TRAFFIC PREDICTION USING DEEP LEARNING

Shivam Jhunjhunwala Affan Sayed   
School of Computer Engineering and Technology   School of Computer Engineering and Technology   
MIT World Peace University MIT World Peace University   
Pune, India Pune, India   
[shivamjhunjhunwala4dec@gmail.com](mailto:shivamjhunjhunwala4dec@gmail.com) [affansayed36@gmail.com](mailto:affansayed36@gmail.com)

Ridhima Kulkarni Shashank Mishra   
School of Computer Engineering and Technology School of Computer Engineering and Technology   
MIT World Peace University MIT World Peace University   
Pune, India Pune, India   
rid1224k@gmail.com  shashankmishr71@gmail.com

***Abstract*—** **Traffic congestion has been one of the major issues that most metropolises are currently facing, despite efforts to minimise and reduce it. People prefer to use their own cars to meet their mobility needs these days rather than taking the public transportation system. This is a result of how easy it is to use the cars and when you can use them. Urban engineers, planners, and policymakers have identified traffic congestion as one of their biggest challenges as a result of the severe traffic congestion and prolonged wait times at traffic lights that result. Congestion has been established as an unavoidable fact of urban life, shaped by car-oriented urban development and the rapid rise in vehicle ownership. Along with other negative consequences on the quality of urban living, the worsening urban air quality is a sign of the growing impact of congestion. Our research focuses on the prediction of traffic congestion using a Deep Learning algorithm, specifically CNN (Convolutional Neural Network) with K-FOLD.**

***Keywords—Traffic Congestion, Data Pre-processing, Deep Learning, Machine Learning***

# **Introduction**

Managing and controlling traffic is a significant issue in many areas, particularly in areas with expanding cities and populations. Time multiplexing is used by traffic signals to reduce congestion at intersections. All signalised intersections in various nations use fixed mode control. The only drawback of traffic light is the delay in reaching the goal is the only negative of traffic lights (stop time or waiting time). A performance metric for traffic controller effectiveness is intersection delay [1]. The effectiveness of traffic travelling through intersections is influenced by the phase, sequence, and timing of traffic lights. Phase, sequence, and timing are handled by the adaptive signal controller. The timing and placement of traffic signals must be optimised in order to reduce traffic congestion. Due to predictability and other considerations, traffic light timing is severe and blind [2].

The topic of anomaly detection in huge, high-volume astronomical datasets is covered in this study. The most potent and intensive deep learning technique is considered in a machine learning-based approach.

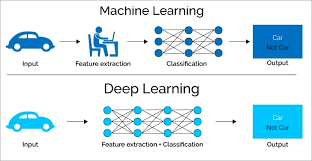


Fig. 1. Basic Machine Learning and Deep Learning approach [3].

# Literature review

The study of traffic congestion has faced a number of data-intensive problems that have been handled utilising machine learning and data science techniques. Machine learning is a powerful technique for identifying patterns and extracting important data. In order to extract dynamic patterns from traffic data streams and to incorporate AI modules for real-time traffic analysis and adaptive traffic control, a new smart traffic management platform was developed in this article.[4]

The main advantage of the proposed platform is that its AI modules are built to effectively address the major issues facing future transportation systems, including the widespread adoption of IoT devices, the need for analysis and control technologies to be more responsive and self-evolving, and the need to take into account social behaviours [5].

In this study, we present a framework for automatically identifying bike riders without helmets in real time using video feeds from the city's surveillance network. We also cover the visual big data analytics framework, its underlying methodologies, and its application. The first phase of the proposed approach uses object segmentation and background subtraction techniques to identify bike riders in surveillance videos [6].

In the second phase, it identifies the cyclist's head and extracts the necessary features to determine whether or not the rider is wearing a helmet. They have offered a performance comparison of three frequently used feature representations for classification, namely the histogram of oriented gradients (HOG), the scale-invariant feature transform (SIFT), and local binary patterns (LBP), in order to assess our approach [7].

This study concentrates on creating prototypes for the various prototype actions listed above and using them to execute tiny traffic mimicry.

They have also researched the benefits and drawbacks of various imitation optimization strategies, as the effectiveness of the aforementioned prototypes will be greatly influenced by the type of traffic, the layout of the roads, and the style of optimization employed to regulate it [8].

In this paper, they planned some new technology. Additionally, it was used as a four-way road. In order to count a vehicle, we frequently use the sensing component. We frequently keep an auto and time count for large amounts of information.

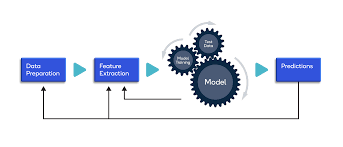
that the sensing component determines a vehicle's count and compares it to the enormous knowledge store [9]. Time is thus saved. Our preferred planning strategies include the Adaboost and statistical regression rules.

This paper presents a system that employs information visualization techniques to analyse urban traffic data and the impact of traffic emissions on urban air quality. Effective visualizations allow citizens and public authorities to identify trends, detect congested road sections at specific times, and perform monitoring and maintenance of traffic sensors [10]. Since road transport is a major source of air pollution, also the impact of traffic on air quality has emerged as a new issue that traffic visualizations should address.

Trafair Traffic Dashboard exploits traffic sensor data and traffic flow simulations to create an interactive layout focused on investigating the evolution of traffic in the urban area over time and space. The dashboard is the last step of a complex data framework that starts from the ingestion of traffic sensor observations, anomaly detection, traffic modelling, and also air quality impact analysis [11]. The paper presents the results of applying our proposed framework on two cities (Modena, in Italy, and Santiago de Compostela, in Spain) demonstrating the potential of the dashboard in identifying trends, seasonal events, abnormal behaviours, and understanding how urban vehicle fleet affects air quality.

This project mainly adopts the method of combining theoretical research and numerical verification. Firstly, the complete and normative design of road traffic template is realized, and the reference sequence of road traffic characteristics is established based on historical road traffic data; Then, the spatial-temporal correlation, redundancy and structure characteristics of road traffic big data are analysed, and an effective method for selecting high-dimensional features of road traffic big data is proposed; Finally, based on artificial intelligence and other methods, the knowledge base of intelligent understanding of road traffic big data is constructed, and the fuzzy reasoning interpretation algorithm of road traffic big data is constructed, and finally the intelligent understanding of road traffic big data is realized [12].

# SYSTEM ARCHITECTURE



The system architecture that was used is depicted in the accompanying diagram. Finding the proper dataset and altering the data to get it ready for an ML model is the first stage in the process. We train the model after selecting the model to create (CNN). The model is then assessed using some test data, and the prediction accuracy is calculated.

# Methodology

* 1. Data Collection:

The dataset being used has two values, "1" and "0," and it describes traffic congestion. Congestion has been identified if the value is "1"; else, it is not. The NGSIM website was used to get the data.

## Data Preparation and Preprocessing :

Verifying the distribution of vehicle width in the data is the first stage in the pre-processing procedure. To accomplish this, just use the matplotlib library.

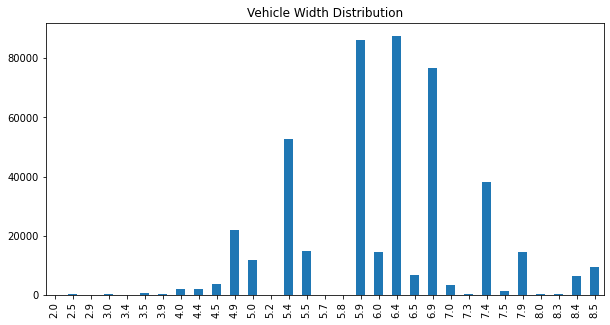


Fig. 2. Distribution of vehicle widths in the data [13].

The flow data is then plotted to ensure accuracy. The flux intensity's biggest and lowest values are very different from one another. It demonstrates the requirement for intensity. It shows that there is a need to normalize the data.

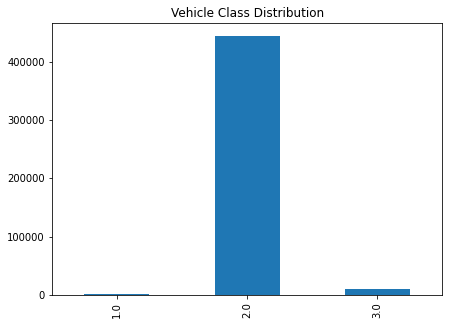


Fig. 3. Graph to show v\_Class distribution [14].

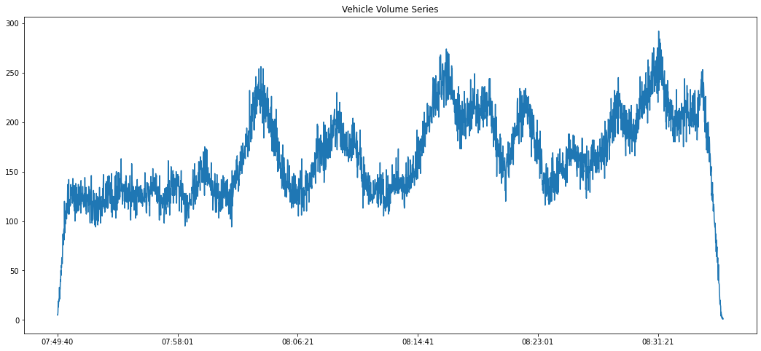
Next, we added vehicle volume at the junction at a particle time.

Fig. 4. Time series to show the behaviour of the vehicle’s velocity and acceleration over time

[15].

By using the undersampling strategy, the majority class's data samples will be reduced to the same number as those in the minority class. To do the undersampling, we'll use the Pandas sample method. We combine the minority class and the under sampled majority class. A single data frame will result from it. The number of data samples in the new balanced dataset is then printed.

* 1. *Algorithms*

Using the Keras Library and Tensorflow, a convolutional neural network has been utilised to detect traffic congestion.

The undersampling technique will reduce the data samples in the majority class (0) to have the same number as the minority class. We use the Pandas sample method to perform undersampling.

Additionally, we must obtain train and test splits. The stratify parameter will make sure that after separating, the classes are still balanced. We employ the same splitting ratio as well. We feed our deep neural network the dataset once more. After that, we forecast traffic congestion using pandas predict function.

**Convolutional Neural Networks:**

Using both time series and picture data, CNNs are a particular kind of neural network that may extract important information. It is extensively used for pattern and object recognition. CNNs can also be used to categorise signal and audio data. Three layers make up a fundamental CNN model: a convolutional layer, a pooling layer, and a fully connected (FC) layer [16].

**Convolutional Layer:**

In this layer, all significant computations take place. To make the network more complex, convolutional layers can be added in multiples. A dot product is produced between the input pixels and the filter over numerous image iterations. The result is referred to as a feature map. The image is finally transformed into numerical data that CNN uses to determine its predictions [17].

**Pooling Layer:**

Like the convolutional layer, the pooling layer likewise sweeps the entire image. However, some of the parameters are reduced, which results in information loss. As a result, the algorithm runs much more quickly and is simpler [18].

**Fully Connected Layer:**

Based on the features that have been gathered from the preceding levels, classification takes place in this layer. We have three convolutional layers installed in our software but no pooling layer. This is because a shallow and simple neural network will suffice to perform the required categorization [19].

**K-Fold:**

A resampling technique called cross-validation is used to assess machine learning models on a small data sample.

The process contains a single parameter, k, that designates how many groups should be created from a given data sample. As a result, the process is frequently referred to as k-fold cross-validation. In applied machine learning, cross-validation is mostly used to gauge how well a machine learning model performs on untrained data. That is, to use a small sample to assess how the model will generally perform when used to generate predictionson data that was not utilised during the model's training [20].

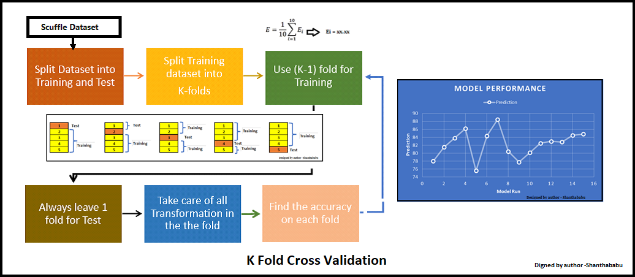


Fig. 5. System architecture for K-Fold algorithm [21].

## Results

Using CNN and K-Fold, we have achieved an accuracy of 88.739% on the test data.

* 1. Conclusion

In our paper, we presented a cross validation method and deep learning algorithm-based algorithm for traffic congestion prediction. There is still much work to be done because we haven't taken into consideration various traffic types or fine tweaking.

##### References

**[1]** <https://tharun-pal1a43.medium.com/intelligent-traffic> management-using-big-data-analytics-and-iot-a5e84If2cf7d

**[2]** “Big Data and Transport: Understanding and assessing options,” OECD/ITF, 2 D15. [Online]. Available: http://www.itfoec “Googles Self-Driving Cars are Coming Sooner Than You Think.” <http://fortune.com/> 2015/10/02/google-self-driving cars-soon/.

**[3]** Vlahogianni, E. [. ZO15. “Computational Intelligence and Optimization for Transportation Big Data: Challenges and Opportunities”. In Engineering and Applied Sciences Optimization, 107—1Z8. Springer.

**[4]** Lippi, N., M. Bertini, and P. Frasconi. 2013. “Short-Term Traffic Flow Forecasting: An Experimental Comparison of Time-Series Analysis and Supervised Learning”. IEEE Transactions on intelligent Transportation Systems 14 (2): 871- B82.

**[5]** Tampere, C. M., R. Corthout, D. Cattrysse, and L. H. Immers. **2011. “A** Generic Class of First Order Node Prototypes for dynamic Macroscopic Imitation of Traffic Flows”. Transportation Research Part B: Methodological 45 (1): 289-309.

[6] Gipps, P.G.19B1. “A Behavioral Car-Following Prototype for Computer Imitation”. Transportation Research Part B: Methodological lS (2): 105-111.

Gipps, P. 1986. “Multsim: a Prototype for Simulating Vehicular Traffic on Multi -Lane Arterial Roads”.

**[7]** Mathematics and Computers in imitation 28 (4): 291-295. [B] Bando, M., K. Hasebe, A. Nakayama, A. Shibata, and Y. Sugiyama. **1995.** “Dynamical Prototype of Traffic Congestion and Numerical Imitation”. Physical Review E 51 (2): 1035.

Helbing, D., and B. Tilch. 199B. “Generalized Force Prototype of Traffic Dynamics”. Physical Review E **58 (1): 133**

**[8]** d.org/sites/default/files/docs/l5epb\_bigdata D.pdf. [Accessed: 30Piar -2017]. Levin, Doron **2015.**

[9]<https://www.google.com/imgres?imgurl=https%3A%2F%2Fassets-global.website-files.com>

[10] <http://ngsim.fhwa.dot.gov>

[11]<https://www.google.com/imgres?imgurl=https%3A%2F%2Fdeveloper.qualcomm.com%2Fsites%2Fdefault%2Ffiles%2Fattachments%2Flearning_resources_01>

[12] <https://www.section.io/engineering-education/how-to-handle-imbalanced-data-in-deep-neural-networks/>

[13] <https://www.analyticsvidhya.com/blog/2022/02/k-fold-cross-validation-technique-and-its-essentials/>

[14]https://www.techtarget.com/searchenterpriseai/definition/convolutional-neural-network

[15]<https://journalofbigdata.springeropen.com/articles/10.1186/s40537-021-00444-8>