (e.g. number of vehicles, average speed, etc.). Based on the research on LSTM, When using a bi-directional LSTM network structure, the use of stable and smooth fixed inputs is required to prevent uncontrollable bias judgements in the neural network[17].

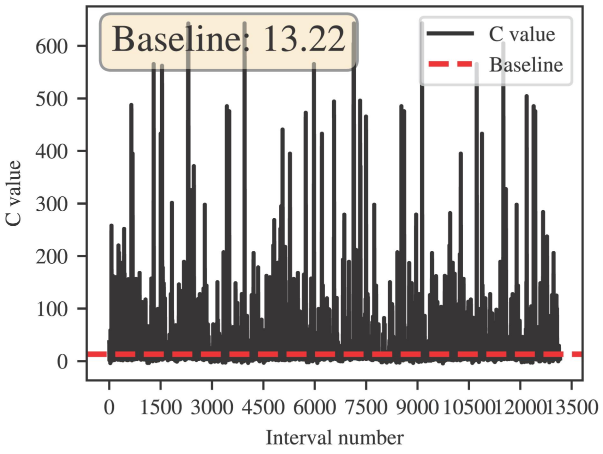

Figure 7. Model Baseline On 11000 Rows SUMO Dataset

G. NGSIM Raw Data Processing

This paper uses the Next Generation Simulation (NGSIM) dataset to analyze vehicular traffic flow, employing a ten-step computational pipeline to transform raw data into a format suitable for quantitative analysis.

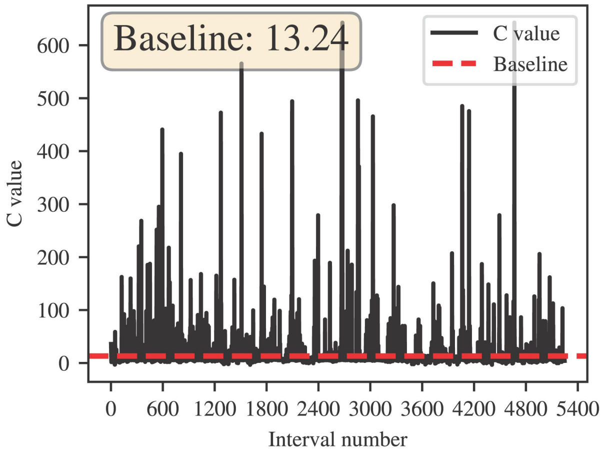

Figure 8. Baseline For Model on NGSIM Real World Dataset

First, road segments are identified based on the vehicle's global position, calculated through Euclidean distance from the origin. Next, the maximum speed for each road segment is determined. The third step involves calculating the speed loss and time loss for every vehicle at each time point.

The 'Global_Time' is then normalized and translated into 45-second intervals. Subsequently, the change in 'Local_Y' for each vehicle at each time point is computed, creating two binary fields 'enter' and 'leave'.

Duplicate records are eliminated, preserving only the first record for each vehicle in each 45-second interval. The data is then grouped by 'Section_ID' and 'Global_Time', and relevant metrics are calculated for each group, resulting in a summarized dataset.

Finally, the 'begin' and 'end' times are converted from milliseconds to seconds.

H. Quantification Formula C

As discussed in the previous subsection, this paper has used a calculation to measure the weight of the traffic flow.

TABLE I. DATA ITEM FORMS MODEL

Original Data	Effect
meanSpeed	Average speed In a period
meanTimeLoss	Average loss of time In a period
nVehLeft	Vehicles left sensor area in a period
nVehsSeen	Vehicles appears in sensor area in a period
nVehEntered	Vehicles entered sensor area in a period
End-Begin	The length of current period

We proposed the *equation C*, which is able to assess the normality of the traffic flow according to different traffic situations.

However, LSTM networks need to prevent the model from diverging from the attention mechanism[18].Using the proposed C as data baseline, it is possible to pay sufficient attention to the key relevant information without making changes to the attention mechanism on the LSTM side of the model, allowing the model to focus on learning more important data features and further improving prediction performance.

Based on the original data, we can define a new parameter D, which represents the difference between entering and exiting vehicles:

① $$D = nVehEntered - nVehLeft$$

This parameter can be used to measure the change in traffic flow. If D is positive (more entering vehicles than exiting vehicles), the traffic flow may become abnormal; if D is negative (more exiting vehicles than entering vehicles), the traffic flow may become normal.

We can then define a new parameter V, which is the ratio of vehicles seen to vehicles entering and leaving:

② $$V = nVehSeen/(nVehEntered + nVehLeft)$$

This parameter can be used to measure the visibility of the traffic flow. If V is high (more vehicles seen), the traffic flow may be abnormal; if V is low (fewer vehicles seen), the traffic flow may be normal.

Finally, we can define a new parameter ST, which is the ratio of speed to time loss:

③ $$ST = meanSpeed/meanTimeLoss$$

This parameter can be used to measure the efficiency of the traffic flow. If ST is high (high speed and low time loss) then the traffic flow is probably normal; if ST is low (low speed and high time loss) then the traffic flow is probably not normal.

We can then combine these parameters to define a new equation:

$$④ \quad C = w1*D + w2*V + w3*ST + w4*(0.01*B)$$

$$⑤ \quad [w_1, w_2, w_3, w_4] = [0.6, 0.6, 0.2, 1]$$

The B that appears here is the bias term to smooth the curve as well as the C value.However, the absence of vehicles during a sensor cycle can affect the calculation of the C-value, especially when calculating average speed and average time loss. The formula will also filter out data where there are no vehicles in a sensor cycle before calculating the C-value. You can add a condition to the process_sensor_data function to only add data to the list if at least one of nVehEntered, nVehLeft and nVehSeen is greater than 0. This ensures that the C-value calculation is based on data with vehicle flow.

It is noticed that due to the data quality, the true baseline for different dataset applys the following formula:

$$⑥ \quad true\_baseline = aveC - 13.22(varies) + minC$$

I. *Model On SUMO Dataset*

At the initial stage, the training set contains 9210 normal, 1319 abnormal samples. After the oversampling technique was applied, the data changed as 9210 normal, 9210 abnormal whereas the validation set kept the same 2316 normal, 317 abnormal.

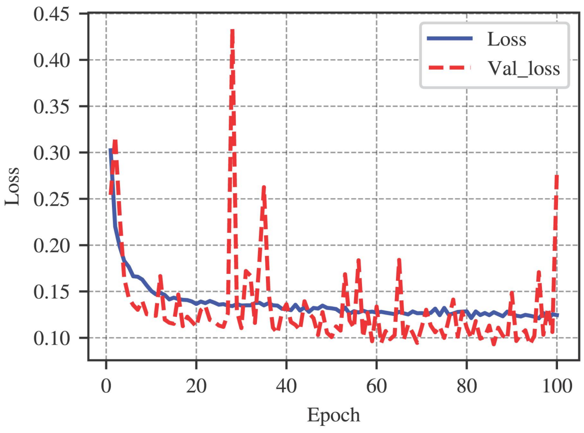

Figure 9. Model Loss for Oversampled 11000 Dataset Model

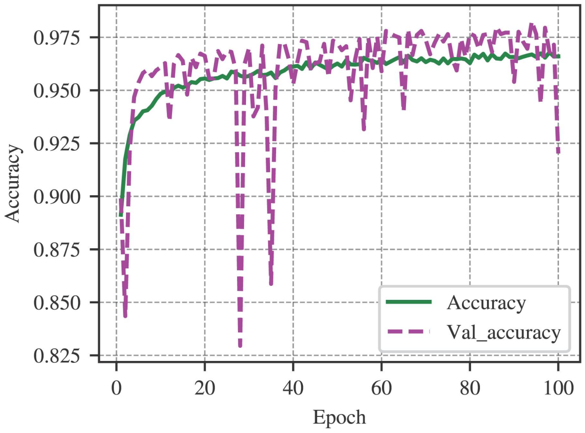

Figure 10. Model Accuracy for Oversampled 11000 Dataset Model.

The oversampling technique has been applied to the training data to address the issue of class imbalance, where the 'Abnormal' class was underrepresented. The results show a significant improvement in the model's performance on the minority class.

TABLE II. CLASSIFICATION REPORT FOR FINAL MODEL ON OS 11000 MODEL

Overall Accuracy				0.98
	Precision	Recall	F1-Score	Support
0	0.99	0.99	0.99	2316
1	0.93	0.94	0.94	317
Macro Avg	0.96	0.96	0.96	2633
Weighted Avg	0.98	0.98	0.98	2633

After oversampling, the precision and recall for the 'Abnormal' class improved to 0.93 and 0.94 respectively. This demonstrates that the model was able to make more accurate predictions (higher precision) and also correctly identify a higher proportion of actual 'Abnormal' instances (higher recall).

The overall accuracy is 0.98, indicating that the model was able to correctly classify a higher proportion of instances. The macro average of precision and recall, which gives equal weight to both classes, improved to 0.96, indicating that the model's performance improved equally for both classes.

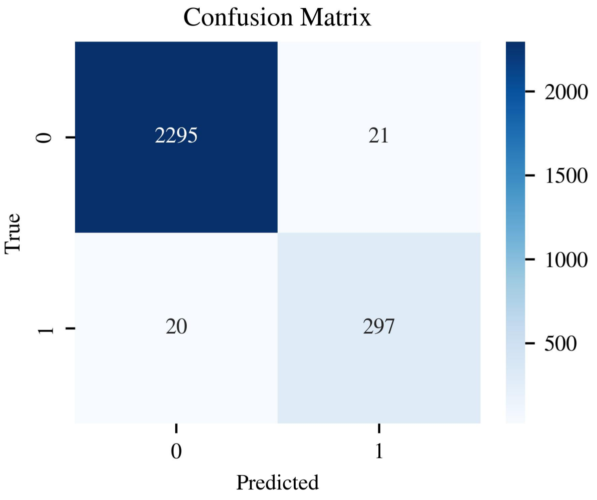

Figure 11. Confusion Matrix for Final Model on OS 11000 Model.

J. *Model On Real-World Dataset(NGSIM)*

The model is derived from the data set partitioned in the data processing stage, and the training set includes 521 normal, 407 abnormal samples and 388 normal, 95 abnormal in validation set.

It can be seen from these results that model A has a better prediction performance for category 0 but a slightly poorer performance in predicting category 1.

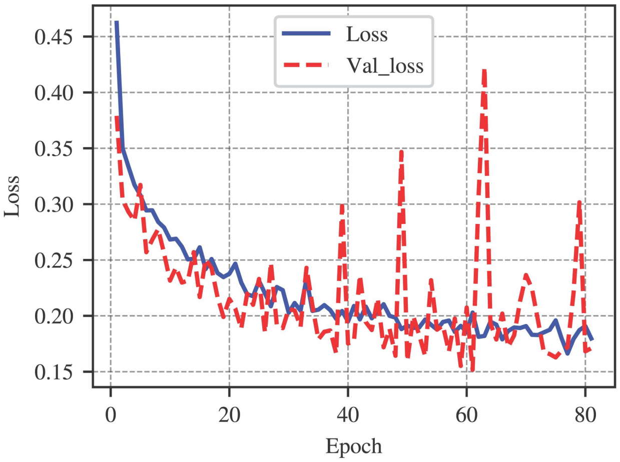

Figure 12. Model Loss on NGSIM Training Dataset

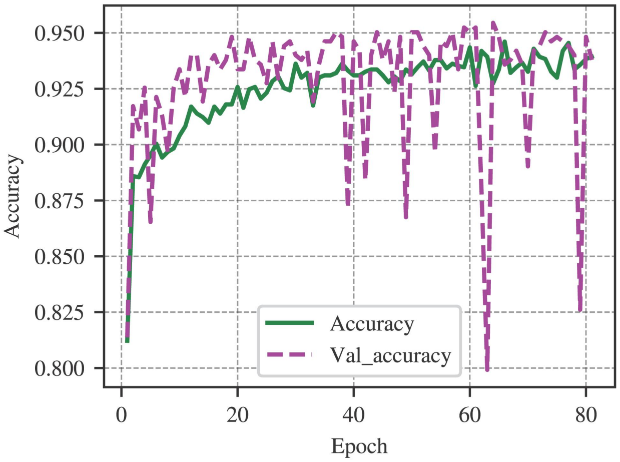

Figure 13. Model Accuracy on NGSIM Training Dataset

From the training process of the model, the model based on the real dataset has a high recognition of the validation set at the beginning of the training and the performance of the model Loss and Accuracy becomes better with iterations, but there is an overfitting fluctuation after 40 rounds, due to the fact that the model architecture used is the same as before in order to control the variables.

Initially, it is hypothesized that this may be due to sample heterogeneity caused by discontinuous traffic flow in real life, which results in overfitting due to misjudgment during training.

It can be seen from these results that model A has a better prediction performance for category 0 but a slightly poorer performance in predicting category 1.

TABLE III. CLASSIFICATION REPORT FOR NGSIM DATASET MODEL

Overall Accuracy				0.94
	Precision	Recall	F1-Score	Support
0	0.97	0.95	0.96	388
1	0.82	0.87	0.85	95
Macro Avg	0.9	0.91	0.9	483
Weighted Avg	0.94	0.94	0.94	483

Comparing the model based on the SUMO dataset with this model, the model based on the SUMO dataset outperforms this model on all evaluation metrics, which may be due to the fact that the model based on the SUMO dataset uses a larger dataset that is better trained. However, it should be noted that the NGSIM dataset has already been processed and may have a higher training quality than the data generated directly by SUMO, as it is more similar to the features of the validation set. Therefore, the next section will compare the performance of both on the full test set.

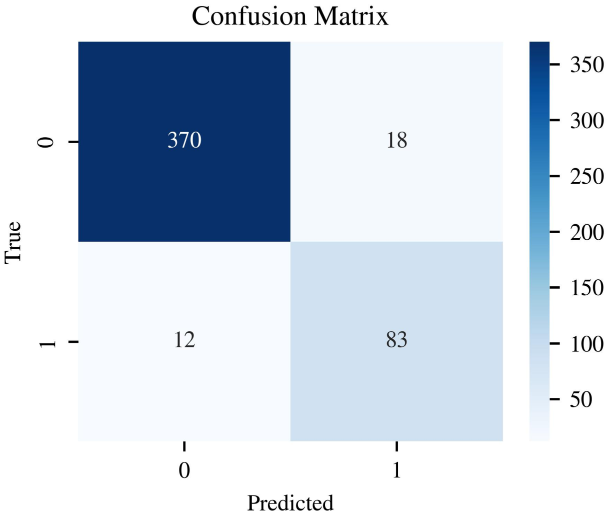

Figure 14. Confusion Matrix For NGSIM Training Set Model.

V. EXPERIMENTS

A. Dataset Preparation

We chose NGSIM as our actual test dataset. Statistics for the Peachtree and Lankershim areas were chosen as our test dataset because they are the most complete urban statistics in NGSIM.

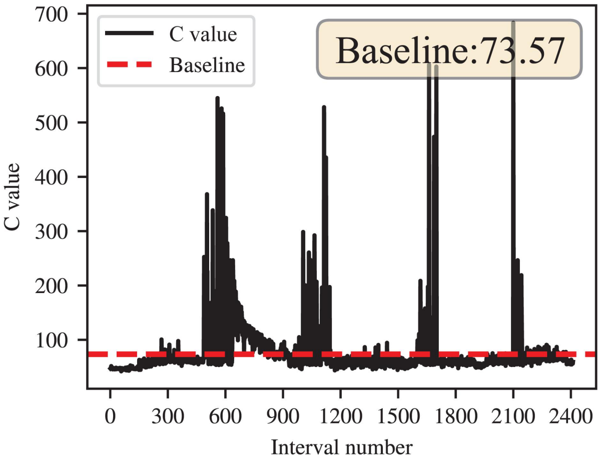

Figure 15. Test Set Baseline.

Our aim is to convert this raw data into a format more suitable for traffic flow analysis. Specifically, we want to calculate a number of metrics (e.g., average speed, average time lost, number of vehicles leaving, seen and entering, and number of seconds sampled) every 45 seconds for each 100 meter section of road.

Processing the raw data of 200,000 vehicle trips resulted in 2,411 data of regional road segments. These 2,411 traffic flows can be approximated as one day's traffic, but due to inaccuracies in the conversion process. The C-value of each data point is calculated, but this C-value is not used directly, but to make a comparison that expresses the dissimilarity and uniqueness of this data to the training data. We also calculated and output the number of normal and abnormal samples in the test set.

It should be noted that due to the existence of negative C data in the real world statistics, and the baseline 13.22 is an absolute data constant, in the initial setting of its is relative to the lowest value point 0 division (see training set), due to the data set differentiation processing is based on the 0-point processing, so in the actual division of the division, we will have 2158 normal, 253 abnormal samples in test set.

B. *Results On SUMO Trained Model*

In terms of the results, the model's predictive ability for the class 0 samples ise, i.e., it can efficiently discriminate the normal traffic flow samples, but as can be seen from the accuracy of 0.48 for the class 1 samples, the model very often treats the normal traffic flows (170 samples) as abnormal traffic flows because its own purpose is to pick out the abnormal traffic flows. 48 for the class 1 samples, the model very often treats the normal traffic flows (170 samples) as abnormal traffic flows, because the model's own purpose is to pick out the abnormal traffic flows, which may be due to the fact that they are relatively close to the cutting baseline, and the fluctuations near the traffic flow increase before the traffic flow becomes abnormal, which could be the cause of the misclassification, since the model is trained with time-slice data.

TABLE IV. Model Classification Report on Test Set

Overall Accuracy				0.89
	Precision	Recall	F1-Score	Support
0	0.95	0.92	0.94	2158
1	0.48	0.63	0.55	253
Macro Avg	0.72	0.77	0.74	2411
Weighted Avg	0.91	0.89	0.9	2411

Meanwhile, the model is better at judging above the class 1 samples, which are the abnormal traffic flows, with 63% of the abnormal traffic flow samples being flagged. Out of a total of 253 samples, only 93 were missed. Again, these samples may be due to their proximity to the cut line baseline, perhaps due to the gradual easing of the traffic flow over the course of several consecutive time slices before and after (e.g., the downward trend shown in the 600-900 interval of the baseline plot).

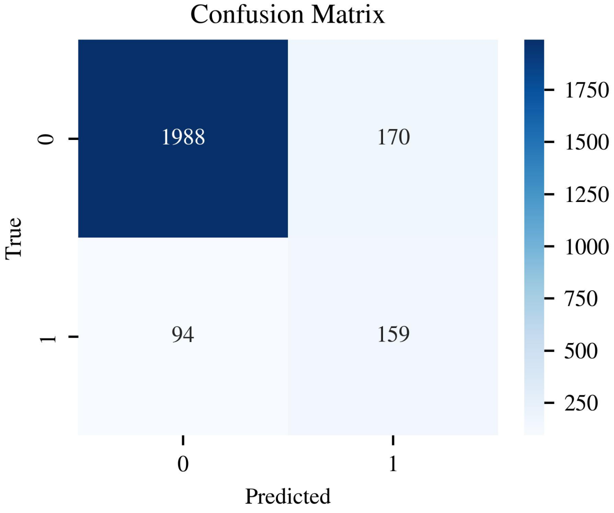

Figure 16. Confusion Matrix on Test Set.

C. *Results On Real World Trained Data*

TABLE V. NGSIM-Model Classification Report on Test Set

Overall Accuracy				0.81
	Precision	Recall	F1-Score	Support
0	0.99	0.77	0.87	1963
1	0.49	0.96	0.65	448
Macro Avg	0.74	0.87	0.76	2411
Weighted Avg	0.9	0.81	0.83	2411

For abnormal traffic (category 1), the accuracy is 0.49, which means that only about 49% of the cases predicted to be abnormal are actually abnormal. This is a relatively high false alarm rate. However, the recall for category 1 is quite high at 0.96, indicating that the model is able to correctly identify 96%

of the instances, but with a very high false alarm rate. The F1 scores, which balance precision and recall, are 0.87 for category 0 and 0.65 for category 1. These scores suggest that the model is more effective at identifying normal traffic conditions than abnormal ones, although it has improved its ability to detect abnormal conditions compared to the previous model.

For category 0, the model correctly predicted 1516 out of a total of 1963 samples but missed 447 normal traffic flows and misclassified them as anomalous, a number that was equated to almost all of the anomalous traffic flows, and while the model is to be commended for its accuracy in predicting the anomalous traffic flows, such a high rate of false positives leaves its true performance open to debate. For all anomalous traffic flows, the model is quite correct in its judgement, identifying 432 anomalous traffic flows out of 448 samples, but this may be due to the fact that the features of the training data are very close to the features of the actual data, and some of the data may have underlying exercises that lead to a rather high level of accuracy.

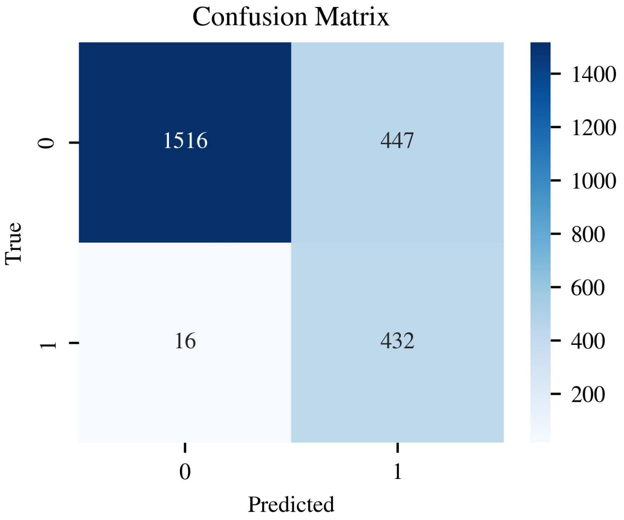

Figure 17. Confusion Matrix For NGSIM-Model on Test Set.

D. *Analyzing of Results*

Based on the results in the previous section, it can be concluded that both models perform well above 80% on unseen real data sets.

The model based on the SUMO dataset has an accuracy of 89.5%, while the model based on NGSIM has an accuracy of only 81.4%. This indicates that the SUMO-based model is generally more accurate and can correctly predict a higher proportion of observations for the categories.

For category 0 (normal traffic flow) both models have a high accuracy and for category 1 (abnormal traffic flow) both models have a lower accuracy, indicating that both models are good at predicting category 0 but struggle more with predicting category 1. This may be because there are fewer samples for category 1, making it harder for the models to learn to predict accurately. Recall is the ratio of correctly predicted positive observations to all observations in the current category. For category 0, the model trained on the SUMO dataset has a higher recall. For category 1, the recall of the NGSIM-based model is higher. This suggests that the SUMO dataset model is better at identifying category 0 samples, whereas the real NGSIM-based model is better at identifying category 1 samples.

However, this does not mean that the model based on the real NGSIM performs better, as the test set is also based on NGSIM, which leads to the possibility that more potential features are learned when the model is trained, which is not entirely fair to the model based on the SUMO dataset, and also considering the high false positive rate of the model based on NGSIM, it can be assumed that the performance of the two models in terms of precision and recall are close, which is in line with the purpose of the thesis to validate the performance of the models based on the SUMO dataset.

At the same time, F1 scores based on the weighted average of precision and recall can be analyzed. When the weighting is not taken into account, the score of the model based on the SUMO dataset is 0.74 and the score of the model based on the NGSIM dataset is 0.76, which means that the two models perform similarly under the absolute average, whereas in the score based on the weighted average, we get that the model based on SUMO has a score of 0.90, while the model based on the NGSIM dataset has a score of only 0.83, which is significantly lower than the SUMO-based model.

Looking at the combined number of false positives, we can see that the number of false positives for the model based on the SUMO dataset is 253, while the number of false positives for the model based on the NGSIM dataset is 463, which is 83% higher, which is an unsatisfactory result, especially when the total number of samples is only 2,411.

The model trained based on NGSIM is inherently specific to the test set, whereas the model based on the SUMO dataset is extensive, with data that is not linked to the NGSIM test set and is not specific, but in the analysis of the results they perform similarly and even better in most of the metrics based on the SUMO dataset, and it is reasonable to conclude that the use of SUMO-generated traffic flow data can somewhat replace realistic datasets in the field of traffic flow prediction.

VI. Conclusions

In this paper, we successfully demonstrated the use of SUMO-generated traffic flow data and LSTM model data segmentation processing software to train a deep learning model capable of determining real traffic flow conditions. Our results showed that the model trained on the SUMO dataset outperformed the model trained on the NGSIM dataset. However, the model's ability to correctly identify abnormal traffic flow indicating room for improvement. This discrepancy may be due to the imbalance in the amount of normal and abnormal traffic flow data in the dataset.

The architecture of the model and the definition of the C-value may have contributed to the decrease in the model's performance on the real dataset. Furthermore, the data required by the model was not directly available on the NGSIM dataset and had to be generated through a data processing mechanism, which could have impacted the accuracy of the C-value or source data.

The datasets generated by SUMO are not identical to real-world datasets, and measurements and counts may vary from country to country.Despite these challenges, the SUMO-based dataset generation approach has significant advantages, including the absence of data nulls, no erroneous collection information, and ease of data manipulation due to its data regularity.

In conclusion, the model's ability to make correct judgments on 89.5% of an unfamiliar dataset during training on simulated data validates our research methodology. This paper underscores the feasibility of using SUMO-based generated traffic flow datasets as a substitute for real datasets in a deep learning approach, although more research is needed to explore a wider range of methods for generating data flows.

References

[1] Li, Ziru, et al. "How do on - demand ridesharing services affect traffic congestion? The moderating role of urban compactness." Production and Operations Management 31.1 (2022): 239-258.

[2] Mandal, Vishal, et al. "Artificial intelligence-enabled traffic monitoring system." Sustainability 12.21 (2020): 9177.

[3] Tian, Yan, et al. "LSTM-based traffic flow prediction with missing data." Neurocomputing 318 (2018): 297-305.

[4] Butcher, Antony, et al. "Local magnitude discrepancies for near - event receivers: Implications for the UK traffic - light scheme." Bulletin of the Seismological Society of America 107.2 (2017): 532-541.

[5] USDOT,2023, NGSIM(Next Generation Simulation) Dataset, available at https://data.transportation.gov/stories/s/i5zb-xe34.

[6] Zhang, Lili, et al. "Research on integrated simulation platform for urban traffic control connecting simulation and practice." Scientific reports 12.1 (2022): 4536.

[7] Zhu, Yuyu, QingE Wu, and Na Xiao. "Research on highway traffic flow prediction model and decision-making method." Scientific reports 12.1 (2022): 19919.

[8] Shaygan, Maryam, et al. "Traffic prediction using artificial intelligence: review of recent advances and emerging opportunities." Transportation research part C: emerging technologies 145 (2022): 103921.

[9] Kumar, Nishant, and Martin Raubal. "Applications of deep learning in congestion detection, prediction and alleviation: A survey." Transportation Research Part C: Emerging Technologies 133 (2021): 103432.

[10] Offor, Kennedy J., Matthew Hawes, and Lyudmila Mihaylova. "Short term traffic flow prediction with particle methods in the presence of sparse data." 2018 21st International Conference on Information Fusion (FUSION). IEEE, 2018.

[11] Greff, Klaus, et al. "LSTM: A search space odyssey." IEEE transactions on neural networks and learning systems 28.10 (2016): 2222-2232.

[12] Gaouar, Nihal, Mohamed Lehsaini, and Tawfiq Nebbou. "CCITL: A cloud - based smart traffic management protocol using intelligent traffic light system in VANETs." Concurrency and Computation: Practice and Experience 35.12 (2023): e7686.

[13] M a, Xiaoyi, et al. "Evaluation of accuracy of traffic flow generation in SUMO." Applied Sciences 11.6 (2021): 2584.

[14] Andrej Karpathy 2014,ICLR(International Conference on Learning Representations), Available at https://iclr.cc/archive/2014/

[15] Kavehmadavani, Fatemeh, et al. "Intelligent Traffic Steering in Beyond 5G Open RAN based on LSTM Traffic Prediction." IEEE Transactions on Wireless Communications (2023).

[16] Guide Company, 2023, Guide Map,available at https://www.amap.com/

[17] Lindemann, Müller, T., Vietz, H., Jazdi, N., & Weyrich, M. (2021). A survey on long short-term memory networks for time series prediction. Procedia CIRP, 99, 650–655. https://doi.org/10.1016/j.procir.2021.03.088

[18] Wang, Ma, C., Qiao, Y., Lu, X., Hao, W., & Dong, S. (2021). A hybrid deep learning model with 1DCNN-LSTM-Attention networks for short-term traffic flow prediction. Physica A, 583, 126293–. https://doi.org/10.1016/j.physa.2021.126293