The Similarity between IOT and Artificial Intelligence

"Smart City" as a study case

*RAJI Fatima Zahraa, ZIYAD Mouhssine, RAFALIA Najat*

Department of Computer Science

Faculty of Science

University of Ibn Tofail, Kenitra, Morocco

# ABSTRACT:

*Every country is currently embracing the concept of smart cities (SC), aiming to incorporate advanced technologies into various aspects of urban life. Among the most challenging tasks in smart cities is the efficient management of traffic, which necessitates the adoption of diverse approaches. To address this issue, an artificial intelligence framework based on Internet of Things (IoT) has been implemented to estimate vehicle traffic in specific city areas and facilitate efficient traffic management*. *Traffic data was collected using IoT edge sensors. The data obtained from an IOT platform has been processed using an AI system which gives as a result a long-term or short-term prior forecast. To create the proposed forecasting system, machine learning (SVM, LR, KNN) and deep learning (LSTM) algorithms were used. The model under consideration has showcased superior performance, making it the most accurate and widely employed approach for smart city traffic management.*

***Keywords: IoT, IA, SC, LSTM, ML, deep learning , traffic, prediction.***

# INTRODUCTION:

Smart cities, characterized by their integration of advanced technologies, aim to improve the quality of life of inhabitants and enhance the efficiency of urban services. Traffic management stands as a crucial area within smart city initiatives [26]. Rapid urbanization and technological advancements have presented modern cities with the critical challenge of effectively managing traffic. Traffic congestion, resulting from the growing urban population, poses numerous obstacles to these cities.

To address this issue, emerging technologies such as the Internet of Things (IoT) [21] have garnered significant interest. By leveraging IoT's interconnection capabilities among sensors and devices, cities can gather real-time data on traffic patterns, road conditions, and vehicle movements. This data, often referred to as big data, holds immense potential for developing accurate and dynamic traffic prediction models. However, the volume, velocity, and variety of traffic-related data generated by IoT devices present both challenges and opportunities for traffic prediction.

Traditional traffic modeling techniques struggle to effectively process and analyze such vast and diverse datasets. Yet, advancements in big data analytics and machine learning algorithms have opened new avenues for extracting valuable insights and enabling accurate predictions of traffic conditions. The benefits of precise traffic forecasting and optimized traffic flow extend beyond congestion reduction. By intelligently routing vehicles and predicting road conditions an hour in advance, cities can minimize travel times, reduce fuel consumption, emissions, and noise pollution caused by traffic crises, enhance the efficiency of public transport, and improve overall safety. Moreover, predictive traffic modeling can support emergency response planning, assist in urban infrastructure development, and facilitate effective transport demand management [17].

This research aims to contribute to the advancement of traffic prediction techniques in the context of smart cities. By exploring the potential of IoT, big data analytics, and optimization algorithms, this study aims to develop a comprehensive and adaptable framework for traffic prediction and flow optimization. The research will investigate the effectiveness of various machine learning algorithms in processing and analyzing large-scale traffic data.

The subject of smart city traffic prediction and forecasting, particularly with a focus on machine learning and deep learning in conjunction with IoT-based data collection [19][20], has gained significant attention in both intervention and research domains. For the review of related work, articles were selected from top journals across various databases.

Neelakandan et al. (2021) reported the outcomes of a benchmarking research study on an IoT-based traffic prediction and signal management system for a smart city. Their proposed system, which utilized an Intel 286 microprocessor, encompassed five phases: data gathering via IoT, feature engineering, data separation, traffic data optimization, and traffic management. The Elman neural network method was employed to classify traffic data in congested areas. The proposed system demonstrated superior performance compared to state-of-the-art approaches.

In another study, Lilhore et al. (2022) designed an adaptive traffic management system for smart cities using machine learning and IoT. Their work encompassed multiple scenarios to address concerns within the transportation infrastructure of the city. Additionally, their suggested system incorporated a machine learning-based clustering algorithm to detect anomalies. The traffic light schedules were regularly updated based on traffic movement and expected flow, considering neighboring signal junctions. Simulation results indicated that the proposed approach outperformed existing transportation strategies, providing an added advantage to smart city-based transportation systems. The authors claimed that the proposed method reduced traffic congestion, vehicle idle times, and accidents.

This article is structured into distinct sections. We begin with an overview of the project's architecture and the employed algorithms in the first section. The subsequent section covers the implementation, providing detailed explanations of the various techniques utilized. Finally, we present the results, draw conclusions based on our findings, and discuss future perspectives.

# METODOLGY

**IoT Architecture for Smart City Traffic**: An IoT-based data collection model was utilized to generate the dataset for training machine learning (ML) algorithms. Figure 1 illustrates the architecture of an IoT system designed for collecting and modeling traffic data. The model incorporates an IoT traffic system installed at two specific points: the starting and ending locations of a road. To ensure precise and uninterrupted data collection, sensors were strategically placed away from intersections to avoid interference caused by vehicles stopping at red lights. The collected data is processed in real-time by the sensor.

Data processing involves the analysis of the collected data to extract relevant information, such as the number of vehicles and their speed. Additionally, the sensor may apply initial data filtering or aggregation techniques to reduce noise and handle high-frequency data. For data transmission, the sensor is equipped with a communication module (e.g., Wi-Fi or cellular connectivity) to transmit the processed data. The sensor establishes a connection with a cloud-based storage platform.

In this case, the cloud storage platform employed was Amazon Web Services (AWS). The transmitted data is received and securely stored in the cloud, providing scalable storage capabilities to accommodate large volumes of traffic data. then for this collected data will be subsequently processed by machine learning and deep learning algorithms

.

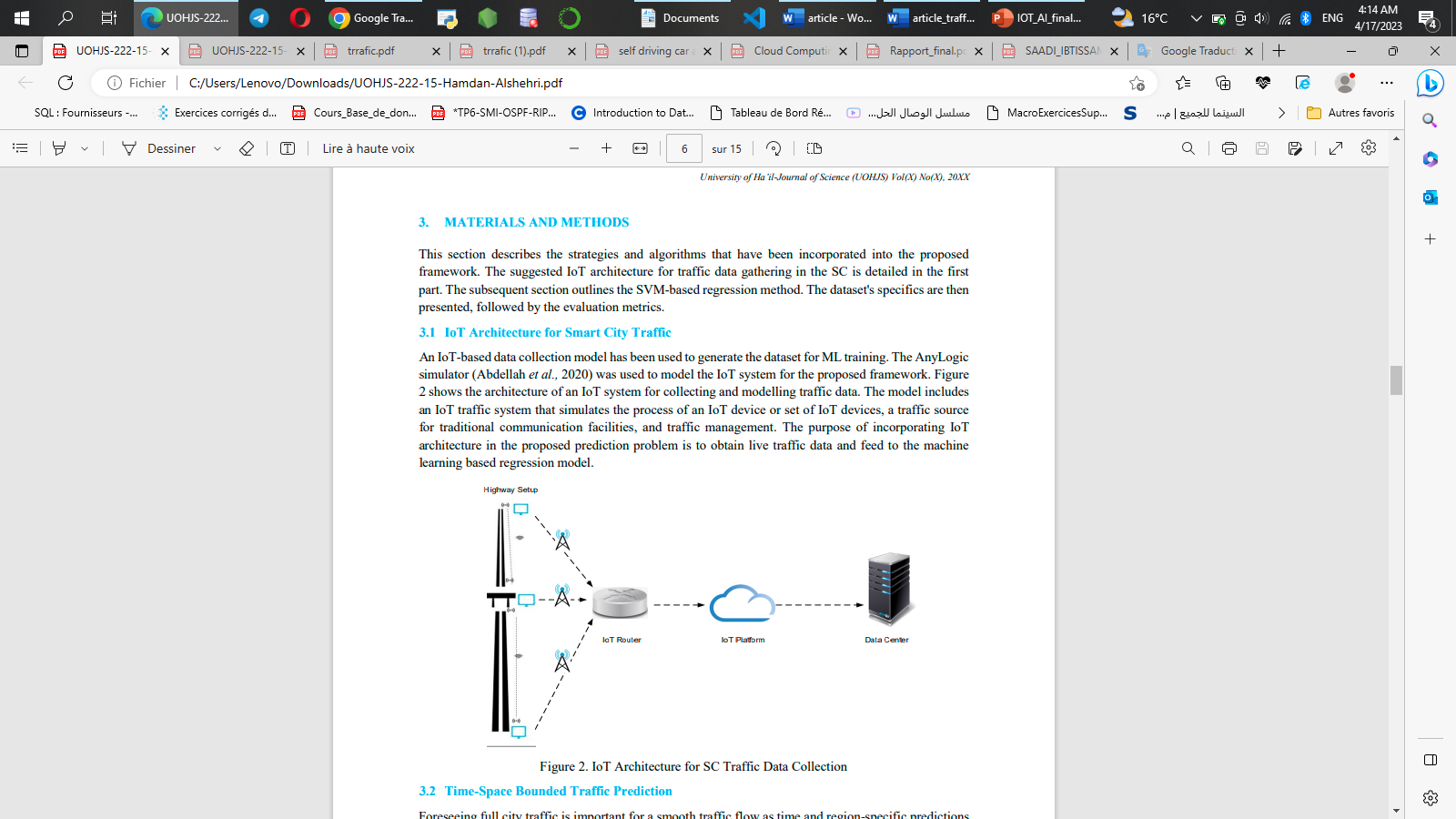


Figure 1 : IoT Architecture for SC Traffic Data Collection

**Linear Regression:** Linear regression may be defined as the statistical model that analyzes the linear relationship between a dependent variable with given set of independent variables. Linear relationship between variables means that when the value of one or more independent variables will change (increase or decrease), the value of dependent variable will also change accordingly (increase or decrease) [27].

**KNN** : stands for k-Nearest Neighbors, which is a popular algorithm used in machine learning for classification and regression. It is a type of supervised learning algorithm that can be used for both classification and regression problems [27].

**SVR (Support Vector Regression)**: is a machine learning algorithm used for regression tasks [28]. It is a variant of the Support Vector Machine (SVM) algorithm used for classification tasks.

The goal of SVR is to find a function that maps input data to output values, while minimizing the prediction error. It does this by constructing a hyperplane in a high-dimensional feature space, which is defined by a subset of training samples, called support vectors.

**Recurrent Neural Networks (RNNs)**: are a type of artificial neural network that is designed to process sequential data. Unlike traditional neural networks, which process fixed-sized inputs, RNNs can handle inputs of variable length, such as sentences, speech, and time-series data [29].

RNNs can be implemented using popular deep learning frameworks such as TensorFlow [25] and PyTorch, and are widely used in a range of applications such as speech recognition, machine translation, and video analysis.

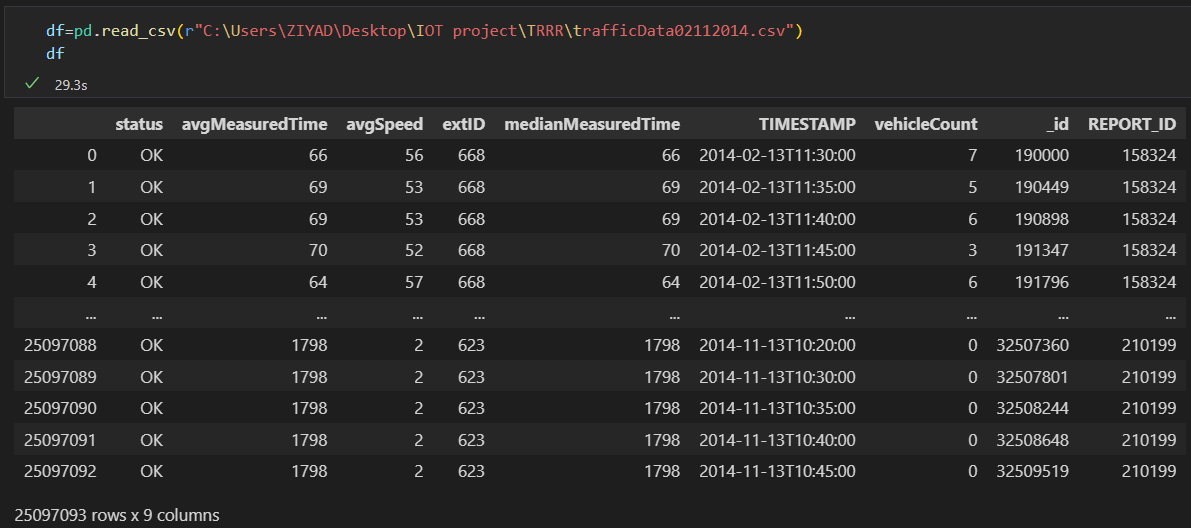
**LSTM** : stands for Long Short-Term Memory, which is a type of recurrent neural network (RNN) that is designed to overcome the vanishing gradient problem that occurs in traditional RNNs. The vanishing gradient problem occurs when the gradients used to update the network weights become very small, making it difficult for the network to learn long-term dependencies.

Training an LSTM involves feeding it a sequence of inputs and targets, and adjusting the network weights using a process called backpropagation through time. The goal is to minimize the error between the predicted outputs and the actual outputs.

LSTM networks can be implemented using popular deep learning frameworks such as TensorFlow and PyTorch, and are widely used in a range of applications such as speech recognition, machine translation, and video analysis.

**DATASET:**

In this article, we explore a valuable collection of datasets focusing on vehicle traffic, Provided by City of Aarhus in Denmark [18].

****

This dataset generated by observing vehicle traffic between two specific points: the start and end of a road. To ensure accurate and uninterrupted data collection, sensors were placed at a distance from intersections, thus avoiding the interference caused by vehicles stopping at red lights. This meticulous setup allows for a comprehensive analysis of traffic flow and patterns along the road segment, providing valuable insights into the dynamics of vehicle movement. The SensorConnection is accountable for gathering sensor readings. This achieved by establishing a network connection through a RESTful endpoint [19].

After fetching the data, it passes through a Data wrapper model that is responsible for semantically annotating the data. However, for our specific case, we are solely interested in the raw data itself and do not require the semantic annotations provided by the Data wrapper model.

The dataset under consideration is divided into two types of files: raw data and metadata.

The raw data files consist of measurements obtained by sensors deployed in the area of interest. These sensors continuously capture and record various traffic-related metrics at regular intervals, typically every 5 minutes. The measurements include the number of vehicles passing through, average speed of the vehicles, and the time taken for each measurement interval.

The columns used in our dataset include:

AVGMEASUREDTIME: the time elapsed during the passage of vehicles according to their calculated speed by average.

AVGSPEED: the Average Speed of the past vehicles.

EXTID: id of each boulevard.

MEDIANMEASUREDTIME: the time elapsed during the passage of vehicles according to their speed calculated by the median.

TIMESTAMP: the date and time of each recording incremented by 5 min

VEHICLECOUNT: the number of vehicles passed

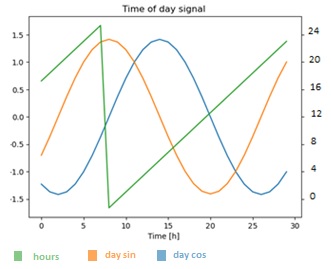
\_ID: primary key of our data

REPORT\_ID: id of the sensor (camera) found in each boulevard

In addition to the raw data files, the dataset also includes metadata that provides essential contextual information. The metadata files contain details such as the position of each of the two sensors used for data collection, the distance between these sensors, and their corresponding geographical coordinates. These specifics help establish a clear understanding of the physical layout and placement of the sensors along the road segment.

**FEATURE ENGINEERING**

Before start building a model, it is crucial to comprehensively analyze the data and ensure its suitability for feeding into the model. While the Date Time column contains valuable information, its current string format is not optimal. Additionally, treating time as a linear input may not capture the underlying patterns effectively, as 23:00 and 00:00 should be considered close to each other and smoothly transition. Given that time exhibits a clear daily periodicity, addressing this periodicity becomes essential. One approach to handle periodicity is by utilizing sine and cosine transforms to extract meaningful signals representing "Time of day" and "Time of year." These transforms can enhance the representation of time-related patterns and enable the model to better capture the temporal dynamics present in the data.

****

In Figure 4, the green plot represents the time in its original form, denoted as hours ranging from 0 to 23. On the other hand, the remaining plots illustrate the time after applying sinusoidal and cosine transformations [20].

To achieve the objective of predicting long-term traffic patterns, it would be advantageous to aggregate the data from 5-minute intervals into larger time intervals, such as one hour. By consolidating 12 rows of data into a single row, a new dataset is generated where each row represents one-hour intervals. This aggregation process reduces the overall data volume, enhancing the manageability and efficiency of subsequent analysis. This becomes particularly significant when dealing with extensive datasets spanning extended time periods.

* **SPLIT THE DATA**

We use a (70%, 20%, 10%) split for the training, validation, and test sets. the data is not being randomly shuffled before splitting. This is for two reasons:

It ensures that chopping the data into windows of consecutive samples is still possible. It ensures that the validation/test results are more realistic, being evaluated on

the data collected after the model was trained.

* **Models**

Preprocessing, training, testing, and simulation processes were conducted on a laptop equipped with an Intel i5 10300H processor, 16 gigabytes of RAM, and a GTX 1650 with MaxQ.

To monitor and compare the performance of our models, we utilize two key metrics: accuracy (R2) and mean absolute error (MAE).

This first task is to predict traffic three hour into the future, given the current value of all features. The current values include the current target.

In Regression, supervised machine learning algorithms (linear regression, k-nearest neighbor's algorithm, and Support vector machines) [27][28]. A model that just returns target after three hours as the prediction. Our models scored as following: (LR: 41%, SVR: 46%, KNN: 43%). Of course, these models will work less well if we make a prediction further in the future.

For deep learning. We chose to work with Recurrent Neural Network (RNN), a type of neural network well suited to time series data. RNNs process a time series step-by-step, maintaining an internal state from time-step to time-step. We used an RNN layer called Long Short-Term Memory.

Prior to training our model, it is imperative to employ the technique of data windowing to prepare our dataset. Data windowing involves partitioning the data into smaller, sequential subsets known as windows. The models will make a set of predictions based on that window. We have chosen a width size of 24 (24 hours as input) and an offset of eight (8 hours of predictions).

After numerous attempts and iterations of the deep learning model, we finally achieved a remarkable score of 91%.

Here is a comparison of all the models we created, highlighting their respective performances:

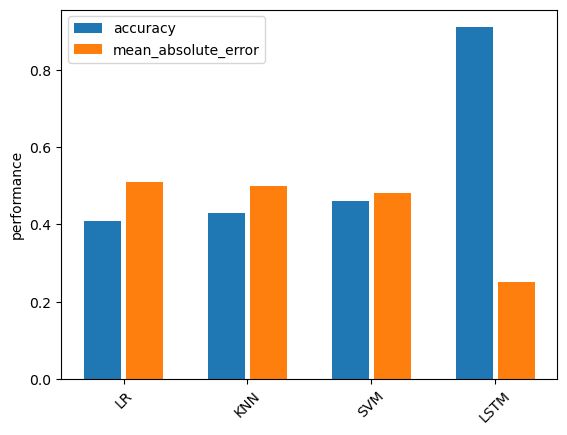


Figure 5: performance comparison graph

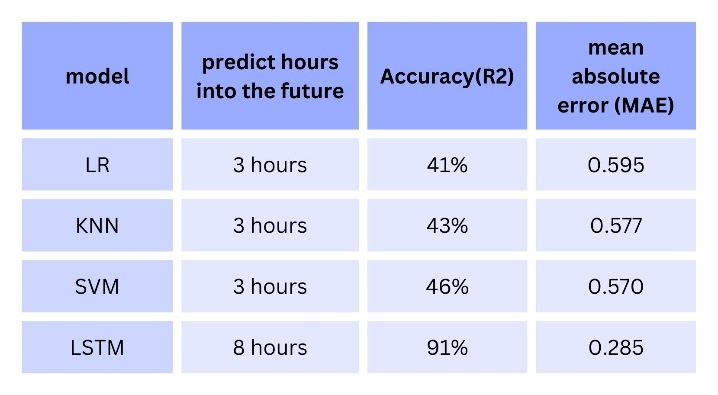


Figure 6: performance comparison table

as shown in figure 6, a comparison of machine learning and deep learning algorithms showing their accuracy and their prediction error rate, machine learning algorithms are not reliable and efficient enough to predict traffic, on the other hand, LSTM was perfect for a prediction with a minimal error rate.

# RESULTS

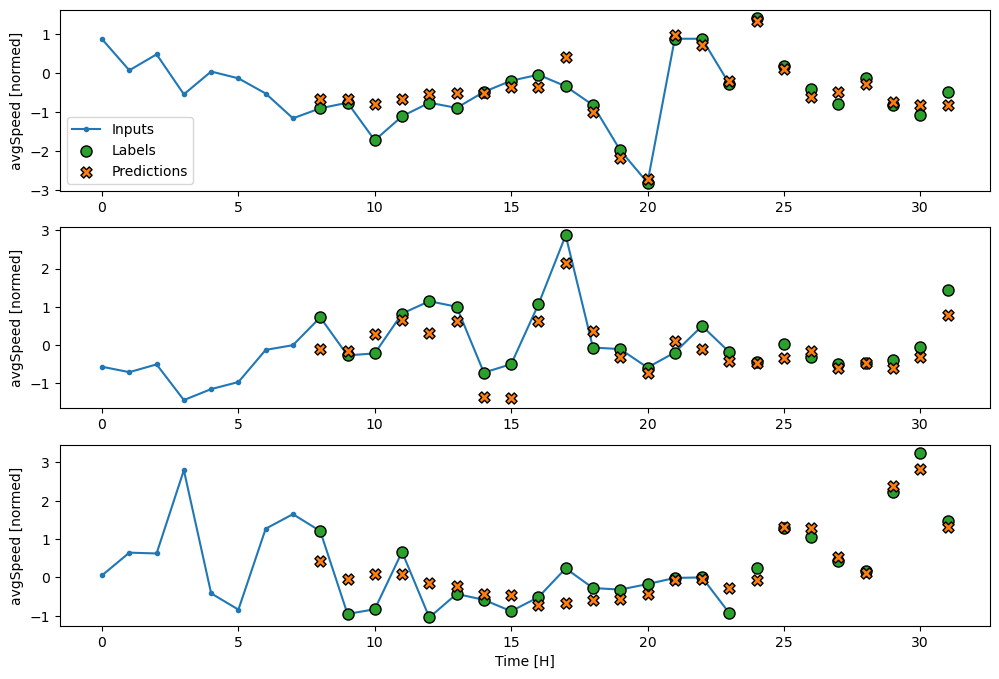
In this phase, we have concluded that the best model for simulation is LSTM Model. 

Figure 7 : LSTM model result

In the figure 7, tree test has been set in order to choose the accurate one for the simulation.

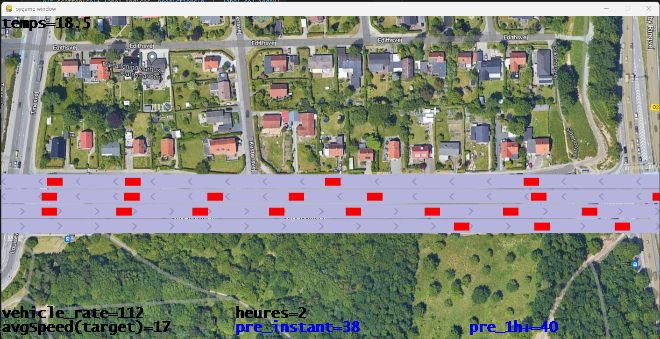


Figure 8:Simulation in case of congestion

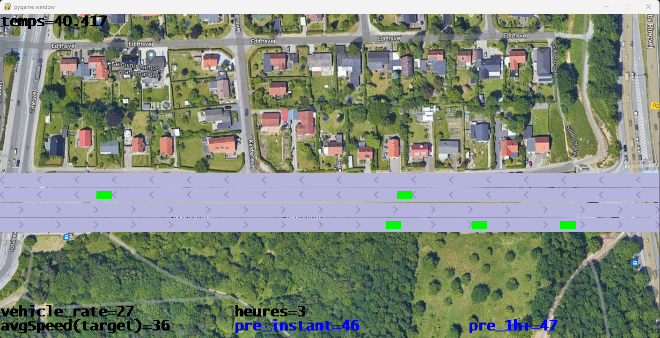


Figure 9:Simulation under normal conditions (no congestion)

In our simulation [30] carried out using pygame, it is designed to show our work clearly and approach the result to reality, we accurately predict the number of vehicles that traverse a specific segment instantly, as well as provide forecasts for the next hour. In case of congestion, we visually represent the affected vehicles in red (Figure 8), while in non-congested scenarios (Figure 9), they are represented in green. This color-based visualization helps to distinguish and understand the different traffic conditions in our simulation.

# CONCLUSION

The Internet of Things (IoT) involves interconnected devices that communicate over the internet, while Artificial Intelligence (AI) enables devices to learn from data and experiences. In the context of smart cities, particularly in transportation engineering, the convergence of these two technologies holds immense potential for traffic forecasting. Leveraging historical data provided by IoT sensors and processed by intelligent systems, time-based predictions can be made.

In our study, we developed a traffic flow prediction system using a deep learning algorithm, specifically Long Short-Term Memory (LSTM), After conducting a comprehensive evaluation of the machine learning algorithms, we determined that their accuracy was not up to the desired standards. Despite our initial efforts, these algorithms did not yield satisfactory results in accurately predicting traffic flow. Therefore, it became evident that an alternative approach was necessary to try an other model , By implementing the LSTM model, we were able to make significant improvements in predicting real-time vehicle count and future traffic patternsThis empowers the audience with valuable insights into the current traffic flow situation, enabling them to plan their routes efficiently, whether for the short or long term.

However, it is important to note that our model is not without limitations. There is room for further enhancements to achieve even better results, which can be utilized in various ways, all with the shared objective of enhancing traffic conditions and improving the lives of citizens.

# REFERENCES

* [1] Deekshetha, Shreyas Madhav , and Amit Kumar Tyagi, “Traffic Prediction using Machine Learning “
* [2] Hamdan Alshehri,”Time-Space Bounded Traffic Forecasting Model for Smart Cities using IoT and Machine Learning “
* [3] Noor Afza Mat Razali1\* , Nuraini Shamsaimon1 , Khairul Khalil Ishak2 , Suzaimah Ramli1 , Mohd Fahmi Mohamad Amran1 and Sazali Sukardi “Gap, techniques and evaluation: trafc fow prediction using machine learning and deep learning “
* [4] Wangyang Wei, Honghai Wu et Huadong Ma. “An autoencoder and LSTM-based traffic flow prediction method”. In: Sensors 19.13 (2019), p. 2946.
* [5] Intelligent Traffic Management System Based on the Internet of Vehicles (IoV) (hindawi.com)
* [6] https://actualiteinformatique.fr/internet-of-things-iot/quest-ce-que-iot-internet-of-thingsinternet-des-objets
* [7] https://www.synox.io/actualites-sectorielle/4-choses-a-savoir-sur-linternet-desobjets/#:~:text=Comment%20fonctionne%20l'IoT%20%3F,fil%20sur%20des%20plateformes%20IoT.
* [8] eelakandan, S., Berlin, M., Tripathi, S., Devi, V., Bhardwaj, I. et Arulkumar, N., 2021. Système de prévision du trafic et de contrôle des feux de circulation basé sur l'IdO pour la ville intelligente. Soft Computing, 25(18), pp.12241- 12248.
* [9] Traffic Prediction using Machine Learning Deekshetha H R 1 , Shreyas Madhav A V\* 1 , and Amit Kumar Tyagi 1,2 1 School of Computer Science Engineering, Vellore Institute of Technology, Chennai, Tamilnadu, India 2 Centre for Advanced Data Science, Vellore Institute of Technology, Chennai, Tamilnadu, India
* [10] Aniekan Essien et al. “A deep-learning model for urban traffic flow prediction with traffic events mined from twitter”. In : World Wide Web (2020), p. 1-24.
* [11] Saurabh Rathor. “Simple RNN vs GRU vs LSTM :-Difference lies in More Flexible control”. In : Web page : https ://medium. com/@ saurabh. rathor092/simple-rnn-vsgru-vs-lstm-differencelies-in-more-flexible-control-5f33e07b1e57, Retrieval Date 12 (2018), p. 2019
* [12] Balachandran Vijayalakshmi et al. “An attention-based deep learning model for traffic flow prediction using spatiotemporal features towards sustainable smart city”. In : International Journal of Communication Systems 34.3 (2021), e4609.
* [13] Wangyang Wei, Honghai Wu et Huadong Ma. “An autoencoder and LSTM-based traffic flow prediction method”. In : Sensors 19.13 (2019), p. 2946.
* [14] Jason Brownlee. 14 Different Types of Learning in Machine Learning. https : / / machinelearningmastery.com/types- of- learning- in- machine- learning/. Accessed : 2021-03-24. 2019.
* [15] Yuan, T., Rocha Neto, W., Rothenberg, C., Obraczka, K., Barakat, C. et Turletti, T. 2021. Apprentissage automatique pour les systèmes de transport intelligents de nouvelle génération : une enquête. Transactions sur les technologies de télécommunications émergentes, 33(4).
* [16] Yin, X., Wu, G., Wei, J., Shen, Y., Qi, H. et Yin, B., 2022. Apprentissage en profondeur sur la prévision du trafic : méthodes, analyse et orientations futures. IEEE Transactions on Intelligent Transportation Systems, 23(6), pp.4927-
* [17] Puiu, Dan & Barnaghi, Payam & Tönjes, Ralf & Kumper, Daniel & Ali, Muhammad Intizar & Mileo, Alessandra & Parreira, Josiane & Fischer, Marten & Kolozali, Şefki & Farajidavar, Nazli & Gao, Feng & Iggena, Thorben & Pham, Thu-Le & Nechifor, Cosmin-Septimiu & Puschmann, Daniel & Fernandes, Joao. (2016). CityPulse: Large Scale Data Analytics Framework for Smart Cities. IEEE Access. 4. 1086-1108. 10.1109/ACCESS.2016.2541999.
* [18] Vehicle Traffic, Provided by City of Aarhus in Denmark (http://iot.ee.surrey.ac.uk:8080/datasets.html)
* [19] Sefki Kolozali, Maria Bermudez-Edo, Daniel Puschmann, Frieder Ganz, Payam Barnaghi, "A Knowledge-based Approach for Real-Time IoT Data Stream Annotation and Processing", in Proc. of the 2014 IEEE International Conference on Internet of Things (iThings 2014), Taipei, Taiwan, September 2014.
* [20] R. Tönjes, P. Barnaghi, M. Ali, A. Mileo, M. Hauswirth, F. Ganz, S. Ganea, B. Kjærgaard, D. Kuemper, S. Nechifor, D. Puiu, A. Sheth, V. Tsiatsis, L. Vestergaard, "Real Time IoT Stream Processing and Large-scale Data Analytics for Smart City Applications", poster session, European Conference on Networks and Communications 2014.
* [21] Singh, A. (2019). SMART CITY With IOT and BIG Data. SSRN Electronic Journal. https://doi.org/10.2139/ssrn.3405839
* [22] Starovoitov, V. V., & Golub, Y. I. (2021, September 30). Data normalization in machine learning. Informatics, 18(3), 83–96. https://doi.org/10.37661/1816-0301-2021-18-3-83-96
* [23] Yang, C. H., Park, S. H., Kim, J. G., & Lee, J. K. (2022, June 7). Traffic operation analysis for underground and ground roads using microscopic traffic simulation. SIMULATION, 98(11), 1071–1082. https://doi.org/10.1177/00375497221099545
* [24] Akashi, F., Bai, S., & Taqqu, M. S. (2018, March 26). Robust Regression on Stationary Time Series: A Self-Normalized Resampling Approach. Journal of Time Series Analysis, 39(3), 417–432. https://doi.org/10.1111/jtsa.12295
* [25] Sang, C., & Di Pierro, M. (2019, March). Improving trading technical analysis with TensorFlow Long Short-Term Memory (LSTM) Neural Network. The Journal of Finance and Data Science, 5(1), 1–11. https://doi.org/10.1016/j.jfds.2018.10.003
* [26] A Smart Traffic Approach. (2022, December). Traffic Technology International, 2022(6), 58–58. https://doi.org/10.12968/s1356-9252(23)40596-5
* [27] Ataş, M., Yeşilnacar, M. R., & Demir Yetiş, A. (2021, November 5). Novel machine learning techniques based hybrid models (LR-KNN-ANN and SVM) in prediction of dental fluorosis in groundwater. Environmental Geochemistry and Health, 44(11), 3891–3905. https://doi.org/10.1007/s10653-021-01148-x
* [28] Chen, H. (2023, May 11). Prediction And Analysis Of Population Aging In Eight Ethnic Provinces Based On Machine Learning (SVR). Highlights in Science, Engineering and Technology, 47, 183–193. https://doi.org/10.54097/hset.v47i.8200
* [29] Electrical Load Forecasting using Machine Learning Methods, RNN and LSTM. (2020, April 18). Journal of Xidian University, 14(4). https://doi.org/10.37896/jxu14.4/160
* [30] Simulating Traffic Flow in Python. https://towardsdatascience.com/simulating-traffic-flow-in-python-ee1eab4dd20f