A Thesis/Project/Dissertation Report

on

Traffic Flow Predictionusing Machine Learning

Submitted in partial fulfillment of the

requirement for the award of the degree of

BACHELOR OF ENGINEERING
IN
COMPUTER SCIENCE & ENGINEERING

GALGOTIAS UNIVERSITY

(Established under Galgotias University Uttar Pradesh Act No. 14 of 2011)

Under The Supervision of
Dr. Ganga Sharma

Submitted By

Nitesh Singh (19SCSE1010709)
Mohit Bhatia (19SCSE1010012)

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING / DEPARTMENT OF COMPUTERAPPLICATION
GALGOTIAS UNIVERSITY, GREATER NOIDA
INDIA
APRIL, 2023

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING
GALGOTIAS UNIVERSITY, GREATER NOIDA

CANDIDATE'S DECLARATION

I/We hereby certify that the work which is being presented in the thesis/project/dissertation, entitled **"Traffic Flow Prediction using Machine Learning** in partial fulfillment of the requirements for the award of the B.tech. submitted in the School of Computing Science and Engineering of Galgotias University, Greater Noida, is an original work carried out during the period of month, Year to Month and Year, under the supervision of Dr. Ganga Sharma, Department of Computer Science and Engineering/Computer Application and Information and Science, of School of Computing Science and Engineering , Galgotias University, Greater Noida

The matter presented in the thesis/project/dissertation has not been submitted by me/us for the award of any other degree of this or any other places.

NITESH SINGH (19SCSE1010709)
MOHIT BHATIA (19SCSE1010012)

This is to certify that the above statement made by the candidates is correct to the best of my knowledge.

Dr. Ganga Sharma

CERTIFICATE

The Final Thesis/Project/ Dissertation Viva-Voce examination of Nitesh Singh (19SCSE1010709) and Mohit Bhatia (19SCSE1010012) has been held in April 2023 and his/her work is recommended for the award of B.tech(CSE).

Signature of Examiner(s)

Signature of Supervisor(s)

Signature of Program Chair

Signature of Dean

Date: April, 2023

Place: Greater Noida

Abstract

The traffic flow prediction is an essential part of a city transportation system. With people relying on various modes of transportation, traffic in big cities is increasing at an exponential rate. In this scenario, prediction of traffic flow becomes very important not only for the administration but also for the common man travelling on the road. For this reason, various traffic flow prediction systems based on machine learning algorithms have been proposed in the past few years. However, accuracy of such systems is still a big concern. Also, the parameters considered for predicting traffic flow have not been very comprehensive in nature.

Therefore, in this research, we propose a traffic flow prediction system based on weather and environment conditions. The two most widely used machine learning algorithms, viz. Support Vector Machine (SVM) and Artificial Neural Network (ANN), have been used in this Traffic Flow Prediction Model to cater to long-term and short-term traffic prediction. The traffic data is based on hourly estimates for various time periods. Past Traffic dataset was gathered from PEMS and tomorrow.io and cleaned according to our prerequisites. We integrated the weather service's API for obtaining weather data. The algorithms' accuracy was verified by comparing the results from the experiments to the actual data. The results show that in short-term prediction scenarios, ANN and SVM give good results while LSTM and ANN used in long-term prediction but the results of ANN are not that promising as compared to LSTM.

Table of Contents

List of Figures

List of Tables

CHAPTER-1
Introduction

Traffic flow prediction has been discussed several times with different machine learning approaches. In this project, we did a review of different researchers and the methodologies which have been used for prediction. Many algorithms show the excellent results on the given data sets. Those datasets have collected from different sources. To get accurate information about current and future traffic flow there are many applications such as vehicle navigation devices, congestion management, vehicle routing, and much more application have been introduced to guide the public on the road but the problem is to get real-time data on the spot and helps the users to plan their routs according to the situations on the road but the main problem to get information about traffic flow which are not well equipped with traffic sensors and many other factors that effect to get data such as bad weather conditions. Generally, there are two ways of traffic flow prediction, such as Short Term and Long-Term Traffic Flow Prediction probably long-term algorithms may be cannot provide accurate prediction results because this mechanisms predict on hourly basis such as twelve hours or twenty- four hours data results, as well as short term mechanisms, provide more good results because they give results in terms of minutes such as a five to fifteen minutes or more precise. So in this way, the short-term time interval can give more accurate prediction values. So our model has been trained within a maximum time interval of one hour to give prediction results. To train this prediction model, we have gone through many researchers' contributions in this machine learning area, which have discussed in the literature review section. To meet all these problems, many researchers make a lot of effort by using different machine learning algorithms to predict traffic flow.

Purpose of Traffic Flow Prediction

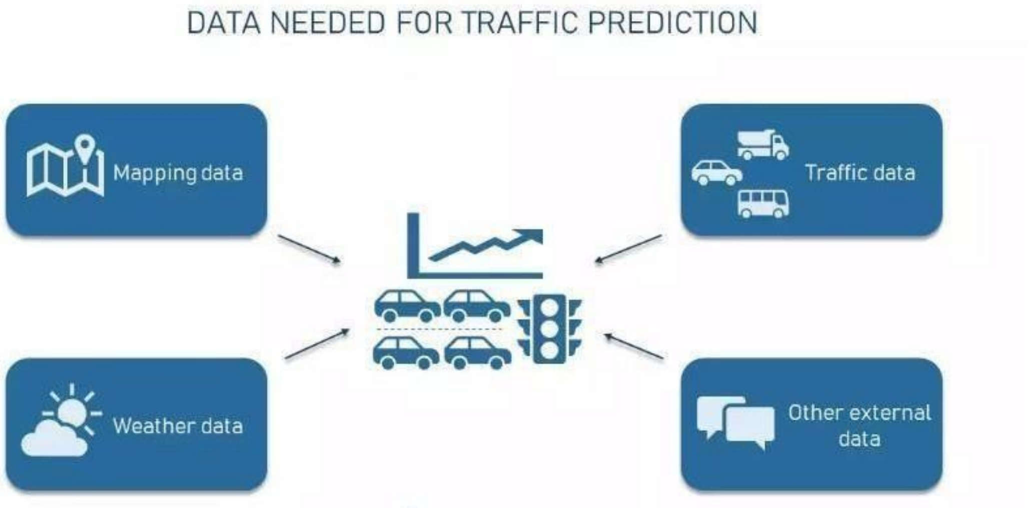
Most of the traffic data reports are actual time, but sometimes it is not so favorable because we use this report when we plan which route we should go. Assume that we are going to office in working hours and we at traffic information and select the best or shorter route to reach our destination but traffic congestion occurred. The issue is to get actual-time information about traffic comes when to resolve this issue by using forecasting? It may be great, but what causes can affect traffic conditions? We need to analyze it. Many causes can affect traffic conditions. The present and ancient traffic condition can be considered to predict. These suggestions are very simple, if traffic is so heavy right now, also acceptable is that after ten or twenty minutes the traffic situation would be same ancient traffic situation, we have indicated the traffic situation on the same day and time, for instance, traffic condition on two Mondays remain same at sane time. Different weekdays and weekends may behave in different traffic situations, and maybe they can also affect traffic conditions. There have been a lot of joint efforts to enhance and down traffic situations; still, there are many chances of progress. Devoted traffic routing system in coordination could be contributing to the reduction of traffic congestion and transport expenditures. With an increasing cost of gasoline, the demand for an efficient routing system to reduce traffic jams is very necessary. In the past years, intelligent transportation systems(ITS) have made great achievements. Intelligent transportation systems help to in-creases the ability of the road traffic system and fix all traffic issues by using new technologies. To achieve rising transportation ITS need to apply for usefultransport infrastructure. One of the most significant demand of ITS is to would be able to predict the traffic truly there are three types of data in a traffic system that are ancient data, present data and short-term calculated data. The capability to predict the transport values such as speed, travel timeor flow, based on real- time data and historical data, collected by various vehicles detector sensors Prediction of traffic variables such as volume, speed, density, travel time, headways, etc. is important in traffic planning and design operations. Various methods are reported in the literature for prediction of traffic parameters such as time series analysis, real-time method, historical method, statistical methods, and machine learning, etc. It becomes essential to understand the working processbehind each of these methods to know the limitations and advantages associated with them. The traffic flow prediction has got a lot of care from transport management and crowded areas of the city management department with the usage of information technology. The goal of traffic flow prediction is to deliver real-time transport data. Whatever to optimize, the traffic on the roads of city areas becomes complex and couldn’t control very well, so such kind of systems are not sufficient for prediction. Therefore, research on traffic flow prediction plays an important role in ITS Systems and Traffic Management Systems.

Problem Statement

To overcome the problems associated with historical, and time series we have to develop a traffic flow prediction model by using machine learning approaches such that SVM and ANN by using these algorithms, we will be developing a graph based prediction system through these users can have interaction with the system and collect the information about current situation of traffic as well as also can check the traffic flow from one to next twenty-four hours of a days with the time interval of different hour data. In this way they may know the weather effects and conditions of the roads that how much traffic will be on which road in the next twenty-four hours, they can also see accidentals records of number of vehicle's and how much chances can be occur for accidents on which road so our system may help them to make their planes that which route or road they should select to make their travel easy.

Approach

Traffic is influenced by many factors, and you should consider all of them to make accurate predictions. So, there are several main groups of data that you'll have to obtain.



Mapping data. First of all, you need to have a detailed map with road networks and related attributes. Connecting to such global mapping data providers as Google Maps, TomTom, HERE, or OSM is a great wayto obtain complete and up-to-date information.

Traffic information. Then, you'll have to collect both historical and current traffic-related information such as the number of vehicles passing at a certain point, their speed, and type (trucks, light vehicles, etc.). Devices used to collect this data are

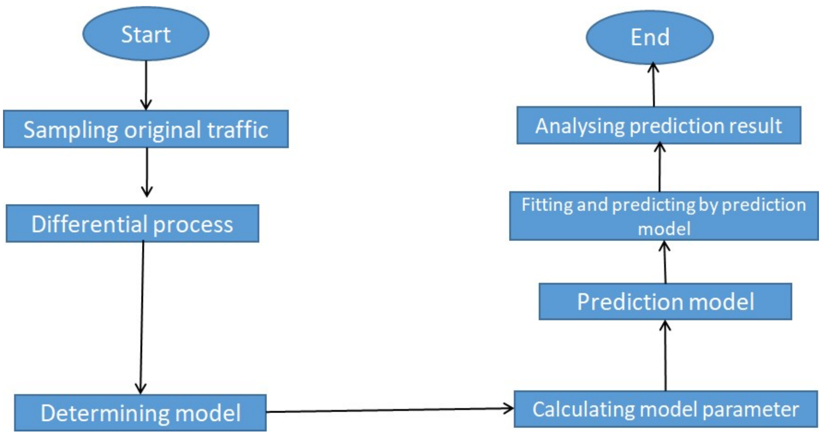
- loop detectors,
- cameras,
- other sensor technologies.

Fortunately, you don't have to install these devices all over the place on your own. It's easier to get this information from the afore mentioned providers that gather data from a system of sensors ,diverse third-party sources, or make use of GPS probe data.

Weather information. Weather data (historical, current, and forecasted) is also necessary as meteorological conditions impact the road situation and driving speed. There are lots of weather data providers you can connect to — such as Open Weather or Tomorrow.io.

Additional data on road conditions. There are external data sources that can provide important information that impacts traffic. Think social media posts about sports events in the area, local news about civil protests, or even police scanners about crime scenes, accidents, or road blockages.

Flow Diagram and System Architecture



CHAPTER-2
Literature Survey

Traffic congestion and resultant issues are the banes of our times, more so in developing countries with inadequate infrastructure. Traffic congestion impact in terms of lost productivity, energy costs, health risk cost through pollution, safety costs, and more are grossly underestimated and present a pressing opportunity to improve the standard of living and quality of life in general (Ahmed Yasser & Kader, 2017). The problem of traffic flow prediction is the accurate estimation of traffic flow in a given region at a particular interval of time in the future. The issue of traffic flow prediction has complex non-linear spatio-temporal dependencies and dependencies on external factors such as weekends, holidays, weather, events, road conditions, and more. Traditionally, traffic prediction models used statistical methods (Lee & Fambro, 1999; Van Der Voort et al., 1996; Williams, 2001; Williams & Hoel, 2003). However, due to the complex dependencies of traffic flow data, researchers have eventually moved from statistical methods to machine learning (Dong et al., 2018; L. Liu et al., 2019; Lu et al., 2020) and deep learning architectures (Sun, Wu, Xiang et al., 2020; Tampubolon &Hsiung, 2018; Yi et al., 2017) to solve the problem. Previous review papers addressed this shift from shallow Neural Network (NN) architectures to Deep Neural Network (DNN) architectures for traffic flow predictions (Ali & Mahmood, 2018; Nguyen et al., 2018; Yin et al., 2020).

Summary of related works

Table 1: Summary of previous researches

Author	**Parameters used**	**Data set**	**Simulation Model**	**ML /DL Algorithm used/Comparative Model**	**Prediction Error Evaluation Criteria**
Kumar, B. R., et al. (2020) [1]	Volume, Travel Time, Speed, Distress Rating, Road width.	videography survey along the Nizamet	ANN (R Software and MATLAB)	SVR ANN	R_2 MAE MSE RMSE ChiSquare

Kumar, K., et al (2013) [2]	Time, speed, Day (Monday to Friday only)	Data samples were collected using video cameras	ANN Model	Artificial Neural Network (ANN)	MSE NMSE MAE ChiSquare
Kim, Y. J., et al (2015) [3]	Environmental variables (e.g., average straight line, number of crosswalk s, bus stops), traffic volume, travel time, speed and Weather Condition .	ITS detectors, GIS and MIS data bases	Multifactor pattern recognition model (MPRM).	Gaussian Mixture Model (GMM)clustering with an artificial neural network (ANN).	MAE
Lana, I., et al. (2016) [4]	Speed, Occupancy or TF, No. of vehicles in the specific time, Day of the Week, Day Type (Working, weekend,	Data from road sensors. Historical Weather data	DBSCAN model	density-based clustering algorithm (DBSCAN)	
Salamanis., et al. (2017) [5]	All days of the week, weekdays, Only weekends. All hours, Normal hours, Abnormal hours	Performance Measurement System (PeMS): The traffic dataset & the incident dataset, GPS- Vehicle Detection Stations (VDS),	Density-Based Spatial Clustering of Applications with Noise (DBSCAN) algorithm evaluation	The k- Nearest Neighbour (KNN), Support Vector Regression (SVR),	RMSE NRMSE

			framework in C++	Autoregressive integrated moving average (ARIMA) model from Time series analysis was trained and tested. Densitybased spatial Clustering of Applications with Noise (DBSCAN) algorithm	
Wu, Y., et al. (2018) [6]	weekly/daily periodicity and Spatial & Temporal feature	PeMS	Hybrid model- Deep neural networks (DNNBTF) model	Convolutional neural network (CNN)and Recurrent Neural Network (RNN)	MAE MRE RMSE
Ma, D., et al. (2018) [7]	Dayof the week, Historical Data Clustering based on workdays and weekends.	sensors on an arterial road	Convolutional Neural Networks and Long short-term memory (CNNs LSTM) model.	deep neural network (DNN)	APE MAPE
Rahman, F. I. (2020) [8]	Day, day type (holiday, working, weekend), Clock time, weather data (sunny, not sunny, raining, not raining) precipitation value and	Transportation Infrastructure Ireland (TII) www.accuweather.com and www. wundergrou nd.com.	All the three algorithm models developed	KNN SVM ANN	R_2 MAPE

	temperature value.				
Jia, T., et al. (2020) [9]	spatiotemporal patterns, including recent, daily, weekly, Workday, holiday	GitHub Historical Data taxicab trajectory data from Wuhan, China.	Python	Deep neural networks (DNN)	RMSE MAE
Tselentis, D. I ., el al (2015) [10]	Travel speed Spatial data Volume Weather data	Camera	ARIMA	ARIMA Bayesian model	RMSE MAPE

Algorithm Used

Artificial Neural Network (ANN): Although there are numerous tasks that computers can perform more quickly than people, such as computing square roots or instantly retrieving a web page, our extraordinary brains still have an edge in certain areas. Artificial neural networks (ANNs) are the solution to make computers more human like and assist robots in reasoning more like people. ANNs are inspired by the structure of the brain.

It is challenging to overcome the constraints of constant models since traffic flow is unpredictable and nonlinear. Methods of statistical machine learning rapidly became the norm. The most prestigious and popular analysis method is the non-constant approach. As the general design of a machine learning system in traffic engineering, artificial neural networks (NN) are typically used to address this drawback. Smith and Demetsky created a NN model that was compared to traditional prediction methods, and their findings suggest that the NN performs better than competing models under peak conditions.

Traffic congestion prediction: Another study has been set up to assess traffic flow utilising neural networks and hybrids of several methodologies. For instance, in the study by Vlahogianni et al., traffic patterns were discovered by grouping them and using neural networks to determine traffic flow. Through networked traffic on the roadways, machine learning algorithms can forecast short-term transport obstruction. The various potential systems of these strategies for progressive prediction algorithms in the combined circumstance served as inspiration for this.

Machine learning approaches: Machine learning techniques have been used to create LSTM-based prediction models, which require constructing networks for training or structures, as well as prediction and prediction implications. Dealing with prediction errors that could happen throughout the prediction process is another objective of deep learning techniques. Big data gathered from the performance measurement system has been subjected to the application of the mentioned methodology (PEMS). The studies demonstrate that, in comparison to shallow machine learning techniques, the LSTM model offers a wide range of capabilities and good performance results.

Models and Architectures

In this section, the specific chosen models are discussed in detail. The reasons for choosing ANN and SVM models are explained and this section also includes the completeimplementation details with the frameworks for searching best performing model parameters.

The data models that are discussed in this section are:-

Long Short-Term Memory (LSTM)

Long time series have long data dependencies. To learn these long-term dependencies the conventional neural based network is not enough to do the job. LSTM was first introduced two decades ago for language processing where it proved to be useful in exhibiting better performance in memorizing the long term dependencies in the data. An LSTM is a memory block structure controlled by memory cells through their respective input, output forget gates and peepholes connections.

$$iitt = \sigma\sigma \,(\, xx^{tt}\, WW_{xxii} + htt{-}1\; WW_{hii} + PPtt{-}1 WW_{ccii} + bb_{ii} \,) \tag{1}$$

$$AAtt = \sigma\sigma \,(\, xx^{tt}\, WW_{xxrr} + htt{-}1\; WW_{hrr} + PPtt{-}1 WW_{ccrr} + bb_{rr} \,) \tag{2}$$

$$DDtt = \sigma\sigma \,(\, xx^{tt}\, WW_{xxmm} + htt{-}1\; WW_{hmm} + PPtt WW_{ccmm} + bb_{mm} \,) \tag{3}$$

$$PPtt = AAtt \cdot PPtt{-}1 + iitt \cdot \tanh(\, xx^{tt}\, WW_{xxcc} + htt{-}1\; WW_{hcc} \quad) + bb_{cc} \tag{4}$$

$htt = DDtt \cdot \tanh(PPtt)$ (5)

$yytt = WWyyh\ htt + bbyy$ (6)

xx^{tt} is the feature input to the memory unit whereas $iitt$, $AAtt$, $DDtt$, htt, $PPtt$ represents the output of the input gate, output of the forget gate, output of the output gate,the final cell state output and the final memory unit output, respectively. $WWxxii$, $WWxxrr$, $WWxxmm$ represents the weights between the input layer and input gate, input layer and forget gate and input layer and output gate, respectively. Similarly, $WWhii$, $WWhrr$, $WWhmm$ are the weights assigned between the recurrent hidden layer and the input layer, forget gate and the output gate, respectively. Likewise, as the subscript suggests,$Wccii$, $WWccrr$, $WWccmm$ the weights associated with the cell state and the input gate, forget gate and the output gate, respectively. All the variables represented by bb are the associated with each of the gates as given in equations (1 – 4). $\sigma\sigma$ represents the sigmoid activation function used. The hidden recurrent unit output htt is passed from the previous LSTM memory unit to the next LSTM unit and from final LSTM memory unit of one layer to the next layer memory unit as an input. THE LSTM model layers are trained using back propagation for different optimisers and layer parameters and the best performing parameters are chosen as the final model forecasts. Like CNN model one max pooling layer is inducted. And every LSTM layer issucceeded by a batch normalisation layer that normalises the batch vectors during each training iteration. Equation 4 represents the overall model output when iteratively calculated by following the equations from (1 – 6).

Convolutional Neural Network (CNN)

Convolutional Neural Networks (CNNs) belongs to the deep learning family and it helps exploit the spatial structure of data (e.g., images) to learn the features of the data for the finalmodel to learn something useful. CNN helps to learn the local features in the data. With having more hidden layers and additionally the convolutional layers at the input. A typical CNN has three components, convolutional layers, pooling layers and a fully connected layer. The CNN inherits all the features of an FFBNN model except that convolutional layers are applied at the start of the normal ANN model and pooling layers are applied in between the ANN layers. Convolution is an operation between two functions, continuous or discrete which in practice has the effect of filtering one of them by another. The general convolutional operator when applied two discrete functions AA and gg can be given as the summation of the following form:

$$(AA * gg)[ii] = \sum_{m=-\infty}^{+\infty} f[m]\, g[n-m] = \sum_{m=-\infty}^{+\infty} f[n-m]\, g[m] \quad (7)$$

The input for CNN is an image representing one state, the pixel values of which represents the scaled input values. The convolutional layers act as a filter which when built into the model does emphasize on certain characteristics of the input feature vectors. It behaves like an automatic custom detector of feature patterns to create a feature map. For an input feature matrix, the commonly applied functions are of two to five elements per dimension and are declared zero on the remaining elements, when strides onto the input feature matrices. The resulting small matrices which represent the groups of filter functions are called kernels. At a given position of the convolutional kernel, the element-wise multiplication of each kernel cell value and the corresponding feature value that overlaps the kernel cell to take the sum of that. For the kernel stride MM of width and height h, convolutional output xx, input ww, the kernel outputs or sub matrices when slid on the input are given as:

$$hii, = \sum^{mm}kk{=}1\sum^{mm}mm{=}1ww(kk\ ,mm\ )\ xx(\ ii{+}kk{-}1\ ,jj{+}mm{-}1) \qquad (8)$$

Pooling layers after the convolutional operation makes the CNN output as translational invariant. Two pooling mechanism are commonly used as average and max pooling.

Average pooling or batch normalisation is used in the CNN model in this thesis. The max pooling operation lists the maximum values as outputs from all the values input by the kernel operations if they fall into the kernel range compensating the number strides used forsliding the kernel. This is mathematically given as below:

$$hii, = \max \{\ xx(\ ii{+}kk{-}1\ ,jj{+}mm{-}1)\ \forall\ 1 \le kk \le MM\ DDiiPP\ 1 \le GG \le MM\ \} \qquad (9)$$

$$yyii, = AAaa\ \sum(\ hii,jj\ + bbii,jj) \qquad (10)$$

The CNN model final output $yyii,$ is given by equation 10 where $AAaa$ is the output layer activation function and $bbii,jj$ is the bias term.

Deep Belief Network (DBN)

The type of deep neural network (DNN) with at least some hidden layers and a significant number of hidden units in each layer. The simplest DBN is made up of stacks of Restricted Boltzmann Machine (RBM) models with a neural layer at the top as the output layer. A typical DBN is trained using a layer by layer greedy algorithm for the supervised data. RBM is an energy based model. Each RBM unit has two layers, visible layer DD and the hidden layer h, both of which are connected by untrained weights. In the stack of RBM the hidden layer of previous RBM is the visible layer of the next RBM. The RBM parameter set $\theta\theta = (ww, bb, DD)$ where $wwiijj$ represents weights between layer $DDii$ and hjj and the biases associated with each of the layers are $bbii$and $DDjj$ respectively, as shown in figure The RBM energy function is given as:

$$EE\ (DD, h\ |\theta\theta) = -\sum ii\ bbiiDDii - \sum jj\ DDjjhjj - \sum ii \sum jj\ wwiijj\ DDii\ hjj \quad (11)$$

Fromthis the joint probability distribution between the hidden and visible layer is given as:

$$p\ (v, h\ |\theta) = \frac{\exp(-E\ (v,h\ |\theta\ ))}{\sum_{v,h} \exp(\ -E\ (\ v,h\ |\theta\ )\ )} \quad (12)$$

And the marginal probability distribution of layer DD is given as:

$$p\ (v|\theta) = \frac{\sum_{v} \exp(\ -E\ (\ v,h\ |\theta\ )\ )}{\sum_{v,h} \exp(\ -E\ (\ v,h\ |\theta\ )\ )}$$

To obtain the optimal parameters for the set $\theta\theta$ for a given data vector DD, the derivative approach is used. For this the gradient log-likelihood estimation is calculated as below:

$$\frac{\partial \log p\ (\ v\ |\theta\ )}{\partial\ w_{ij}} = \langle\ v_i\ h_j\ \rangle_{data} - \langle\ v_i\ h_j\ \rangle_{model}$$

$$\frac{\partial \log p\ (\ v\ |\theta\ )}{\partial\ a_j} = \langle\ h_j\ \rangle_{data} - \langle\ h_j\ \rangle_{model}$$

$$\frac{\partial \log p\ (\ v\ |\theta\ )}{\partial\ b_i} = \langle\ v_i\ \rangle_{data} - \langle\ v_i\ \rangle_{model} \quad (14)$$

Where $\langle\ .\ \rangle$ is expectation of the distributions. There are no corresponding connections between the RBM layer units themselves. So, the layer distributions are easily estimated though conditional probability distributions, given as:

$$p\ (h_j|v, \theta) = \frac{1}{1 + \exp(-\ \sum_i w_{ij}\ v_i -\ a_j)}$$

$$p\ (v_i|h, \theta) = \frac{1}{1 + \exp(-\ \sum_j w_{ij}\ h_j -\ b_i)} \quad (15)$$

The weights in the RBM components are fine-tuned by the contrastive divergence (CD) by default though fast and greedy unsupervised method, and with one additional layer ofneuron at the end overall model weights are trained with the back propagation using supervised learning approach. An activation function is also used in the last output layer.

Support Vector Regressor (SVR)

Support Vector Regressor (SVR) is chosen to deal with the nonlinear data prediction with its capability to fit regression functions to the set of data points. SVR model is a non-parametric model and can be applied without any prior data knowledge with ease. The SVR is a classical statistical theory-based learning models that works implements the structural risk minimisation principle from computational learning theory. It works like a pattern recognition; the basic aim is to map input data vector xx into a high dimensional feature space FF using a non-linear mapping function Φ. The linear regressions are carried out in higher space F which corresponds to the non-linear regression in the low dimensional input space. The corresponding regression function is given as:

$$AA\ (xx) = (ww\ .\ (xx)\ ) + bb \qquad wwiiDDh\ (\ \Phi\Phi\ :\ SS^{nn} \rightarrow FF,\ ww\ \epsilon\epsilon\ FF\ ) \tag{16}$$

ww represents the vector in the feature space Φ(x) is the mapping function input x and b is the threshold. The mapping function is usually a kernel function. Four different kernels linear, poly with third order degree, sigmoid and radial basis function (RBF) considered. Replacing the dot product with kernel function enables the higher dimensional feature spacemapping easy. RBF is usually among the most popular choice for nonlinear mapping, chosen because of its robustness short state predictions and is defines as:

$$kk\ (xx, yy) = DDxx(\ -\gamma\gamma\ |\ xx - yy|^{2}\ ) \tag{17}$$

$\gamma\gamma$ in equation 17 represents – the gaussian bandwidth. With the aim to find the optimal weight ww and bias . To consider the regression problem the flatness of weights and the error generated though empirical risk estimation process are considered. The ww value is optimized by minimizing the sumof empirical risk $SSeemm(AA)$ and the complexity term $|ww|^{2}$. The regression function is given as:

$$R_{ref}(f) = R_{emp}(f) + \frac{\lambda}{2}\ |w|^{2} = C\ \sum_{i=1}^{N} \Gamma\ (\ f(x_i) - y_i\ ) + \lambda\ |\ w\ |^{2} \tag{18}$$

Where C is a curve fitting number usually defined beforehand, N is the sample size and $\Gamma\Gamma$ is the loss function with $\lambda\lambda$ being the regularization constant. Different loss functions are

$$minmise\left(\frac{1}{2} \sum_{i,j=1}^{N} (\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*)\ k(x_i , x_j) - \sum_{i=1}^{N} \alpha_i^* \ (y_i - \varepsilon) - \ \alpha_i (y_i + \ \varepsilon) \right)$$

Given that $\quad \sum_{i=1}^{N} \alpha_i - \alpha_i^* = 0 \quad \alpha_i , \alpha_i^* \epsilon \ [0, C]$

considered which when input into equation 18, it can be reduced to a quadratic statistical problem defines as:

$\alpha\alpha ii$ and $\alpha\alpha ii^*$ are Lagrange multipliers and are found throughthe constraints of equation 19.

$$ww = \sum_{i=1}^{n} (\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*)\ \Phi(x_i) \tag{19}$$

Equation 19 represents the weight term in terms of the data which when input back into the original equation 16 gives it the form given as:

$$AA(x) = \sum_{i=1}^{n} (\alpha_i - \alpha_i^*)\ k(x_i, x) \tag{20}$$

CHAPTER-3

Proposed System

For the traffic flow prediction, we use three methods, that is, Support vector machine (SVM), Long short-term memory (LSTM) and Artificial neural network (ANN), that have been considered for a considerable amount of timewhere SVM is used for short-term prediction , LSTM is used for long-term prediction while ANN is used for both long-term and short-term prediction.

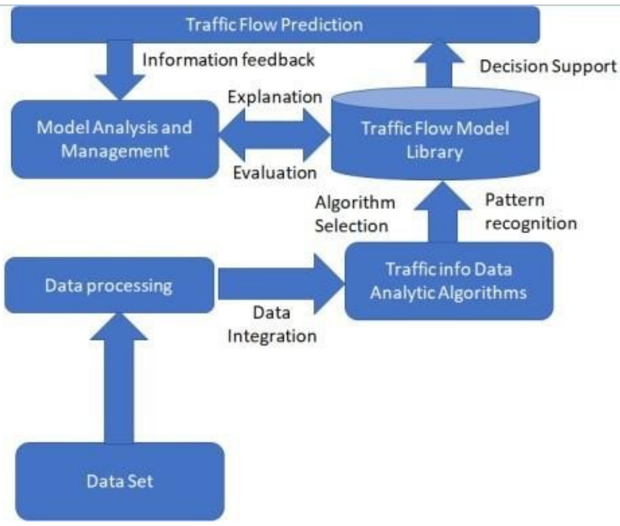


Fig-3 Flow Chart

Data Set - For this proposed system dataset collected from PEMS site for the implementations of approaches that we are going to used to show the output in our predicted system. Data set contains the data of vehicles that has been passed from the road with the help of the vehicles detected sensors and road sensors. By preprocessing the data that has been aggregated over a period of one to twenty-four hours in order to calculate traffic flow prediction, the undesirable data that was present during the extraction process from the vehicles detected sensors and road sensors has been removed.

Artificial Neural Network -ANN helps for the traffic flow prediction by splitting the whole data into two sets, that is, train data set and test data. Train data set is basically used to train the model. This model works in three different layers named as input layer, hidden layer and the output layer(fig 2).
ANN trained the data set for short time interval predictions which takes the three criterion, that is, hours, days and different vehicles. After preprocessing both the data setswe got a accuracy fluctuated from 0.85 to 0.76 for test information and 0.99 for preparing information. The structure of the ANN model with its three layers for predicting a data set is depicted in the following “Fig-”

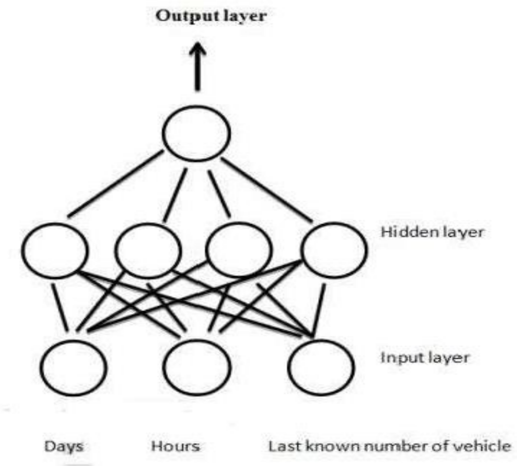


Fig 4 ANN layer Structure

However, the collected data have also been used to train the ANN algorithm in Python, resulting in a graphic representation of the traffic flow prediction over an hour. Through this method, user can check the traffic prediction by using our system.

Support Vector Machine - The support vector machinecontains large number of formulas and approaches that can be used to forecast future traffic data. To improve the prediction outcome, SVM collaborates with the kernel function. In order to improve the prediction result, SVM collaborates with the kernel functions. because the kernel function may assist SVM in data transformation. Because SVM is a method of supervised machine learning, therefore it requires training data for this purpose. It can function in any dimension. When nonlinear data are used as a training sample, this function makes it possible to move them into high dimensions. These functions have been attempting tofit the equation for the best results.

$$d=\omega\varphi(n)+\varepsilon$$

ω ,n, ε refers to the weight vector , input vector and offset value respectively.
The following two values can be used to minimize theeerrors during training. where c is used only for the penalty

$$\sum i(y-\bar{f}(x))$$

is used to calculate errors terms;

$1/2\omega^2$ is a regular item;

L(d-f(x)) refers loss function, balance function, weight for training, and calculating errors.
ε refers loss function because its values can be affected support-vector.
In the following equation, slack values are included, ξ*. so the collected problem can be converted into the following equations.

$$\min 12|\omega|2+c\sum i(\xi i+\xi^* i) \quad (4) \quad di-\omega\,\varphi(x)-\varepsilon \le \varepsilon+\xi i$$

The Lagrange multiplier terms used as xi and x*i are introduced, and the problem is transferred further into a simple optimization problem of the dual problem.

$$\max\sum(xi-x{*}i)-\theta\sum i(xi+x{*}i)-12\sum i\sum j(xi-x{*}i)(xj-x{*}j)k(ni,nj)-(xi-x{*}i)=0$$

So , $(0\le xi\le C, 0\le x^*i\le C)$

Therefore, we got the final prediction function by computing all the above equations.

$$d = \sum(x\boldsymbol{i} - x{*}\boldsymbol{i})\boldsymbol{k}(n\boldsymbol{i}, n\boldsymbol{j}) + \varepsilon \text{ k (ni,nj)}$$

The above equation is the representation of the kernel function because its play k ofrole for SVM model for the prediction of our collected data,
So, we solved our issues in this manner and attained the desired outcome.

Long-short term Memory - A unique variety of RNN called LSTM is made specifically to learn long-term dependencies. There are various memory blocks that make up the LSTM architect re. Each block has three gates—an input gate, a forget gate, and an output gate—as well as one or more self-contained memory cells. Figure 1 depicts the usual layout of an LSTM memory block with a single cell. An outside input is received by the input gate, which processes the fresh data. The forget gate chooses the ideal time lag for the input sequence by determining when toforget the prior state. The output gate produces output for the LSTM cell using all the calculated results.

CHAPTER-4

Tools and Technology

All the development and experiments will be done on the language Python 3.7.11. The libraries, dependencies, and binaries for data science and machine learning were compiled using Anaconda, an open-source distribution. Python, a well-known dynamically interpreted language, is quick and suitable for real-time processing. Since C-based scientific computing and heavy processing libraries can be easily integrated with python, the language has gained popularity. The extensive community support of Python contributors, which is already growing, is one of the other factors.

ML Implementation Library

The machine learning libraries we will be going to use are Keras3 and TensorFlow4. A high-level machine learning API called Keres is built on top of the TensorFlow framework and is written in Python. Because Keras allows for quick experimentation and can create a sophisticated network from scratch in a matter of minutes as opposed to more time with the TensorFlow library, it is chosen over TensorFlow. Convolutional and recurrent neural networks are supported by Keras, as well as their combination.

Processing speed can be increased by defining the workloads for each CPU or processor core. Additional characteristics of the Keras library include its compatibility with Python, modularit , and ease of extension.

You should consider all the factors that influence traffic in order to make accurate forecasts. There are hence a number of significant data groupings that you need to gather.

Mapping data - First things first, you need a detailed map with road networks and other details. A great way to get complete and current data is to connect to global mapping data providers like Google Maps, OSM[6].

Weather information - Moreover, historical, present-day, and anticipated weather data is required because meteorological conditions affect traffic and speed limits on the road. You can connect to a variety of weather data providers, including Open Weather and Tomorrow.io.

For collecting the dataset we use two open source platform

for the weather and for the vehicles travel in different time intervals on the roads.

Tomorrow.io.- By providing the location, fields such as temperature, wind speed, etc., timesteps ("1h," "1d"), and the start time and end time, you can query weather conditions (limited to historical data layers) and receive a response that includes a historical timeline.

You may get up-to-date weather information for your location, including minute-by-minute

forecasts for the following hour, hourly forecasts for the following 120 hours, and daily forecasts for the following 5 days, using the weather forecast API.
There are several location kinds that we support for the location query parameter:
Latitude and Longitude location= 28.4744° , 77.5040°
City name location = Greater Noida
Pin code – 201310
UP post code- NE1

PEMS - The Caltrans Performance Measurement System provides the data (PEMS). Over 40,000 individual detectors spread across the motorway network in all major urban regions are used to collect data in real-time.
The Freeway Performance Measurement System (PEMS) offers a simple way to get historical and current traffic information. It is a compiled database of data obtained from traffic management centres (TMCs) all around the
state using Caltrans loop detectors. It may be accessed using a typical internet browser and has a number of integrated analytical features to support a range of usage. It promotes the system management goals outlined in Caltrans' Traffic Operations Strategy (TOPS), including relieving traffic congestion, boosting customer safety, fully using the current system, and being demand-sensitive. It can be used as a decision-support tool to keep track of how well highways are doing using metrics that are in line with the state-wide Performance Measurement Initiative, which is overseen by Caltrans. Through Value Added Resellers, it makes traffic information available to the general public (VARs).

CHAPTER-5
Result

This section includes a summary of a few representative outcomes from various application tasks. We provide the current top performance strategies based on research from the literature on various tasks. Fig-5 how's the traffic prediction by ANN whereas Fig-6 shows the traffic prediction by SVM. The following observations are possible:

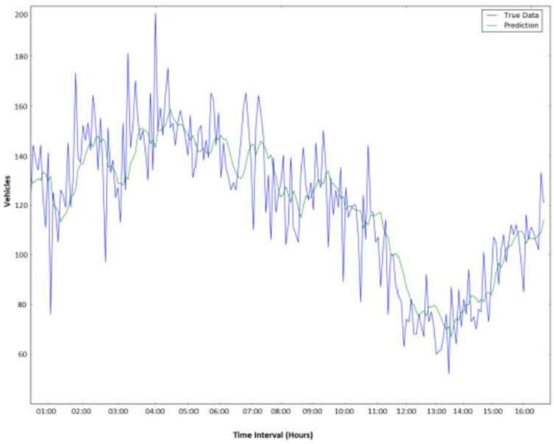


Fig- 5 Res ult by ANN

This following graph show the prediction of traffic using ANN algorithm where lue lines indicates the true data that has been collected from the PEMS and the green line shows the predicted data applying after ANN algorithm of existing data.

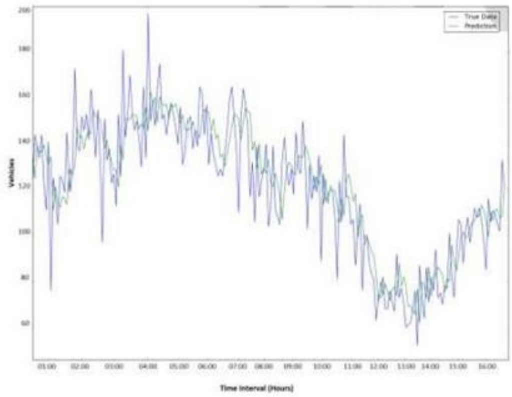


Fig- 6 Res ult by SVM

This following graph show the prediction of traffic using SVM algorithm where lue lines indicates the true data that has been collected from the PEMS and the green line shows the predicted data applying after SVM algorithm of existing data.

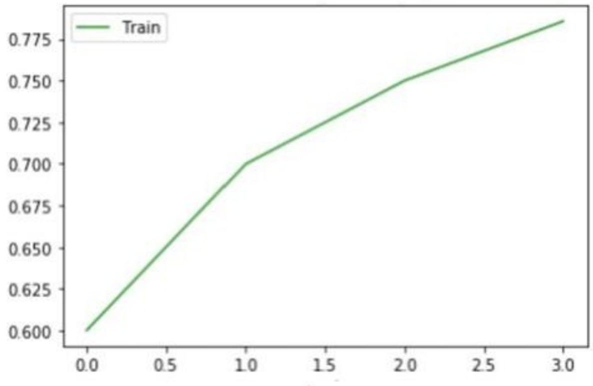


Fig- 7 Result by ANN in Long-term

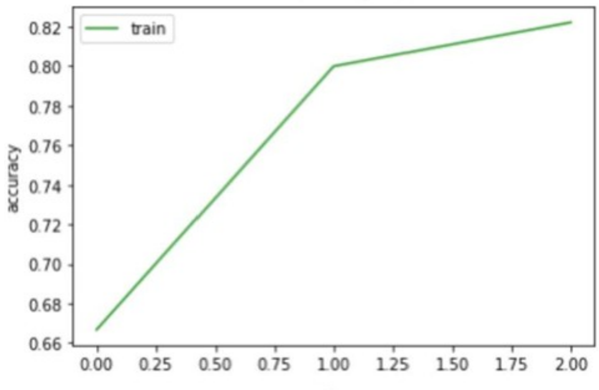


Fig- 8 Result by LST

From Fig-7 and Fig-8 we get to know that the in long term prediction LSTM gives more acuuracy as compare to ANN

CHAPTER-6
CONCLUSION

In the system, it has been concluded that we develop the traffic flow prediction system by using a traffic flow prediction algorithm. By using two existing prediction algorithms,those are ANN and SVM. We try to utilize these models for our system to give the best prediction result on the developed system. The public can take many benefits by using this system because the users can know what the situation of traffic flow on the current situation is and they can also check what will be the flow of traffic on the right after one hour of the situation. This system also helps to check the weather conditions of the roads. In future, we can improve this system by making traffic congestion prediction, and many more factors that affect the management, as well as the flow of traffic, can be considered by using many other deep learning methods, as well as user, can use the system to find which route would be easiest to reach on destination. The system can suggest to the user according to their search.

CHAPTER-7
REFERENCES

1. Kumar, B. R., Chikkakrishna, N. K., & Tallam, T. (2020). Short Term Predictions of Traffic Flow Characteristics using ML Techniques. *In 2020 4th International Conference on Electronics, Communication and Aerospace Technology (ICECA) 4*(1)*,*1504-1508.
2. Wu, Y., Tan, H., Qin, L., Ran, B., & Jiang, Z. (2018). A hybrid deep learning based TFP method and its understanding*. Transportation Research Part C: Emerging Technologies, 90*(1), 166-180.
3. Ma, D., Sheng, B., Jin, S., Ma, X., & Gao, P. (2018). Short-term traffic flow forecasting by selecting appropriate predictions based on pattern matching. *IEEE Access, 6*, 7562975638.
4. Rahman, F. I. (2020). Short-Term Tfp Using Machine Learning-Knn, Svm, And Ann With Weather Information*. International Journal for Traffic & Transport Engineering, 10*(3).
5. Jia, T., & Yan, P. (2020). Predicting citywide road traffic flow using deep spatiotemporal neural networks. *IEEE Transactions on Intelligent Transportation Systems, 22*(5), 3101-3111.
6. Shamshad N, & Sarwr. D. (2020). A review of TFP Based on Machine Learning approaches. *International Journal of Scientific &Engineering Research, 11*(3),1-5.
7. Abdollahi, M., Khaleghi, T., & Yang, K. (2020). An integrated feature learning approach using deep learning for travel time prediction. *Expert Systems with Applications*,
8. Ahmed Yasser, M. Z., & Kader, N. A. (2017). VANET routing protocol for V2V implementation: A suitable solution for developing countries. *Cogent Engineering*, 4(1), 1362802.
9. Ali, U., & Mahmood, T. (2018). Using deep learning to predict short term traffic flow: A systematic literature review. *Lecture Notes of the Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering, LNICST*, 222.
10. Bae, B., Kim, H., Lim, H., Liu, Y., Han, L. D., & Freeze, P. B. (2018). Missing data imputation for traffic flow speed using spatio-temporal cokriging. *Transportation Research Part C: Emerging Technologies*, 88, 124–24.
11. Baldi, P., Brunak, S., Frasconi, P., Soda, G., & Pollastri, G. (1999). Exploiting the past and the future in protein secondary structure prediction. *Bioinformatics*, 15(11), 937–946.
12. Ballard, D. H. (1987). Modular learning in neural networks. *Aaai*, 279–284.
13. Bengio, Y., Courville, A., & Vincent, P. (2013). Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8), 1798–1828.
14. Bengio, Y., Lamblin, P., Popovici, D., & Larochelle, H. (2007). Greedy layer-wise training of deep networks. *Adv. Neural Inf. Process. Syst*, (January), 153–160.
15. Bhatia, J., Dave, R., Bhayani, H., Tanwa1r8, S., & Nayyar, A. (2020). SDN-based real-time urban traffic analysis in VANET environment. *Computer Communications*, 149(October 2019), 162–175.
16. Boukerche, A., Tao, Y., & Sun, P. (2020). Artificial intelligence-based vehiculartraffic flow prediction methods *Computer Networks*, 182, 107484.