International Journal of Science and Research (IJSR)
ISSN: 2319-7064
SJIF (2022): 7.942

# Traffic Control and Management System for Smart City Using Geo-Social Networks

**M. R Sumalatha[1], Lakshmi Harika Palivela[2], Srimathi Ravisankar[3], Kanimozhi Mahendran[4], Srilakshmi A[5]**

Department of Information Technology, Anna University, India

**Abstract:** ***Traffic prediction is vital in smart city traffic management and control system because of the ever-changing nature of the Traffic. Route planning and smart city traffic congestion reduction benefit from long-term traffic predictions. This challenge is difficult to solve in long-term traffic forecasting because of the complex and dynamic spatio-temporal interactions between various components of the road network. Traffic forecasting in smart cities is necessary for traffic management and public safety since it is impacted by a range of factors such inter-regional travel, activities such as social events, tolls, and VIP gatherings and weather. This novel model proposes a Spatio-Temporal Residual Neural Network model with External Fusion for Traffic Prediction System to forecast inflow and outflow, to handle temporal and spatial dependency, and to model the temporal properties and proposed a Graph Encoder and Decoder and Attention layer is introduced between the encoder and decoder. To decrease error propagation, transformation attention layer is inserted between the encoder and decoder. Traffic Monitoring system is modelled with SARIMA model to monitor and handle the seasonality in data. The Traffic congestion control and management is developed using Angular Fire and Firebase Database in which Location, Intensity of Traffic, Traffic Type is inputted and alerted to the drivers which will help them plan the route in advance and aid them prevent last minute hassle and the events is classified occurring before the accident and predict the traffic event and intensity by which will cause the traffic and prevents traffic jam. Users may extend their assistance by providing any updates they get in their surroundings, which will aid peer travelers in accommodating traffic and predicting if severe traffic jams will occur in the near future based on traffic incidents.***

**Keywords:** Traffic Prediction, Smart City Traffic Management, Data Analytics, Geo-social Networks

## 1. Introduction

### 1.1 Overview & Problem Statement

In an intelligent transportation system, traffic prediction is critical. Accurate traffic predictions may help with route design, vehicle dispatching, and traffic congestion reduction. The smart city is rapidly altering the modern metropolis. The rate of urbanization, as well as the fast expansion of the urban population, puts a strain on city traffic management. The city's population and the number of automobiles on the road are both fast growing. As the urban population and hence the number of automobiles has expanded, controlling streets, highways, and roads in densely populated metropolitan areas has become a critical challenge. The fundamental reason of today's traffic congestion is the traffic control measures used. Traffic management solutions now pay little attention to real-world circumstances that affect traffic dynamics, resulting in inefficient traffic management. This work, which aims to avoid excessive traffic jams, was built using deep learning [1, 2]. The growing number of cars using road junctions has resulted in issues such as traffic congestion, conflicts, and bottlenecks. The only way to overcome these issues presently is to provide effective traffic management at crossings. The management and mitigation of traffic congestion is another major difficulty for urban planners. Due to this increase in traffic, long queues of vehicles form at junctions, leading commuters to miss valuable time, especially during rush hours and peak times.

The objective of this work is to develop a traffic monitoring system which predicts the Traffic in road networks by considering Geographical structure of road, Impact of traffic dynamics and social factor influence. Also to develop a system that also handles the problem, by forecasting Traffic congestion and aids in vehicle re-routing and advanced planning in road networks.

***Motivation***

The Intelligent Transportation System (ITS), which is a crucial component of a smart city, includes traffic prediction. Many real-world applications need accurate traffic predictions. Given the rise in vehicle line lengths waiting to be processed at intersections, increased traffic flow, and increased traffic congestion in metropolitan areas, traffic congestion must be monitored and managed. Traffic spatiotemporal sequence data is influenced by weather, social events, and road networks. The current system does not mine the spatial-temporal externally defined patterns, this work focuses on Considering geographical road structure and social factors impacting the traffic dynamics to be able to forecast traffic and plan the route accordingly. This work focuses on considering geographical road structure and social factors impacting the traffic dynamics to be able to forecast traffic and plan the route accordingly.

***Challenges***

The following elements have an impact on traffic prediction:
1) Spatial Temporal Traffic Data: Data on spatial and temporal traffic is subject to continual changes in both place and time and is subject to intricate and dynamic spatiotemporal relationships.
2) External influences: The traffic spatiotemporal sequence data are affected by external variables such as weather, events, and route characteristics.
3) The dynamic and complicated traffic situation in big cities
4) Unexpected traffic incidents (VIP Tolls, Road Blocks)

***Scope***

The work's goal is to create a prediction module that can be used to efficiently regulate and manage traffic in a specific zone depending on the dynamics. Early detection of traffic jams is possible using traffic prediction, allowing traffic authorities to take proactive actions to relieve road congestion. Traffic prediction, on the other hand, is a key step in providing commuters with realistic travel time predictions. Through a geo social network, deliver real-time traffic reports on traffic congestion and odd traffic events. Citizens will benefit from early-warning messaging and travel planning, especially during busy hours.

***Objective***

To design a Traffic Prediction system which will consider Geographical and social attributes involved in Traffic dynamics and sub structure of Road networks. To create a method for predicting traffic flow that simultaneously accounts for spatial and temporal interdependence in urban road networks. To predict Traffic congestion and manage Traffic flow in particular region and aid in advanced route planning.

***Application***

Traffic flow forecasting can assist the city in reducing congestion. Car-sharing firms may employ demand forecasting to pre-allocate vehicles to high-demand regions. Aid in advanced route planning and vehicle rerouting to decrease commuting routes, traffic forecasts to spot bottlenecks early, and better-informed judgments to handle these problems. It will assist drivers in making more informed travel decisions, reducing traffic congestion, lowering $CO_2$ emissions, and increasing transportation efficiency.

***Time series analysis***

A time series is a group of data points that emerge throughout time in a certain sequence. One of the two main activities of supply chain management is demanding management (DM) and the other is central planning. An essential component of DM is the assessment of future demand.

**Components of Time Series Analysis:**

**Trend**: With no defined intervals, a trend is a continuous chronology of irregularities in a particular data collection. Negative, positive, or zero trends are all possible.

**Seasonality**: Seasonality is defined as movement in a data collection on a continuous timeline at regular or fixed periods [3]. Serrations or bell-shaped curves will be present.

**Cyclical:** There are no set intervals or motions, and the nature of these movements is unknown.

**Irregularities**: Unexpected situations/events/scenarios and explosions in a brief period are examples of irregularities.

The goal of forecasting approaches is to uncover patterns in historical data that assist explain demand (trend up or down, seasonal (cyclical), leaps, and autocorrelations), and then utilize these patterns to estimate future demand under the assumption that the past is predictive of the future. Time series decomposition is the process of dissecting a sequence into its level, pattern, periodicity, and noise components. Decomposition is a valuable conceptual paradigm for better comprehending time-series data in general, as well as the challenges that can occur when planning and forecasting time-series data [4]. In time series analysis, stationarity is a key property. If a time series' statistical properties do not change over time, it is said to be stationary. i. e., the covariance does not remain constant over time, but the mean and variance do. Data with a high trend or seasonality is referred to as "non-stationary data. [5]" Seasonality refers to swings that occur on a regular basis. A time series is said to have a seasonal pattern when seasonal factors like the season or day of the week have an influence on it. Seasonality has a known and observable regularity. A trend is defined as a long-term increase or decrease in statistics. It doesn't have to be in a straight line. The changes in the upward and downward direction are called as Trend. A trend is said to be "changing direction" when it shifts from an upward to a downward trend.

***Augmented dickey fuller test* (ADF)**

The ADF Test (Augmented Dickey-Fuller Test) is a statistical method for determining the stationary state of a time series. It's one of the most used statistical tests for assessing whether or not a series is stationary. The Dickey-Fuller Test has been upgraded and is now known as the Augmented Dickey-Fuller Test. It assesses the relationship between the expected Y value and time-lag factors, as well as lagged differencing terms, as shown below. Seasonality or a propensity may typically be eliminated by differentiating. Apply the Dickey-Fuller test to check if a regression model exists. Because $p>0$, the process is not stationary in this example. If $p=0$, the null hypothesis is rejected, but if $p > 0$, the alternative hypothesis is accepted.

***Encoder – decoder architecture***

An encoder and a decoder are two of the model's two sub-models.

- Encoding: The encoder is in responsible of encoding the whole series into a corrected context vector after processing each of the input time steps.
- Decoding: The decoder is in responsible of taking from the context vector while the output time stages are iterated through.

Recurrent neural networks are organized for sequence-to-sequence prediction issues using the encoder-decoder paradigm [6]. A network that encodes, extracts, or gets information from input data is known as an encoder. The input sequence is scanned, and the data is condensed into internal state vectors, also referred to as context vectors. Typically, the model just maintains track of the internal states and disregards the encoder's outputs. This context vector aims to contain all the information for all input items to help the decoder make accurate predictions. The hidden and cell states of the network are sent into the decoder as input. A decoder decodes and interprets the context vector supplied from the encoder. The context vector is received by the first cell of the decoder network from the encoder's final cell. The decoder builds the output sequence from these beginning states, and it also uses these outputs to anticipate the future. At a specific time, step t, each LSTM unit in this stack forecasts an output. Every recurrent unit gets a hidden state from the preceding unit and generates an output in addition to its own hidden state for network propagation. A potential difficulty with this encoder-decoder method is that a neural network has to be able to fit all relevant data from a given phrase into a corrected vector. The neural network may have trouble processing large phrases, especially ones that are lengthier than those from the training corpus.

***Attention mechanism***

One of the network's primary flaws is its inability to extract significant contextual links from seq2seq models, which reduces the model's performance and, eventually, accuracy. Attention is a step forward from the previous network of sequence-to-sequence models that solve this issue. It can deduce meaning from sequences. It functions by providing the decoder with a learning mechanism that helps it determine which encoding network to concentrate on when anticipating outputs at each time step in the output sequence. This context is sent from the encoder to the decoder in a more weighted or indicated manner. This is accomplished by keeping the intermediate outputs from each step of the input sequence of the encoder LSTM network, which correlate to a particular degree of relevance, while concurrently training the model to learn and pay selective attention to these intermediate components, and then linking them to elements in the output sequence.

***Resnet Unit***

In a CNN, the mistake rate rises. In this architecture, the vanishing/exploding gradient issue is resolved using the residual network. A method known as skip connections is applied in this network. The skip connection links straight to the output after skipping a few training stages. Instead of letting the layers learn the underlying mapping, instead let the network fit the residual mapping. Instead of using H (x) as the initial mapping, use F (x): = H (x) – x, which provides H (x): = F (x) + x. As a consequence, problems brought on by vanishing/exploding gradients can be avoided while training very deep neural networks. The term "artificial neural network" (ANN) refers to a particular type of ANN called the residual neural network (ResNet) (ANN). Another use is the Control Neural Network. It's an open-gated or gateless version of the Highway Net, the world's first operational very deep feed forward neural network with hundreds of layers, far more than previous neural networks.

***Traffic prediction***

Traffic prediction is the technique of forecasting real-time traffic information using historical traffic data, including such traffic flow, average speed, and incidents, as well as floating automobile data. People lose significant time, money, and patience due to traffic congestion every day. Governments are impacted by high traffic since they are required to manage traffic flow for the delivery of goods, the reduction of pollutants in certain areas, and the safety of people on the road. People at all layers of society are impacted by congestion, which is a global problem. The most common causes of traffic congestion are well known to every driver who has experienced a delay in traffic on their route. Traffic problems, such as accidents and construction, regularly result in unforeseen delays. Bad weather also contributes to low traffic flow rates, and inefficient traffic signal timing reduces capacity on smaller inner roads. There are too many cars for the world's limited road network, though, and this is what is causing the greatest increase in traffic congestion. Governments may also gain from extremely accurate traffic statistics by encouraging the creation of intelligent and environmentally friendly transportation solutions to reduce overall levels of congestion. Possible data sources include automotive partners, smart phone partners, well-known ride-hailing and navigation applications, smart phone partners, fleet tracking systems, and GPS probes from connected GPS systems and smart phone apps.

***Advantages of traffic prediction***

Individual travelers, City Corporations, and government agencies all require current traffic flow information that is accurate and timely. Traffic prediction will aid drivers in making travel choice and plan routes in advance, lessen traffic congestion, cut down on carbon emissions, and increase the effectiveness of traffic operations in Urban areas where Traffic volume is enormous. It is considered as being a crucial element for the effective deployment of ITS components such advanced traveler data systems, advanced traffic control systems, advanced systems for public transit, and advanced systems for commercial vehicle operations. Traffic flow prediction makes use of inductance loops, radar systems, cameras, mobile Satellite Tracking Systems, crowd funding, social networks, and other sensor sources in addition to previous and real-time traffic data [7]. The amount of traffic data has increased as a result of the extensive use of both conventional and newer traffic sensor technologies.

***Firebase and angular fire***

The official Firebase Angular binding is called Angular Fire. Google Firebase Database and Angular Fire were used to create the database for the traffic streams. A cloud database that saves data in JSON format is called the Firebase Realtime Database. Every client connected to the server receives real-time data synchronization. Clients hold a common Real-time Database server and get real-time data updates while creating cross-platform apps using the iOS and JavaScript SDKs. A NoSQL database called the Firebase Realtime Database holds and instantly syncs data among all of our users. It's a sizable JSON object that programmers may interact with in real time. The Firebase database includes the app with both the information's existing value and any changes to it over a single API. Real-time synchronization allows our customers to access the data from any platform, including the web and mobile. The Realtime database enables our users to cooperate with one another. We may create our app without using any servers thanks to the inclusion of mobile and web SDKs. The Real-time Database SDKs employ the device's local cache to give and maintain updates when our users go offline. When the device is linked to the internet, the local information is automatically synced.

## 2. Literature Survey

Luiz Fernando et al. [8] suggested a traffic light system's operational state was reported to central management, and a control system was built for unusual circumstances such road closures brought on by accidents or public events. This technology may be used in many works to develop algorithms that reduce the time it takes for traffic lights to turn on and off or to synchronize subsequent pieces of data in real-time. However, this system lacks an algorithm for altering the phase of traffic lights as well as sensor-based vehicle flow monitoring.

Dominil Cvetek, et al. [9] proposed the ARIMA model for predicting traffic using historical time series. The data-driven approach can be beneficial to inform drivers about better route selection and provides a guide for corporate strategy traffic planning, but the model can only predict a single point on the road map, making it unsuitable for uncertain circumstances and missing the peak as well as rapid fluctuations.

Chuanpan Zheng et al. [10] integrated discrete regional disturbances brought on by a variety of area functions with spatio-temporal propagating effects (STD). A two-phase, end-to-end deep learning architecture called DeepSTD uses STD to predict citywide traffic flow. The restrictions are It is impossible to forecast how traffic would behave during unusual and important incidents like serious accidents during rush hour.

Zhiyuan Wang et al. [11] developed the updated SqueezeNet model in conjunction with the MOG2 (Mixture of Gaussians) noise removal approach. (H-SqueezeNet). This is suitable for real-time systems. Therefore, care should be used while choosing the vehicle classification. Since this method works so well, it may be used to systems for smart cities and intelligent traffic monitoring, for example. We do not anticipate doing a traffic density survey or identifying license plates. The mechanism for smart cities is still not fully functional.

Zhigaung Cao, et al. [12] proposed a Multiagent Pheromone-based traffic control system. Combine vehicle rerouting with signalized intersections management to relieve traffic congestion. To reduce traffic congestion, two pheromone-based methods for dynamically managing traffic lights were created. It is not anticipated to analyze the performance on some other complicated road networks with different car densities.

Haifeng Zeng, et al. [13] introduced a deep learning model that employs bidirectional LSTM module and hybrid and multilayer architecture to extract daily / monthly periodic information. A potential substitute for conventional CNN-LSTM broadband applications for traffic flow prediction in complicated systems is a family of attn graph neural networks. The use of conventional hybrid networks to predict traffic in extensive road networks is not explored.

Yadong Yu, et al. [14] presented a low rank compelling aspect decomposition model that forecasts traffic flow using a phase transition matrix. This illustrates the relationship between temporally neighboring traffic flow segments with low rank regularization. This model is not as reliable as the other deep learning techniques and also organizes traffic data for various road sections in a vectorized way without utilizing the deeper relationships between the sections.

Zhishuai Li et al. [15] introduced a multi stream feature fusion method that creates graphs that used an information neighboring matrix instead of a way away matrix to extract and combine rich attributes from traffic data. The technique has the drawback of requiring additional training parameters for the fine-tuning procedures in the neighboring matrix. As additional monitor stations are added to the road sensor network, the network will clog up and training effectiveness will drop.

Xiaoming Shi et al. [16] proposed a unique Attn periodic Spatiotemporal CNN Model, an end-to-end represents the expected system that incorporates spatial, short term and long-term periodical relationships [28]. The first method represents both spatial and monthly dependency via an encoder attention mechanism. The model requires more data to operate properly since it contains more parameters than earlier baselines, which might be a disadvantage.

Yuanli Gu et al. [17] suggested a deep learning-based Bayesian combination model. The IBCM framework was created to deal with the problem of multiplication mistakes. To determine how recent traffic flow corresponds to present traffic flow, correlation analysis is utilized. Weather, traffic accidents, speed, and occupancy are just a few more examples of the various forms of data that are not taken into account.

Zhengang Guo et al. [18] developed the CPS model to represent the current status and dynamic behavior of road segments and vehicles by merging short-term traffic forecasting with real-time routing optimization [19]. To extract previous knowledge from the acquired data, an online learning data-driven model incorporating model learning and parameter learning is built based on the online traffic data collected from road segments and real-time vehicular data obtained from automobiles. For short-term traffic forecasting and in-the-moment routing optimization, other models and techniques are not taken into account or investigated.

Zhongda Tian et al. [20] proposed a novel short-term time series forecasting approach based on observable mode decomposition and combination model fusion. In addition to looking into the amplitude-frequency characteristics of short-term traffic flow series and using empirical decomposition to break down traffic flow into several components with different frequencies. An enhanced fruit fly optimization method is put forth to optimize the weight coefficient of the combination model. Prediction We'll also look into other circumstances, such passenger flow and traffic forecasting in workplaces.

Yang Liu et al. [21] introduced DNN and examined the passenger flow from scopes at both the macroscopic and microscopic levels in order to tackle the complex bus passenger flow prediction problem. The shortcomings and distinguishing qualities of decision-tree based models are then examined. The results of the research can be used to improve the architecture of the deep learning network.

employing a decision-tree-based method to model and predict passenger flow. Data from public transportation does not forecast the traveler behavior patterns.

Dongfang Ma et al. [22] proposed a new deep learning-based approach for daily road traffic forecasting that takes contextual factors and traffic flow patterns into consideration. The traffic flow recorded by certain surrounding detectors is associated with each other, therefore the suggested approach cannot be extended to anticipate traffic flow for multiple detectors and further parameters were not added. Contextual elements such as hourly weather and change were not adequately worked during the day.

Jinlei Zhang et al. [23] suggested a deep learning architecture made up of the residual network size, graph convolutional network, and long-short term memory. While network topology data is retrieved using GCN, temporal correlations are extracted using the attention LSTM. Network topology data is extracted using GCN, while temporal correlation data is extracted using attention LSTM. GCN is utilized to extract network topology data, attention LSTM is used to extract temporal correlations, and ResNet is used to capture deep abstract spatial correlations between subway stations. Because the suggested model is a "black box, " where data are provided to produce accurate predictions without revealing the applied technique, model interpretability is low. Xianwei Meng, et al. [24] proposed a long short-term memory with total variation warping (D-LSTM) model that includes a Dynamic Time Warping Algorithm that really can fine tune the time feature, altering the current data distribution to be close to the prior data. Even without particular holiday treatment, the refined data can still significantly improve. Meanwhile, because the data under different feature distributions has varying effects on the prediction outcomes, the model includes an attention mechanism. The disadvantage is that the DTW Algorithm is inefficient. The model's prediction effect isn't very strong.

Kan Guo et al. [25] proposed in their dynamic graph convolutional approach for traffic forecasting. A latent network is used to extract spatial-temporal information for dynamically creating dynamic road network graph matrices. The recommended method is put to the test using three sets of actual traffic statistics. According to the experimental results, the recommended method works better than cutting-edge traffic forecasting technologies. A quick and effective way to dynamically abstract Laplace matrices is essential for road network forecasting on a wide scale.

Shuoi Xu et al. [26] created Venue2Vec, a brand-new embedding model for pinpoint user position prediction. It automatically combines sequential connections, semantic information, and temporal-spatial context. Similar locations, those adjacent physically or those are often visited by individuals, will all be grouped closer together inside the embedding area [27]. The influence of a set of friends or community on location prediction is not taken into account.

Cheng Te Li, et al. [28], who also connected crowd simulation with social network analysis, demonstrated that it is possible to tackle challenges from several sides. This discovery resembles the growth of a real civilization or community in certain ways. The creation of more precise or effective social network construction and link detection technology is hampered by ineffective management.

Binbing Liao et al. [29] established a method for enhancing traffic prediction. Include in the mix offline geographical and social elements, information about road intersections, and online crowd searches. By including three categories of implicit yet crucial factors kept in auxiliary data. Testing using both qualitative and quantitative methods on a real-world dataset has demonstrated the framework's applicability. Real-time traffic forecasting is not taken into account and the limitations are exact. Bruna et al. [30] were the first to build a spectral network that computed the eigen factorization of the network Laplacian matrix to execute convolution operations on graph data in the spectral domain B.

Yu et al. [31] characterized it as a generic graph, where the monitoring stations in a traffic network represent the graph's nodes instead of considering the traffic network as a grid, this is a logical and natural approach to construct the road network, with the connections between stations serving as a representation of the edges and the distances between stations being used to create the adjacency matrix. The computational complexity was then decreased by employing two graph convolution approximation methodologies based on spectral techniques to extract patterns and features in the spatial domain.

X. Geng et al., [32] s use of graphs allowed them to show various connections between locations, including those based on neighborhood, functional similarity, and transportation connection. After that, three sets from GCN based on ChebNet were utilized to model regional correlations, and further combining temporal data led to the development of a traffic demand estimate.

Y. Li et al. [33] characterized the traffic network as just a directed graph, capturing the dynamism of the traffic flow based on the diffusion. The spatial connection is then modelled using a diffusion convolution process, which has a more natural interpretation and works well for representing spatial and temporal relationships. Particularly, diffusion convolution reflects the bidirectional diffusion, enabling the model to take into consideration both downstream and upstream traffic.

S. Fang et al [34] used dilated causal convolution as the temporal convolution layer of their models. Convolutions, unlike recurrent models, construct representations for fixed-size contexts; however, the network's effective context size may be readily increased by stacking numerous layers on top of each other. This allows the maximum length of dependencies to be modeled to be carefully controlled. Because the convolutional network does not rely on the previous time step's processing, each element in the sequence may be parallelized, making greater use of GPU power and making optimization easier. RNNs, on the other hand, keep the complete hidden state of the past, forbidding concurrent calculations in a series.

Tampubolon et al. [35] demonstrated hyper parameter tuning techniques to optimize the networks and yield better results to the generalization and over-fitting problem. Stochastic Gradient Descent was utilized to determine the gradient and update the weights using a back propagation approach (SGD). A dropout layer was added and batch normalization was done to ensure that the input distribution is standardized before passing the activations to the nonlinear layers.

Y. Liang et al. [36] suggested RNN-based methods for traffic prediction. RNN and its variants, LSTM or GRU, are multi-layer perception for analyzing sequential data. The non-linear temporal dependence of traffic data has been modeled using RNN-based methods for traffic prediction. Since these models depend on data order to manage data in order, one drawback is that when modeling lengthy sequences, their ability to recall what they have learned in previous time steps may suffer. A unique network structure called an encoder-decoder has been used in RNN-based sequencing learning for the prediction of traffic. The fundamental idea is to decode the original sequence into a fixed-length vector and then build the forecast from it. The length of the decoder is one potential flaw in the encoder-decoder system. The other works help in processing the data sets various learning models and providing the security using block chain [37].

***Summary***

Therefore, the literature review discusses different methods, methodologies, and disadvantages of different methods of Traffic Monitoring and prediction models. Current ARIMA forecasting methods cannot handle seasonality well, so we choose SARIMA Model for Traffic forecasting because it can handle (Annual, Weekly and daily seasonality) and capture the trend in the data effectively. CNN can handle spatial structure information well. The spatial relationships between any two places in the city must be captured. The more layers there are, the more difficult it is to train them, and residual learning can successfully handle this problem, whereas the current approach ignores weather and social gathering occasions. By incorporating the Attention layer into the encoder-decoder structure, creating a fusion model with Resnet would effectively forecast traffic under any given scenario and overcome the disadvantage of the encoder-decoder model.

## 3. System Design and Architecture

Individual travelers, businesses, and government organizations all need accurate and fast traffic flow information right now [38]. It has the potential to help drivers make better travel decisions, reduce traffic congestion, reduce carbon emissions, and improve traffic operations efficiency. The purpose of traffic flow prediction is to provide such traffic flow information. Traffic flow prediction has grown in popularity as a result of the rapid development and implementation of intelligent transportation systems (ITSs). It is acknowledged as an essential component for the successful deployment of ITS subsystems such as advanced traveler information systems, advanced traffic management systems, advanced public transportation systems, and commercial vehicle operations. In traffic flow prediction, historical and real-time traffic data from various sensor sources, including as inductive loops, radars, cameras, mobile Global Positioning Systems, crowd sourcing, social media, and so on, is commonly employed. As a result of comprehensive traditional traffic sensors with new developing traffic sensor technologies, traffic data is exploding, and we have reached the era of Big Data transportation. In transportation monitoring and administration, data is becoming increasingly vital. Weekends, weekdays, and varied traffic dynamics in different regions of the city all have an impact on traffic. On weekends, there is a large increase of people near malls and parks compared to weekdays, while traffic near schools follows a pattern on weekdays based on school and college schedules. Understanding and factoring in peak hours, weather, and holidays are critical in forecasting and managing traffic effectively. As a result, we present a model that externally integrates the Holiday and weather factors with the Resnet Unit and an Attention-based Encoder-Decoder model to overcome the disadvantage of Encoder-Decoder Architecture by successfully reducing information loss for traffic prediction. The SARIMA model has been proposed for traffic monitoring, and a web app built with Firebase and Angular Fire has been presented for a Traffic Management system that uses GPS to inform users to traffic incidents such as road blocks, VIP tolls, and accidents. The architecture is shown in Fig 1.

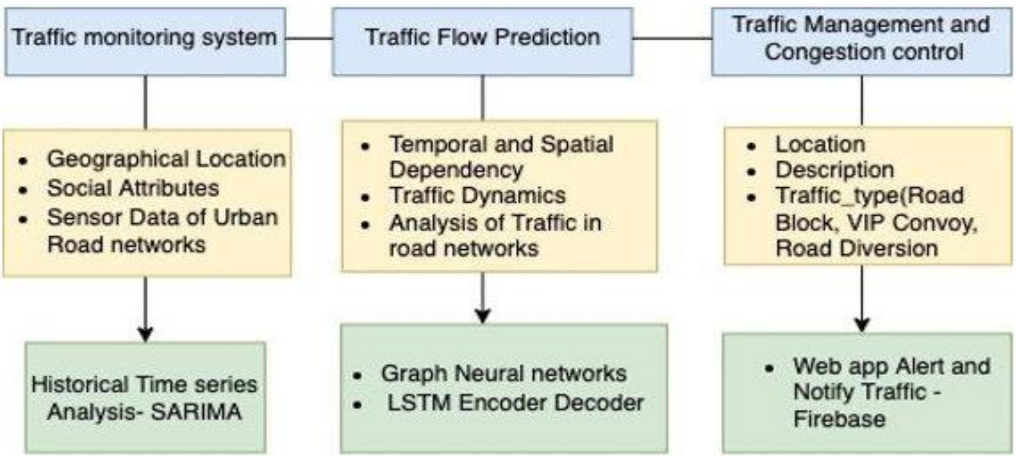


**Figure 1:** Block diagram of the proposed work

The System architecture composes of three modules

International Journal of Science and Research (IJSR)
ISSN: 2319-7064
SJIF (2022): 7.942

1) Traffic monitoring system – SARIMA which observes the Daily Trend, Weekly Annual Seasonality in the Traffic.
2) Traffic flow prediction system — based on two algorithms, predicts the predicted traffic flow in a region.
   a) Attention based Encoder-Decoder for Spatio-Temporal Traffic Prediction
   b) Deep Residual Network Spatio – Temporal with External Fusion
3) Traffic congestion control system-Web app based on Firebase and Angular fire Alert system for drivers to notify them of Road Block, VIP tolls and Accidents.

***Traffic Monitoring System:***

**Drawbacks of existing model**

1) ARIMA Model is generally designed for small datasets, and are not suitable to deal with complex and dynamic time series data.
2) The ARIMA model struggled to deal with seasonality.
3) Neural networks weren't adequate to the task of dealing with a large number of category variables and structural data; they were too expensive, time-consuming, and difficult to large volume of Traffic Data

**Proposed model**

SARIMA Model is proposed to model the Daily Trend, Weekly and Annual Seasonality (Peak hours and Weekdays) to monitor the Traffic Trend in a region

**SARIMAX algorithm:**
**(Seasonal Auto Regressive Integrated Moving Average)**

SARIMAX is more specialized than ARIMA in dealing with seasonal peaks in demand, such as those associated with holidays or festivals. In time series modeling we encounter cases where there is an external factor which can influence the outcome for a particular time period. These extraneous variables or regressors might be regarded as such. SARIMA uses a set of annual parameters (P, D, Q) m to describe the seasonal elements of the model. The number of data (rows) in each seasonal cycle is m, and the moving average, differencing, and seasonal coefficient of determination are P, D, and Q, respectively.

$$\phi_p(L)\tilde{\phi}_P(L^s)\Delta^d\Delta_s^D Y_t = A(t) + \theta_q(L)\tilde{\theta}_Q(L^s)\epsilon_t$$

*Eqn 1-Sarimax Algorithm*

$\phi_p(L)$ *is the non – seasonal autoregressive lag polynomial*

$\tilde{\phi}_P(L^s)$ *is the seasonal autoregressive lag polynomial*

$\Delta^d\Delta_s^D Y_t$ *is the time series, differenced d times, and seasonally differenced D times.*

$A(t)$ *is the trend polynomial (including the intercept)*

$\theta_q(L)$ *is the non – seasonal moving average lag polynomial*

$\tilde{\theta}_Q(L^s)$ *is the seasonal moving average lag polynomial*

**P**: Seasonal auto regressive order
**D**: Seasonal difference order
**Q**: Seasonal moving average order
**m**: A single seasonal period's number of time steps.
Represented as **(p, d, q) × (P, D, Q) m.**

***Traffic Prediction System:***

**Drawbacks of existing model**

1) RNN-when modelling lengthy sequences, their capacity to retain knowledge acquired prior to several time steps may deteriorate.
2) Encoder-decoder structure – When the input is too long, some data will be lost since the length of semantics vectors between encrypting and decrypting is always fixed, regardless of length of the incoming and outgoing sequences.
3) CNN-Effectively handles spatial dependence. More layers, higher error values
4) The current model does not take the weather or holidays into consideration when predicting traffic.

**Proposed model:**

**External fusion of a deep residual network for spatio-temporal traffic prediction**

1) Convolution-based residual networks may explain nearby and distant spatial relationships between any two sites in a city while ensuring that the deep structure of the neural network does not undermine the model's prediction accuracy.
2) Temporal proximity, period, and trend are the three categories that may be used to categories the temporal aspects of population movements. Three separate residual networks are used by ST-ResNet to mimic these properties.
3) Aggregates the output of the three networks dynamically, weighting various branches and areas differently.
4) Fusion with external variables is the aggregation (Weather and Social Holidays).

**Attention based encoder for traffic prediction**

To change past traffic characteristics into future representations, a transform attention method is used. This attention mechanism reduces the problem of mistake propagation by modelling direct linkages between past and future time steps. The decoder foretells the output sequence after the encoder has coded the input traffic information. A transform attention layer is utilized between the encoder and the decoder to transform the encoded traffic characteristics into serial representation of future timeframes as the decoder's input. A stack of Attention bricks is used to construct both the encoder and the decoder. A space-time focus mechanism for modelling non-linear temporal correlations, a spatial attention method for modelling dynamic spatial correlations, and proposed spatial and temporal attention approaches are all included in each Attention block.

**Spatial Attention:**

Other roads have varying degrees of effect on a route's traffic situation. This kind of effect is very fluid and shifts throughout time. To imitate these traits, we develop a spatial attention approach that adapts to the correlations between sensors in the road network. Various weights are dynamically allocated to different vertices at different time steps (e. g., sensors).

**Temporal Attention:**

The present traffic condition at a point is linked to previous data, and the correlations change non-linearly with time. To mimic this, a temporal attention strategy was utilized to adaptively replicate the non-linear relationships between different time steps. Both the traffic circumstances and the related time context have an impact on the temporal correlation. Congestion that occurs during morning rush hour can cause traffic to be disrupted for several hours. To quantify the importance across distinct time steps, we take into account both traffic characteristics and time. We employ a multi-head approach to compute the attention score by concatenating the hidden layer with the spatiotemporal embedding.

**Dependencies in traffic prediction:**
The model is designed to incorporate the following parameters in the network to
1) Integrate the **Socio-Geographical parameters** in the network
- Social Attribute (Holiday)
- Geographical Attribute (Weather)
- Local road network

**2) Spatial Dependency: (*Between Regions*)**
The traffic in one location is impacted by the traffic in the other. The inflow of region 2 is influenced by the outflow of neighboring regions (1), as well as other locations. The outflow from region 2 would have an influence on inflows from other locations (e. g., region 3). Region r2's input would have an effect on its own outflow.

**Temporal Dependency:**
The movement of people in an area is influenced by recent time intervals, both local and remote. For example, traffic at 8 a. m. will affect traffic at 9 a. m., Morning rush hours, Weather-related Traffic Influence (Winter and summers).

**Traffic Management System:**

**Drawbacks of existing model**
The existing system predicts the shortest route from Source to Destination and Highlight the Traffic in map.

**Proposed model**
The proposed model helps people to know the traffic details from the current location to the desired destination address. Alerts the occurrence of traffic incidents like accidents, traffic issues, cop traps, VIP Tolls and Road blocks in way to destination. Tell the user few minutes before he leaves, that the usual route he takes will lead him to jam and help him to choose a better route. This web app has been built by the Angular Fire and Firebase to Add Alert and show Alert to the user. There are two Entities in the System
1) Admin (City Traffic Corporation)
2) Citizens

Functionalities provided by the System
1) Add Alert
2) View Alert
3) View Alerts in a Particular Region

Updates or breaking news on any issue that may cause a delay in reaching your destination [39]. It might be with the aid of traffic coordinating authorities who use the handling website to publish live alerts regarding traffic jams, road blockages, diversions, VIP transport, rallies, construction alerts, and so on, which are displayed on the app displays of those who are using the app. Users may expand their assistance by sharing any updates they get in their immediate region, which helps to cover a greater area. To avoid bogus news and updates, these are first examined by the relevant authorities via the admin panel, and then published on the app after proper verification.

***Challenges in long term traffic prediction***
Long-term traffic forecasting is difficult due to the complexity of transportation infrastructure and the dynamic nature of numerous influencing factors.
- Sensitivity to error propagation -In the long-term traffic prediction, small errors in each time step in the time series will increase when predictions are made further into the future [40]. This is quite challenging to handle Complex spatio-temporal correlations.
- Spatial correlations that change throughout time. In a road network, the dynamics of traffic conditions among sensors alter dramatically over time (before and during peak hours, during weekdays and weekends, Holidays and weather).
- Temporal correlations that aren't linear. Because of unanticipated incidents and accidents, the traffic state near a sensor may shift abruptly, altering the relationship between the different time steps.
- To address the difficulties stated above, a Transform Attention layer is added between the encoder and the decoder, which converts the encoded past traffic characteristics into future representations before passing them to the Decoder.

## 4. Implementation

**Traffic Monitoring System**
The time series for this study of road traffic statistics were taken from data. gov. uk. The data was collected during a 15-minute period from 2010 to 2014 (1826 days). We have 96 observations per 24 hours. As a result, there are a total of 1826*96 = 175296 observations.

**Hypothesis Testing**

**Traffic Flow Trend**
**Claim:** The population of the area is expected to increase over time. As a result, the traffic flow is anticipated to rise over time in fig2. .

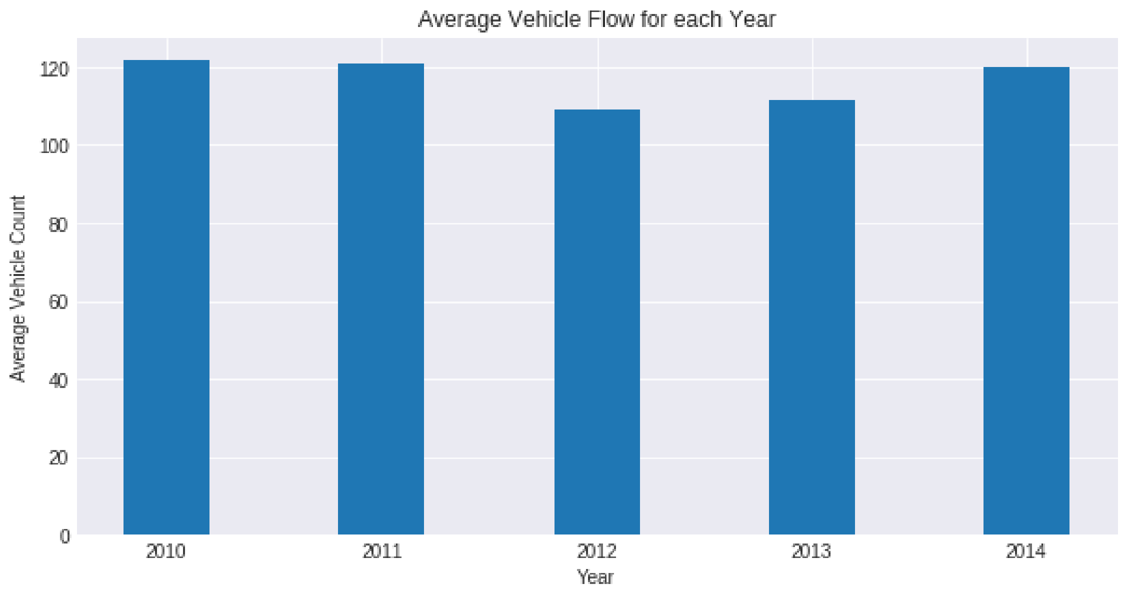


**Figure 2:** Average Vehicle Flow for each year

**Conclusion**
No trend exists. Alternative hypothesis holds true.

**Daily Seasonality**

**Claim:** The flow of traffic on the roads will vary depending on the time of day (rush hours vs off hours). As a result, daily seasonality is to be expected in fig3.

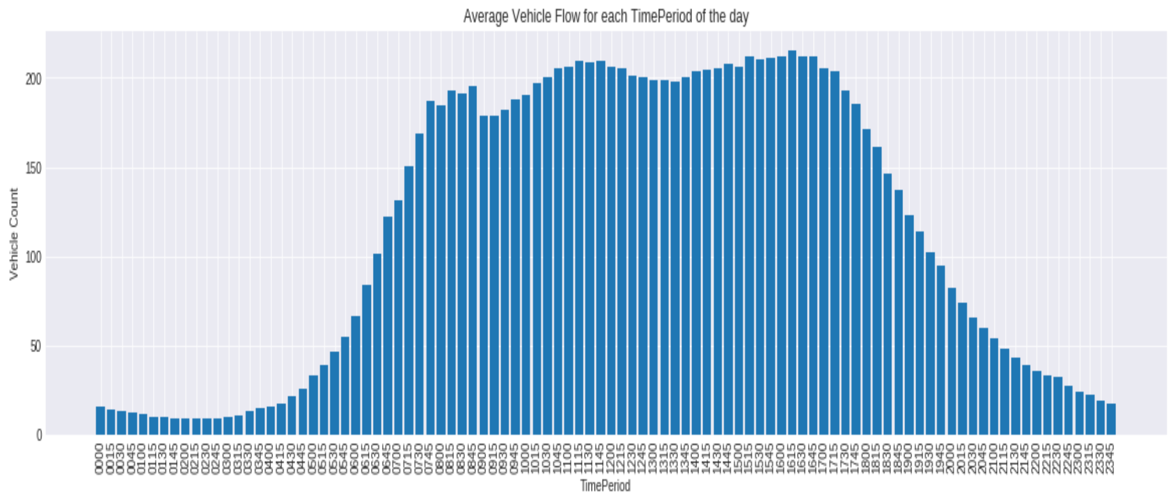


**Figure 3:** Average Vehicle Flow for each Time Period of the day

**Conclusion**
- Because identical patterns in traffic frequency may be detected over specific time intervals, it can be assumed that the data has daily seasonality.
- On weekdays, the morning and evening peak hours are at 11: 45 a. m. and 16: 15 p. m., respectively.
- The Null Hypothesis is correct.

**Weekly Seasonality**

**Claim:** The volume of traffic will be higher on weekdays than on weekends. As a result, seasonality on a weekly basis is to be expected in fig 4.



Paper ID: SR221018164349 DOI: 10.21275/SR221018164349 

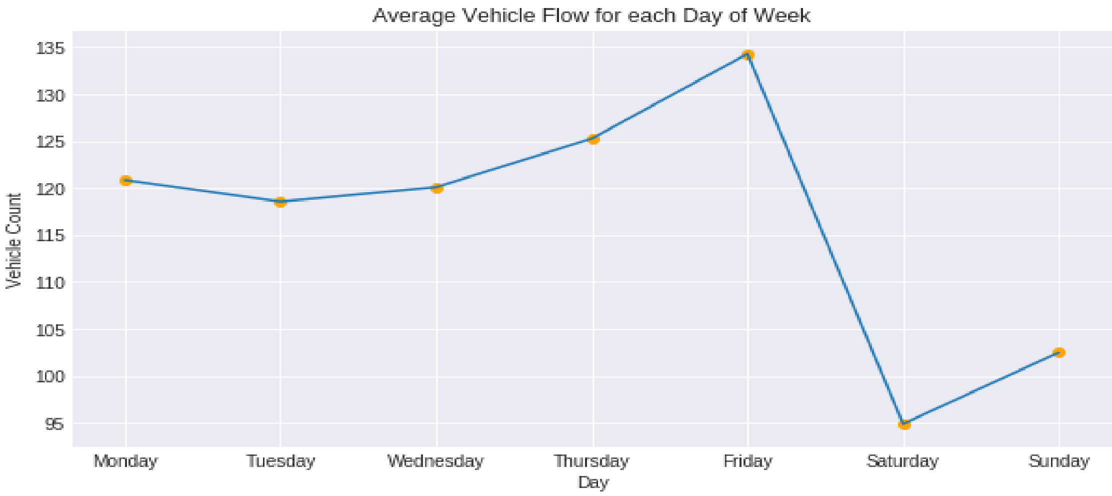


**Figure 4:** Average Vehicle Flow for each day of the Week

**Conclusion**

- Similar trends in traffic frequency may be detected for each week on a monthly scale, indicating that the data has a weekly seasonality.
- On weekdays, traffic is heavier, with Friday being the busiest; yet, the Null Hypothesis holds true on weekends, when traffic is lighter.

**Annual Seasonality**

**Claim:** Due to changes in weather conditions, the flow of traffic on the roads will vary throughout the year. As a result, yearly seasonality is to be expected in fig 5.

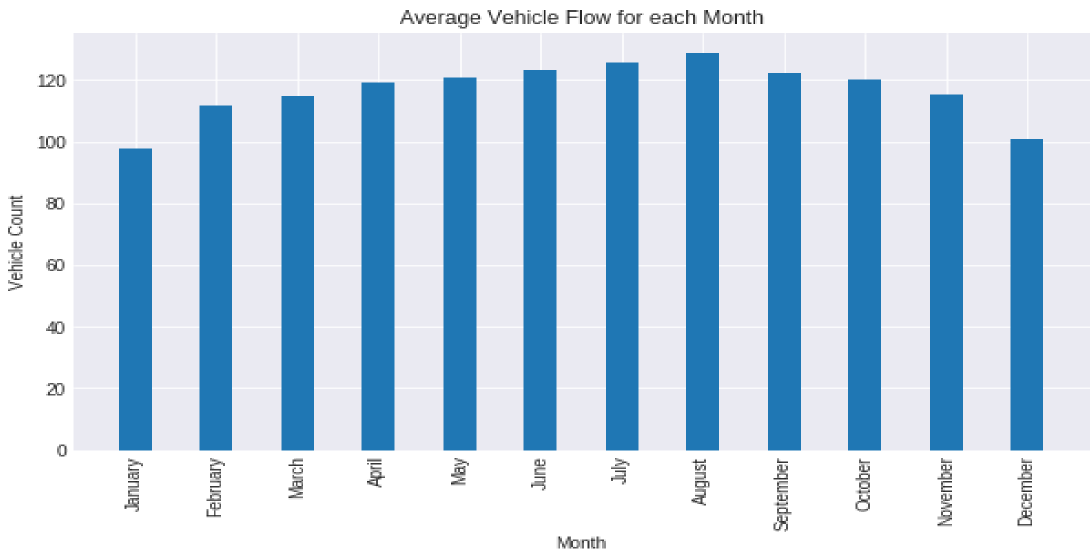


**Figure 5:** Average Vehicle Flow for each Month

**Conclusion**

- The data has annual seasonality, since comparable trends in traffic frequency may be seen for each month on a yearly basis.
- Until August, the average traffic flow increases, then drops until December-January.
- The Null Hypothesis is true.

**Algorithm:**

**Input:** Date, Time period, Average Time taken, Average Speed, Road Length, Flow of vehicle

**Output:** Short term Prediction of Traffic in a particular region

**Date:** The timestamp for a given day.

STEP 1. Hypothesis Testing for Trend and Seasonality Component.

STEP 2: Testing the Null Hypothesis on Seasonality on a Daily, Weekly, and Annual Scale.

STEP 3: Verify Stationarity of Time Series Using the Dickey Fuller Test

STEP 4: Plot ACF, PACF (Auto correlation and Partial Auto correlation) plots of the remaining time series after decomposition.

STEP 5: Using the SARIMA Model to handle data seasonality.

**Traffic Prediction System**

**Deep Residual Network for Spatio-Temporal Traffic Prediction with External Fusion**

**Data Preprocessing**

The original data set has divided the whole of Beijing into a 32 × 32. The small area has also been counted for the incoming and outflow of each small area every half an hour (a time slice), which has been expressed as [2, 32, 32].

The model uses the first 3 time slices of the current moment to simulate the proximity (Closeness), and uses a time slice of the same moment of the day before the current moment to simulate the periodicity (Period). A time slice at the same time of the week to simulate the trend (Trend), that is, len_closeness=3, len_period=1, len_trend=1 in the code as three hyper parameters in fig 6.

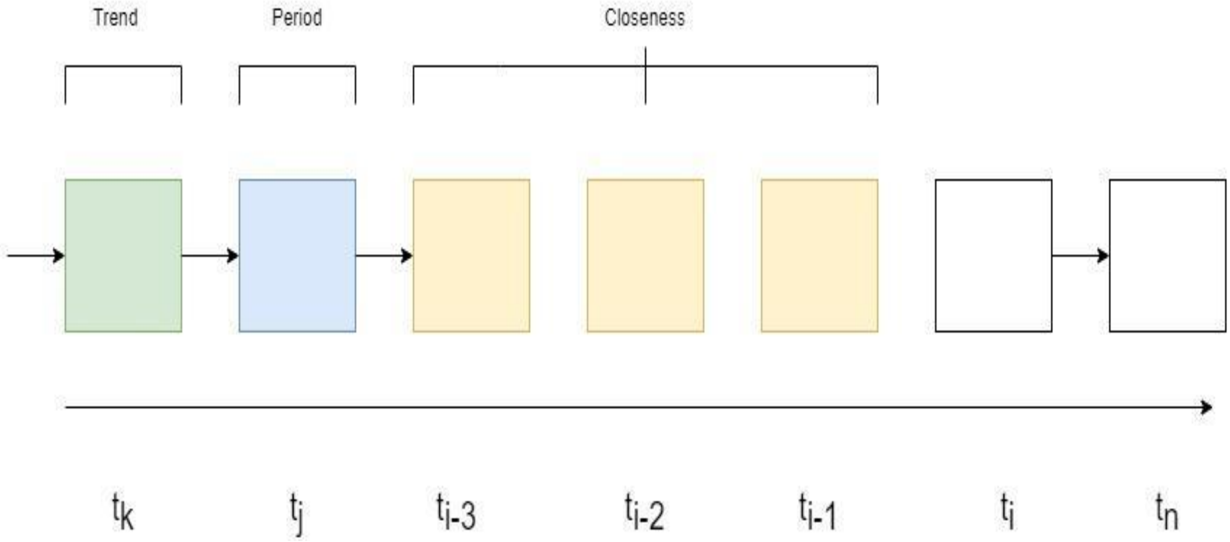


**Figure 6:** Time Series Period

For each time slice:
**time_feature**-There are 8 dimensions, and the first seven dimensions are in the form of one-hot, and finally it is thought to indicate whether the day is a working day.
**holiday_feature**-There is a dimension, 0 indicating that the day of the time slice is a working day, and 1 indicates a holiday.
**meteorol_feature**-There are 19 dimensions; the first 17 are also one-hot, denoting a particular sort of weather, and the final two, representing the wind speed and temperature, respectively.
These three vectors are then stitched together to form a 28-dimensional vector. The network is receiving this data form

**Inflow/Outflow Formulation**
Let P be vehicle trajectories at the time interval (t). Grid (i * j), the inflow and outflow of the crowds at the time interval t are defined respectively as in Eq.2.

$$x_t^{in,i,j} = \sum_{Tr \in P} |\{k > 1 \,|\, g_{k-1} \notin (i,j) \wedge g_k \in (i,j)\}|$$

$$x_t^{out,i,j} = \sum_{Tr \in P} |\{k \geq 1 \,|\, g_k \in (i,j) \wedge g_{k+1} \notin (i,j)\}|$$

**Equation 2:** Inflow/Outflow mathematical formulation in geo spatial co ordinate

*G-Grid (i, j)*
*t-Time interval*
*Tr:* $g_1 \rightarrow g_2 \rightarrow \cdots \rightarrow g\ |T_r|$ *is a trajectory*
$g_k$ *is the geospatial coordinate*
*I-inflow*
*O-outflow*

Traffic prediction is the process of predicting the inflow and outflow flow at a future point time in a certain location using previous data. There are two types of Traffic flow

1) Inflow
2) Outflow

**Model Design**
Convolution-based residual networks are used to model nearby and distant spatial dependencies between any two regions in a city while ensuring that the model's prediction accuracy is not harmed by the deep structure of the neural network, and to classify the temporal properties of crowd flows into three categories: temporal closeness, period, and trend. ST-ResNet mimics these traits by dynamically aggregating three residual networks with varied weights allocated to distinct branches and regions. Fusion with external elements is the aggregation method (Weather and Social Holidays). With three networks, the model captures the Trend, Closeness, and Period attributes individually, before fusing them with External Time, Meteorological, and Holiday variables. To fit periodicity, proximity, and trend, the model employs three CNN models. CNN employs a total of 12 residual units. Each residual unit has two ReLu and Conv modules that are cascaded. The problem of disappearing gradients has been handled. The three attributes are linked by weights, and different weights are employed in different areas and components due to the sensitivity of each region to various causes. Other features are connected by a fully connected network. Finally, their output is combined with the output of three CNN fusions, and are added and mapped to-1 through the tanh function as in Eq.3 and 4.

$$X_{Res} = W_c . X_c + W_p . X_p + W_q$$

*Eqn 3 – ResNet Model Equation*

c, p, t denotes Closeness, Period, Trend

$$X_{Final} = X_{Res} + X\_Ext$$

*Eqn.4-ResNet Model Final Equation*

By translating inflow and outflow throughout a city at every time period, we first generate an e matrix. Next, split the time line into three sections: recent history, recent past, and distant past. To describe the three previously stated temporal features of proximity, period, and trend, the interval matrices for each chosen period are then divided into first three factors independently. A convolutional neural network's initial three elements and the Residual Unit sequence that follows them all have the same network structure. The spatial relationship between local and far-off places is captured by this structure. In the external component, we physically extract different characteristics from other datasets, such meteorological occurrences and conditions, and input them into a two perfectly neural network. Using variable matrices that apply varying weights to the outcomes of various components in various places, the outputs of first three elements are fused as $X_{Res}$. The output of Ext, an external component, is also integrated with $X_{Res}$. The proposed work architecture in Fig 7.

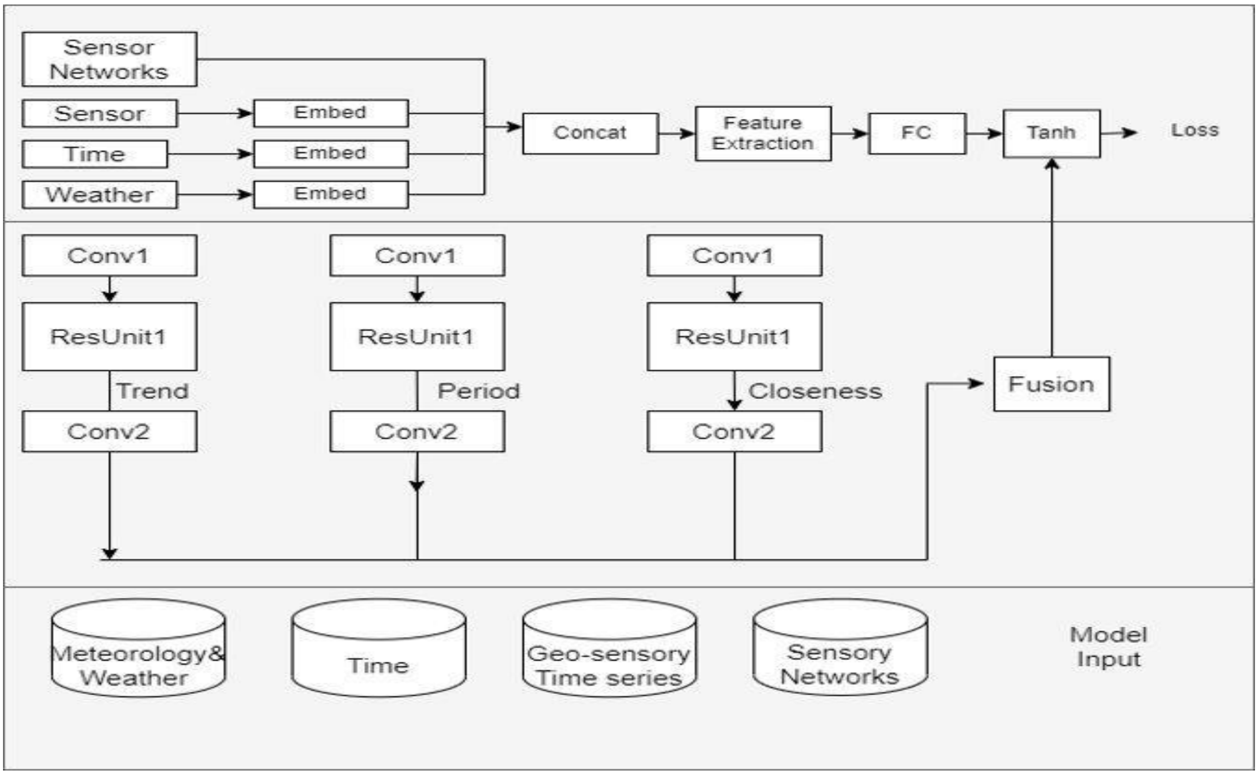


**Figure 7:** Architecture Diagram of the Proposed Work

**Proposed Algorithm:**

**Spatio-Temporal Model with External Fusion Traffic Prediction Algorithm**

**Input**: Time Series Historical Data of Traffic {T0… Tn-1}
Factors for External Fusion: $\{F_0, \cdot\cdot\cdot, F_{n-1}\}$;
Closeness, period, trend length of Sequences: lc, lp, lq;

**Output**: Trained and Learned Model for Traffic Prediction
1) Range of time interval t ($1 \leq t \leq n-1$)
2) Seqc = [Tt-lc, Tt-(lc-1), · · ·, Tt-1]
Seqp = [Tt-lp·p, Tt-(lp-1) ·p, · · ·, Tt-p]
Seqq = [Tt-lq·q, Tt-(lq-1) ·q, · · ·, Tt-q]
3) Fusion based on Parameter to assign weights to result of different components in (2)
4) Fusion based on External Features – Fuse the model with External Feature vector Et at t
5) Training instance is created

**Graph Encoder Decoder with Attention Layer**
To transform past traffic characteristics into future representations, a Transform attention method is used. This attention mechanism reduces the problem of mistake propagation by modelling direct linkages between past and future time steps. The decoder foretells the output sequence once the encoder had digested the incoming traffic information. A transform attention layer is used between the encoder and the decoder to transform the encoded traffic conditions into sequence representations of upcoming time steps like that of the decoder's input. A stack of Attention bricks is used to construct the both encoder and the decoder. A spatial attention process for modelling fluid spatial relationship, a temporal focus mechanism at model non-linear uses the results, and proposed spatial and temporal attention techniques are all included in each Attention block.

**Definition**
It is an undirected graph. Real-time traffic information and the road network are shown. The number of features connected to traffic is indicated by the letter C. (such as traffic volume and traffic speed). Based on P prior time step data, the aim is set. Get the traffic scenario for the next Q stages.

**Architecture**
Attention blocks with residual connections make up the encoder and decoder. A spatial attention mechanism, a temporal attention mechanism, and a gate fusion mechanism make up each space-time attention block. The intermediary

Attention transfers the encoder's output properties to the decoder.

**Spatiotemporal Embedded Module (STE)**

According to the traffic to the road, the road structure information is embedded in the D-dimensional vector using the embedding method of the graph. The node2vec to obtain the representation results. This, however, is a static depiction that does not reflect the dynamic connection of the road network's midway. The time step information is then added, resulting in each time interval having its own space embedded. To get the time step, a single-heat approach is used to encode it, and then a two-layer fully connected network is used to embed it into the D-dimensional. Then sum the two to get the Vi node and tj moment's spatial embedding result. The output of each point will be equal to the product of the information and attention coefficient of each node in the previous layer. Time correlation is affected by traffic conditions and corresponding time background.

To adaptively replicate the nonlinear interaction between multiple time steps, the following temporal attention technique is proposed. Attention is utilized to limit the error propagation impact between anticipated time steps and thus it provides the future representations for further implementations of output sequence. It decodes encoded traffic characteristics into produced future representations as decoder input, modelling the direct relationship between each upcoming time step and each historical time step. SoftMax is used to calculate the attention coefficient, and the attention coefficient is combined with historical time data. The model uses historical observation data to generate the whole connection layer, which is subsequently encoded using the attention conversion module. After that, go through the decoder. Finally, the final output is generated using the entire connection. The mean square error, sometimes known as "loss, " is defined as follows:

**Graph Encoder Decoder with Attention LayerAlgorithm**

Input: X = (X t1, X t2, X tj)-N vertices j-Historical steps Traffic condition

Graph G = (V, E, A), N = |V| vertices, A – adjacency matrix

Output: Y = (Xˆt P+1, Xˆt P+2, Xˆt P+i) for next i steps

1) Historical observation is transformed before encoder
2) Encoder-encodes the traffic feature
3) Transformation attention to generate future sequence Seq (L)-> Seq (L+1)
4) Decodes the Traffic feature and produces Y for next i steps

**Traffic Management and Control System**

Web App built using Angular JS, Angular Fire and Firebase to create a personalized Traffic Feed which alerts the user about traffic events on his/her route from Source to Destination.

1) User enters the Location
2) Classify the Traffic according to Traffic Type (Rally, VIP Convoy, Diversion, Road Block, Work in progress and Accident)
3) Classify the Intensity of the Traffic (Low, Medium and High)
4) Publish the Alert to the peers and Peer user will be notified of the event

Input: Location of the Traffic, Intensity of Traffic, Type of Traffic (VIP Convoy, Diversion)

Output: Alert the User about Traffic

Live Alerts about the

*Traffic jams, Road blocks, Diversions, VIP convey, Rallies, Construction alerts*

1) Dynamically combining the three networks to assign different weights to the three CNN networks, this method can model the importance of different regions affected by different factors.
2) Further combined with other external factors, the real-time flow of people is predicted, such as rain/snow, whether major events occur and holidays have a great impact on the flow of people. After integration, it can effectively predict the flow of major events in the future.

## 5. Results and Discussion

**Traffic Monitoring System**

**ACF and PACF Analysis**

To determine the order of AR (Autoregressive) and/ or MA (Moving Average) term is shown in Tab I and fig 8 & fig 9

Parameters that are required for the ARIMA model:

'p' or the number of Auto-Regressive terms: These are the lags of forecasted variable

'q' or the number of Moving-Average terms: These are the lagging forecasted error

'd' or number of differences: This is the number of non-seasonal differences

**Table I:** ACF and PACF Parameters

| MODEL | ACF | PACF |
|---|---|---|
| AR | Tails off gradually | Significant till 'n' lags |
| MA | Significant till 'n' lags | Tails off gradually |
| ARIMA | Tails off gradually | Tails off gradually |

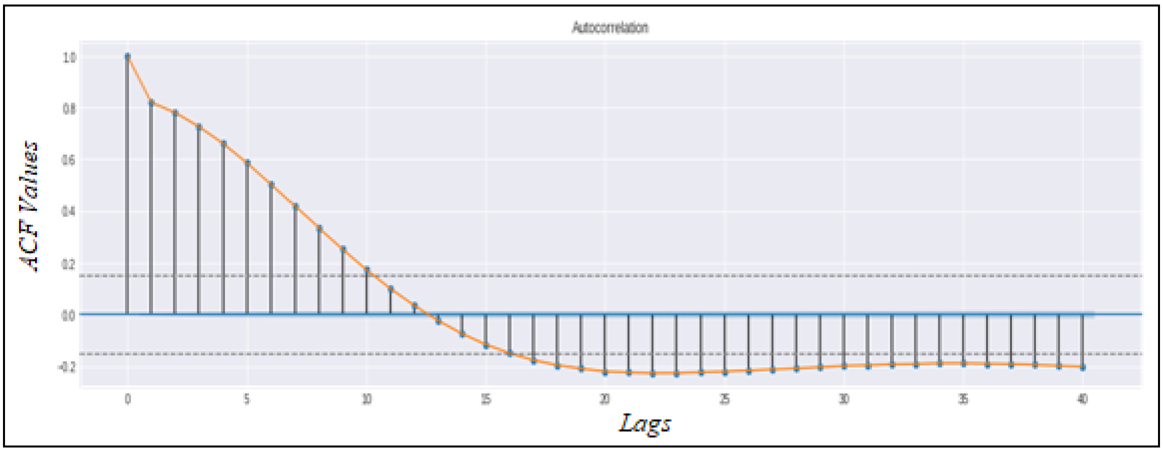


**Figure 8:** ACF Plot

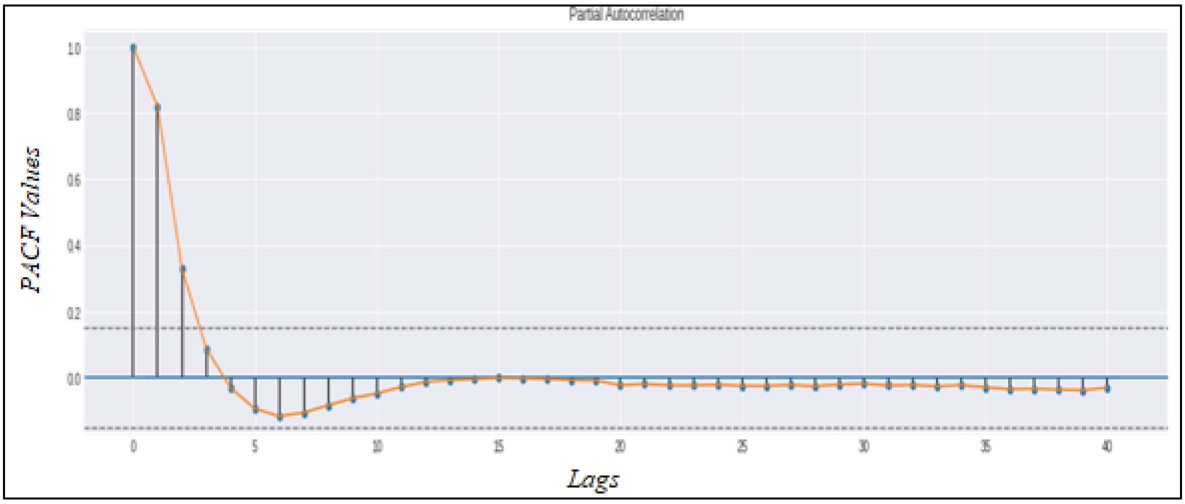


**Figure 9:** PACF Plot

**Inference**
- From the given PACF plot, value of 'p' is 2.
- From the given ACF plot, value of 'q' is 10.

**Stationarity of Time Series**

To ascertain the stationarity of a particular time series, the Augmented Dickey-Fuller normality test is applied. The study's concept is as follows. The time series is non-stationary, according to the null of the Dickey-Fuller test (presence of unit root). The data series is stationary, according to the Dickey-Fuller test's Alternate Hypothesis (absence of unit root).

**Results of the Augmented Dickey-Fuller Test:**

**Table II:** ADF Test Result

| Statistics | Value |
|---|---|
| Test Statistic | - 26.380273 |
| P-value | 0.000000 |
| Lags Used | 37.000000 |
| Number of Observations Used | 8602.000000 |
| Critical Value / Significance Level (1%) | - 3.431110 |
| Critical Value / Significance Level (5%) | - 2.861876 |
| Critical Value / Significance Level (10%) | - 2.566949 |

**Inference**
- A significance criterion of less than 1% is indicated by a p-value of 0. With a 95% confidence level, this proves that the null hypothesis is untrue. As a result, time series become immobile.
- The Test Statistic is significantly smaller than the Critical Values. As a result, the null hypothesis is rejected, and the stationary nature of the time series is proven.
- There are no de-trending operations or differencing necessary since the time series is stable.

**Traffic Prediction**

These Plots Fig 10 and 11. Depicts the Epoch vs Accuracy and epochs vs Loss in terms of Root Mean Square Error (RMSE).

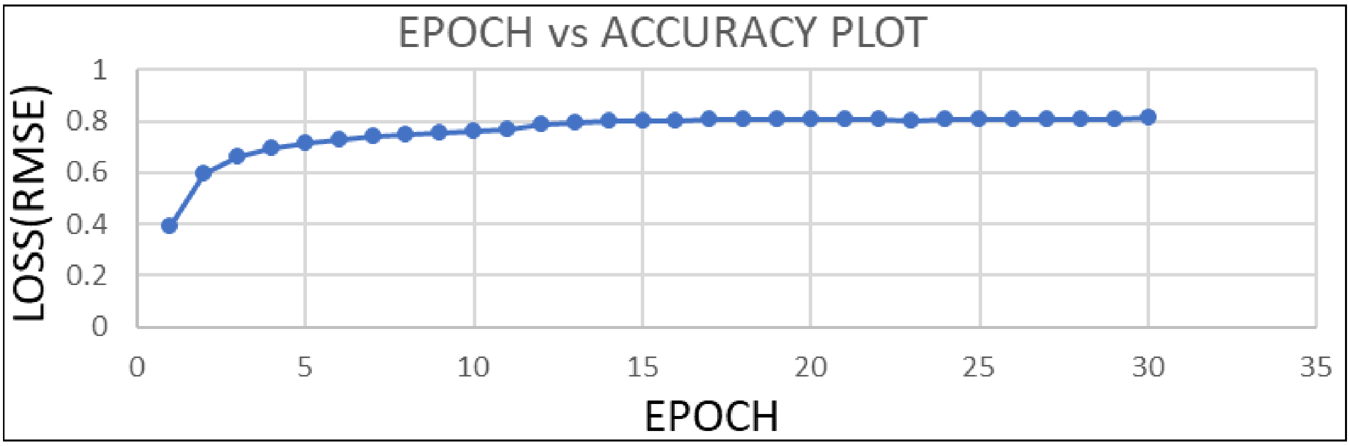


**Figure 10:** Epoch vs Accuracy Plot

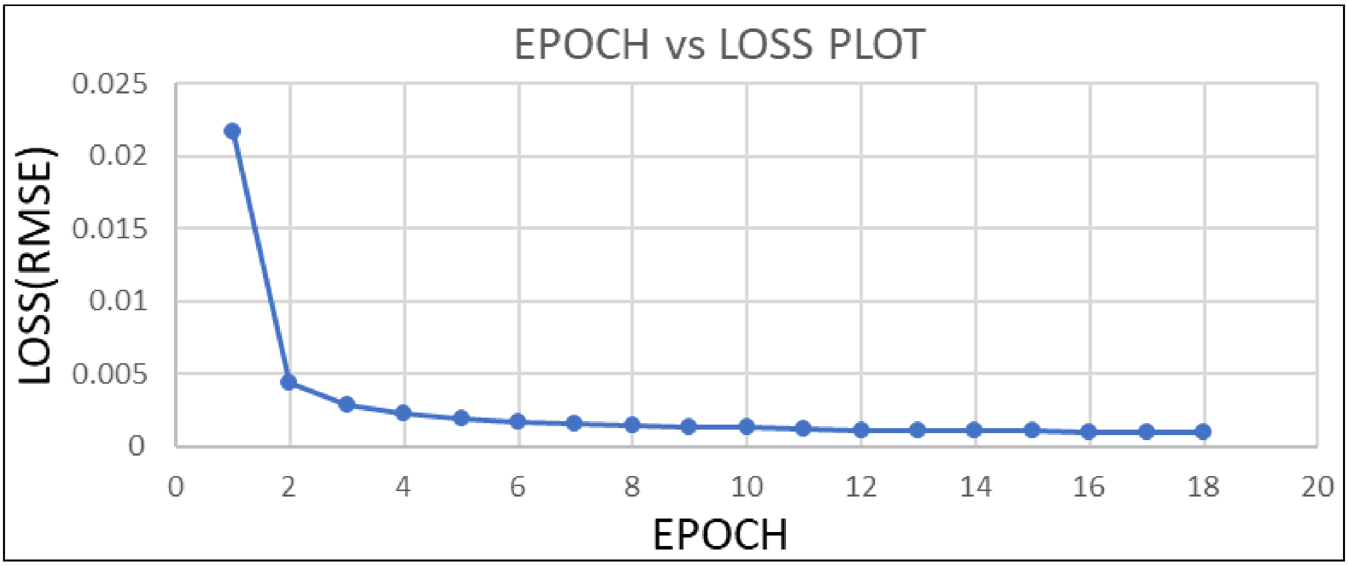


**Figure 11:** Epoch vs Loss Plot

**Comparative Analysis**

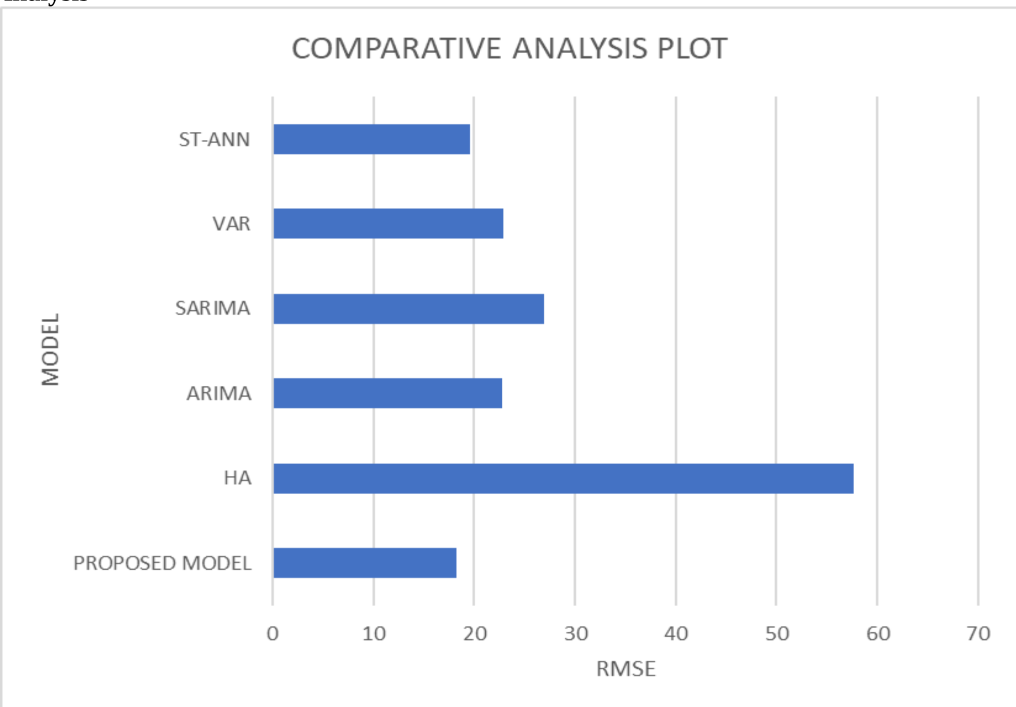


**Figure 12:** Comparative Analysis of model

**Table III:** Comparative Analysis of Model

| MODEL | LOSS (RMSE) |
|---|---|
| HA [32] | 57.69 |
| ARIMA [32] | 22.78 |
| SARIMA [32] | 26.88 |
| VAR [32] | 22.88 |
| ST-ANN [32] | 19.57 |
| PROPOSED MODEL | 18.18 |

In terms of Root Mean Square Error (RMSE), the suggested model outperformed the baseline models (ANN, ARIMA, SARIMA and HA). The proposed model is contrasted with the following six baselines: The Auto-Regressive Integrated Moving Average is a well-known methodology for evaluating and forecasting future values in a time series (ARIMA). It is determined by using the average value of past inflow and outflow throughout the relevant periods. A more sophisticated geographical model that can capture bilateral correlations among all flows is seasonal ARIMA, Vector Auto-Regressive (VAR), however because of the enormous number of parameters, it has high processing costs [43]. ST – ANN extracts spatial (nearby 8 regions' values) and temporal (eight previous time intervals) features, which are then fed into an artificial neural network is shown in Table III. The comparative analysis is shown in above fig 12.

**Model Training parameters**

**Flow**

Traffic flow is defined as the volume of vehicles passing through a specific area of a road in a predetermined amount of time.

**Speed**

A vehicle's actual speed is calculated using the distance it covers in one unit of time. Each automobile on the road will often go at a different pace than others around it owing to factors such geographic location, traffic patterns, driving time, surrounds, and the car's personal circumstances.

**Demand**

The amount of start/pick-up or end/drop-off requests is utilized as a measure of the demand in an area at a particular moment. Predict the number of cab requests for a region inside a future time step.

**Travel Time**

Time to travel between two routes (source to destination) including the waiting time in intersection.

**Occupancy**

Road utilization is gauged by how much space vehicles take up on the road [42]. The model training graphs for data training in fig 13, Training epochs in fig 14, the snippet for the traffic event classification in fig 15 and finally the fig 16. Shows the training and validation accuracy.

```
loading data...
trainX: torch.Size([36458, 12, 325])          trainY: torch.Size([36458, 12, 325])
valX:   torch.Size([5189, 12, 325])           valY:   torch.Size([5189, 12, 325])
testX:   torch.Size([10400, 12, 325])         testY:   torch.Size([10400, 12, 325])
mean:   62.7354         std:   9.4388
data loaded!
compiling model...
trainable parameters: 209,923
**** training model ****
Training batch: 5 in epoch:0, training batch loss:64.1121
Training batch: 10 in epoch:0, training batch loss:38.0095
Training batch: 15 in epoch:0, training batch loss:47.9868
Training batch: 20 in epoch:0, training batch loss:46.6951
Training batch: 25 in epoch:0, training batch loss:39.4574
Training batch: 30 in epoch:0, training batch loss:35.5275
Training batch: 35 in epoch:0, training batch loss:29.8470
Training batch: 40 in epoch:0, training batch loss:28.2383
Training batch: 45 in epoch:0, training batch loss:27.3480
Training batch: 50 in epoch:0, training batch loss:44.0216
Training batch: 55 in epoch:0, training batch loss:42.5420
Training batch: 60 in epoch:0, training batch loss:30.9145
Training batch: 65 in epoch:0, training batch loss:55.3763
Training batch: 70 in epoch:0, training batch loss:34.1983
Training batch: 75 in epoch:0, training batch loss:42.0270
Training batch: 80 in epoch:0, training batch loss:41.4296
Training batch: 85 in epoch:0, training batch loss:37.6397
Training batch: 90 in epoch:0, training batch loss:35.3683
Training batch: 95 in epoch:0, training batch loss:42.4540
Training batch: 100 in epoch:0, training batch loss:32.7066
Training batch: 105 in epoch:0, training batch loss:48.7892
Training batch: 110 in epoch:0, training batch loss:46.1539
Training batch: 115 in epoch:0, training batch loss:32.5278
Training batch: 120 in epoch:0, training batch loss:36.8000
Training batch: 125 in epoch:0, training batch loss:36.8974
```

**Figure 13:** Sample Model Training of Data

```
Train on 12355 samples, validate on 1373 samples
Epoch 1/50
12355/12355 [==============================] - 1649s - loss: 0.0217 - acc: 0.3896 - rmse: 0.1261 - val_loss: 0.0067 - val_acc: 0.4181 - val_rmse:
Epoch 2/50
12355/12355 [==============================] - 1649s - loss: 0.0044 - acc: 0.5940 - rmse: 0.0661 - val_loss: 0.0035 - val_acc: 0.5558 - val_rmse:
Epoch 3/50
12355/12355 [==============================] - 1637s - loss: 0.0028 - acc: 0.6639 - rmse: 0.0524 - val_loss: 0.0028 - val_acc: 0.6199 - val_rmse:
Epoch 4/50
12355/12355 [==============================] - 1655s - loss: 0.0022 - acc: 0.6959 - rmse: 0.0464 - val_loss: 0.0021 - val_acc: 0.6433 - val_rmse:
Epoch 5/50
12355/12355 [==============================] - 1664s - loss: 0.0019 - acc: 0.7151 - rmse: 0.0429 - val_loss: 0.0020 - val_acc: 0.6558 - val_rmse:
Epoch 6/50
12355/12355 [==============================] - 1634s - loss: 0.0016 - acc: 0.7294 - rmse: 0.0402 - val_loss: 0.0019 - val_acc: 0.6668 - val_rmse:
Epoch 7/50
12355/12355 [==============================] - 1613s - loss: 0.0015 - acc: 0.7399 - rmse: 0.0386 - val_loss: 0.0018 - val_acc: 0.6791 - val_rmse:
Epoch 8/50
12355/12355 [==============================] - 1621s - loss: 0.0014 - acc: 0.7483 - rmse: 0.0373 - val_loss: 0.0015 - val_acc: 0.6989 - val_rmse:
Epoch 9/50
12355/12355 [==============================] - 1617s - loss: 0.0013 - acc: 0.7561 - rmse: 0.0360 - val_loss: 0.0013 - val_acc: 0.7100 - val_rmse:
```

**Figure 14:** Training Model Output Screenshot

```
{1: 'road closed due to construction',
 2: 'traffic jam',
 3: 'stopped car on the shoulder',
 4: 'road closed',
 5: 'other',
 6: 'object on roadway',
 7: 'major event',
 8: 'pothole',
 9: 'traffic heavier than normal',
 10: 'road construction',
 11: 'fog',
 12: 'accident',
 13: 'slowdown',
 14: 'stopped car',
 15: 'small traffic jam',
 16: 'stopped traffic',
 17: 'heavy traffic',
 18: 'minor accident',
 19: 'medium traffic jam',
 20: 'malfunctioning traffic light',
 21: 'missing sign on the shoulder',
 22: 'animal on the shoulder',
 23: 'animal struck',
 24: 'large traffic jam',
 25: 'hazard on the shoulder',
 26: 'hazard on road',
 27: 'ice on roadway',
 28: 'weather hazard',
 29: 'flooding',
 30: 'road closed due to hazard',
 31: 'hail',
 32: 'huge traffic jam'}
```

**Figure 15:** Traffic Event Classification

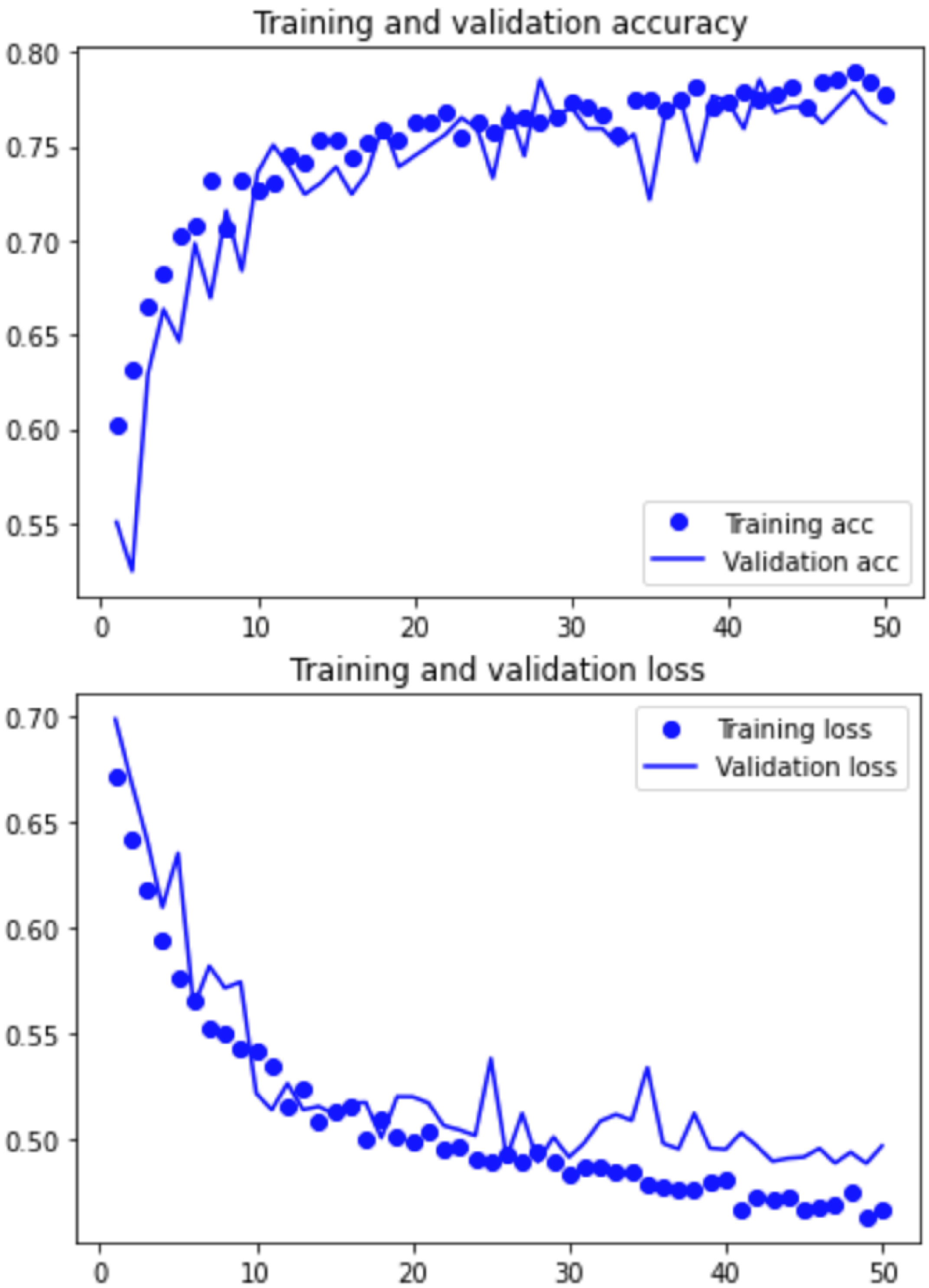


**Figure 16:** Accuracy vs Epoch Plot

## 6. Conclusion & Future Work

There are three modules that have been implemented as part of the Work namely, Traffic monitoring system, Traffic Prediction system and Traffic congestion control and management system. The Traffic monitoring system module is implemented with SARIMA Algorithm. The Trend and Seasonality in the Time series (Weekly, Daily, Annual) is analyzed and features Average Speed, Length of Road and Average time taken is visualized for every 15 minutes' interval. The second module Traffic prediction system, two algorithms has been proposed namely Graph Encoder Decoder with Attention Layer and Temporal – Spatio Resnet model with External Fusion. In the Graph Encoder Decoder model the Road structure is visualized as Graph as Nodes and distance is being given as input in form of Adjacency matrix for Historical Data. Temporal-Spatio Resnet model, Historical Observation, Weather Features, Time feature and social attributes like Holiday is fused externally with Resnet model. Parameter-based Fusion to assign weights of different components of Temporal Dependency (Period, Trend and Closeness) separately. The Third module Traffic management system is built using Angular Js, Firebase and Angular Fire and developed as web – application. The location of traffic, Intensity of traffic and Type of traffic is being inputted from the user end / City corporation and notified to the respective person in the Region and can reroute their plan accordingly and for Prevention of Traffic jam Traffic events has been classified that occur before the Traffic jam and predict the intensity of Traffic Jam is predicted given the event type of Traffic. These three modules can be incorporated by the city corporation or as part of Smart city Development to manage and control the traffic in metropolitan cities. This system can be developed into an android application and the alerts can be created by the concerned authorities so that Drivers can reach the destination hassle free without being interrupted by sudden Traffic events.

## References

[1] R. Bhuvaneshwari, N. Mathubaala, Pranusha S Bavan, Dr. P. Lakshmi Harika, Dr. M R Sumalatha, 2022, Detection of Autism Spectrum Disorder using Machine Learning, International Journal Of Engineering Research & Technology (IJERT) Volume 11, Issue 07, pp 533-536, July 2022.

[2] Lakshmi Harika Palivela Thanuja Varshini, Bhagya Narayanan, Sowmya R, M R Sumalatha, “Cyber Bullying Detection using Machine Learning and Deep Learning”, International Journal of All Research Education and Scientific Methods (IJARESM), vol 10 (7), pp: 2783-2790, July 2022. (Impact Factor: 7.429)

[3] M. R Sumalatha, M Ananthi, “Efficient data retrieval using adaptive clustered indexing for continuous queries over streaming data”, Cluster Computing, Vol.22, no.5, pp.10503-10517, 2019.

[4] Srimathi Ravisankar, Kanimozhi Mahendran, Srilakshmi Arulmurugan, MR Sumalatha, “Flexible Demand Forecasting in Intelligent Food Supply Chain Management, May 25, 2022. SSRN: https: //ssrn. com/abstract=4119151”.

[5] E Pugazhendi, M. R Sumalatha and Lakshmi Harika P, “Weight Based Deduplication for Minimizing Data Replication in Public Cloud Storage”, Journal of Scientific & Industrial Research Vol.80, March 2021, pp.260-269.

[6] Sum`alatha, M. R & Lakshmi Harika, P, “An Intelligent System to Detect Human Suspicious Activity Using Deep Neural Networks”, Journal of Intelligent & Fuzzy Systems of IOS Press. vol.36, no.5, May 2019, pp.4507-4518.

[7] M. R Sumalatha, M Parthiban, “SaaS multitenant performance testing over social networks”, International Journal of Enterprise Network Management, vol.9, no 3-4, pp 234-250, 2018.

[8] Luiz Fernando Pinto de Oliveira, et. al "Development of a Smart Traffic Light Control System with Real-Time Monitoring" IEEE Internet of Things Journal, vol.8 (5), pp.3384-3393, 2021.

[9] Dominik Cvetek, et. al "Traffic Flow Forecasting at Micro-Locations in Urban Network using Bluetooth Detector" IEEE Transactions on Intelligent Transportation Systems, pp.57-60, 2020.

[10] Chuanpan Zheng, et. al "Mining Spatio-Temporal Disturbances of Multiple Context Factors for Citywide Traffic Flow Prediction" IEEE Transactions on Intelligent Transportation Systems, vol.21 (9), pp.3744-3755, 2020.

[11] Zhiyuan Wang, et. al "A Robust Vehicle Detection Scheme for Intelligent Traffic Surveillance Systems in Smart Cities" IEEE Access, vol.8, pp.139299-39312, 2020.

[12] Zhiguang Cao, et. al "A Unified Framework for Vehicle Rerouting and Traffic Light Control to Reduce Traffic Congestion" IEEE Transactions on Intelligent Transportation Systems, vol.18 (7), pp.1958-1973, 2017.

[13] Zheng H, Lin F, Feng X, et al. “A hybrid deep learning model with attention-based conv-LSTM networks for short-term traffic flow prediction” IEEE Transactions on Intelligent Transportation Systems, vol.22 (10), pp.6910-6920, 2020.

[14] Yu Y, Zhang Y, Qian S, et al. “A low rank dynamic mode decomposition model for short-term traffic flow prediction [J]”. IEEE Transactions on Intelligent Transportation Systems, vol.22, pp.6547-6560, 2020.

[15] Li Z, Xiong G, Tian Y, et al. “A Multi-Stream Feature Fusion Approach for Traffic Prediction [J]”. IEEE Transactions on Intelligent Transportation Systems, vol.23 (2), pp.1456-1466, 2020.

[16] Shi X, Qi H, Shen Y, et al. “A spatial-temporal attention approach for traffic prediction [J]”. IEEE Transactions on Intelligent Transportation Systems, vol.22 (8), pp.4909-4918, 2020.

[17] Gu Y, Lu W, Xu X, et al. “An improved Bayesian combination model for short-term traffic prediction with deep learning [J]”. IEEE Transactions on Intelligent Transportation Systems, vol.21 (3), pp.1332-1342, 2019.

[18] Guo Z, Zhang Y, Lv J, et al. “An online learning collaborative method for traffic forecasting and routing optimization [J]”. IEEE Transactions on Intelligent Transportation Systems, vol.22 (10), pp.6634-6645, 2020.

[19] M. Ananthi, M. R. Sumalatha, “Efficient Query Processing of Streaming Data Using dynamic heuristic optimization approach”, Asian Journal of Information Technology, Vol 15 (15), pp: 2671-2678.

[20] Tian Z. et. al “Approach for short-term traffic flow prediction based on empirical mode decomposition and combination model fusion [J]”. IEEE Transactions on Intelligent Transportation Systems, vol.22 (9), pp.5566-5576, 2020.

[21] Liu Y, Lyu C, Liu X, et al. “Automatic feature engineering for bus passenger flow prediction based on modular convolutional neural network [J]”. IEEE Transactions on Intelligent Transportation Systems, vol.22 (4), pp.2349-2358, 2020.

[22] Ma D, Song X, Li P, et al. “Daily traffic flow forecasting through a contextual convolutional recurrent neural network modeling inter-and intra-day traffic patterns [J]”. IEEE Transactions on Intelligent Transportation Systems, vol.22 (5), pp.2627-2636, 2020.

[23] Zhang J, Chen F, Cui Z, et al. “Deep Learning Architecture for Short-Term Passenger Flow Forecasting in Urban Rail Transit [J]”. IEEE Transactions on Intelligent Transportation Systems, vol.22 (11), pp.7004-7014, 2020.

[24] Meng X, Fu H, Peng L, et al. “D-LSTM: Short-Term Road Traffic Speed Prediction Model Based on GPS Positioning Data [J]”. IEEE Transactions on Intelligent Transportation Systems, vol.23 (3), pp.2021-2030, 2020.
[25] Guo K, Hu Y, Qian Z, et al. “Dynamic Graph Convolution Network for Traffic Forecasting Based on Latent Network of Laplace Matrix Estimation [J]”. IEEE Transactions on Intelligent Transportation Systems, vol.23 (2), pp.1009-1018, 2020.
[26] Shuai Xu, Jiuxin Cao, Phil Legg, Bo Liu, Shancang Li, et. al. “An Efficient Embedding model for fine-grained user location prediction in Geo Social Networks” IEEE Systems Journal, vol.14 (2), pp.1740-1751, 2020.
[27] M. Rajmohan, C. Theophilus, M. R. Sumalatha & S. Saravanakumar “Facility Location of Organ Procurement Organizations in Indian Healthcare Supply Chain Management”, South African Journal of Industrial Engineering, vol.28 n.1 Pretoria May.2017
[28] Cheng-Te Li, Shou-De Lin, et al. “Simulating Crowds to discover the connection between Spatial-Temporal movements of people and social structure” IEEE Transactions on computational social systems, vol.5 (1), pp.33-45, 2018.
[29] Binbing Liao, Jingqing Zhang, Chao Wu, et al. “Deep Sequence Learning with Auxiliary Information for Traffic Prediction” International Conference on Knowledge Discovery and Data Mining, pp.537-546, 2018.
[30] J. Bruna, W. Zaremba, A. Szlam, and Y. LeCun, “Spectral networks and locally connected networks on graphs”. International Conference on Learning Representations, 2014.
[31] B. Yu, H. Yin, and Z. Zhu, “Spatio-temporal graph convolutional networks: a deep learning framework for traffic forecasting”.27th International Joint Conference on Artificial Intelligence, 2018, pp.3634–3640.
[32] X. Geng, Y. Li, L. Wang, L. Zhang, Q. Yang, J. Ye, and Y. Liu, “Spatiotemporal multi-graph convolution network for ride-hailing demand forecasting, ” AAAI Conference on Artificial Intelligence, vol.33, pp.3656–3663, 2019.
[33] Y. Li, R. Yu, C. Shahabi, and Y. Liu, “Diffusion convolutional recurrent neural networks: data-driven traffic forecasting”, International Conference on Learning Representations, 2018.
[34] S. Fang, Q. Zhang, G. Meng, S. Xiang, and C. Pan, “Gstnet: Global spatial-temporal network for traffic flow prediction”, Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI, pp.10–16, 2019.
[35] Tampubolon, H., & Hsiung, P. A., “Supervised deep learning based for traffic flow prediction”. International Conference on Smart Green Technology in Electrical and Information Systems. pp.95–100, 2018.
[36] Z. Pan, Y. Liang, W. Wang, Y. Yu, Y. Zheng, and J. Zhang, “Urban traffic prediction from spatio-temporal data using deep meta learning”.25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp.1720–1730, 2019.
[37] Sumalatha M. R, Prathika V, Krishna Murthi S, Abarna T, “Blockchain based Transaction Support System in Kidney Supply Chain Management” International Journal Of Advances In Electronics And Computer Science (IJAECS) Volume-6, Issue-9 (Sep, 2019), 37-42.
[38] kavin velmurugan, Santosh Yuvaraj, M. R. Sumalatha, Lakshmi Palivela, “Supply Chain Financial Risk Management Using Business Intelligence”, Journal of the Maharaja Sayajirao University of Baroda, ISSN: 0025-0422 Volume-56, No.1 (VI) 2022, 129-137.
[39] K. Harshitha, P. Veejendhiran, Aditya V, Dr. M R Sumalatha, Dr. P. Lakshmi Harika, 2022, Fake News Detection, International Journal Of Engineering Research & Technology (IJERT) Volume 11, Issue 07, pp: 537-541, July 2022.
[40] M. Ananthi, M. R. Sumalatha, " Timestamp Embedding Query Stream Processing Engine", Indian Journal of Science and Technology, Vol.8, Issue 27, pp.1-6 (2015).
[41] Sumalatha Ramachandran, SelvakumarChithan, SiddharthRavindran, and K. Divya. "Towards cost effective approach for storing intermediate datasets in cloud. " Advances in Natural and Applied Sciences 8, no.18 (2014): 7-14.
[42] M. R. Sumalatha, E. Pugazhendi “An Efficient Utilization of Storage System using Text Intersection Based Deduplication in Nano Cloud Environment”, International Journal titled “Journal of Computational and Theoretical Nanoscience. Volume 15, Number 1, January 2018, pp.53-62 (10).
[43] Sumalatha M. R, Prathika V, Krishna Murthi S, Abarna T, “Blockchain based Transaction Support System in Kidney Supply Chain Management” International Journal Of Advances In Electronics And Computer Science (IJAECS) Volume-6, Issue-9 (Sep, 2019), 37-42