12-20-2018

AN APPROACH OF TRAFFIC FLOW PREDICTION USING ARIMA MODEL WITH FUZZY WAVELET TRANSFORM

Sukruti Vaghasia
University of Windsor

AN APPROACH OF TRAFFIC FLOW PREDICTION USING ARIMA MODEL WITH FUZZY WAVELET TRANSFORM

BY:

SUKRUTI VAGHASIA

A Thesis
Submitted to the Faculty of Graduate Studies
through the School of Computer Science in
Partial Fulfillment of the Requirements for the
Degree of Master of Science at the University of
Windsor

WINDSOR, ONTARIO, CANADA

2018

AN APPROACH OF TRAFFIC FLOW PREDICTION USING ARIMA MODEL WITH FUZZY WAVELET TRANSFORM

By

Sukruti Vaghasia

APPROVED BY:

_____

G. Bhandari

Odette School of Business

_____

J. Lu

School of Computer Science

_____

X. Yuan, Advisor

School of Computer Science

October 5, 2018

Declaration of Originality

I confirm here I am the sole creator of this theory and no piece of this proposition has been distributed or submitted for production.

I approve to the best of my insight, my proposition does not encroach upon anybody's copyright nor damage any restrictive rights and their thoughts, methods, citations, or some other material from crafted by other individuals incorporated into my postulation, circulation or something else, and are entirely recognized as per the standard referencing hone. Moreover, I have included copyrighted material that outperforms the limits of reasonable managing inside the significance of the Canada Copyright Act. I affirm to acquire a composed consent from the copyright owner(s) to incorporate such material(s) in my postulation and have included duplicates of such copyright clearances to my reference section.

I pronounce this is a genuine copy of my thesis including any last updates, as affirmed by my thesis advisory group and the Graduate Studies office, and this theory has not been submitted for a higher degree to some other University or Institution.

Abstract

It is essential for intelligent transportation systems to be capable of producing an accurate forecast of traffic flow in both the short and long terms. However, the counting datasets of traffic volume are non-stationary time series, which are integrally noisy. As a result, the accuracy of traffic prediction carried out on such unrefined data is reduced by the arbitrary components. A prior study shows that Box-Jenkins' Autoregressive Integrated Moving Average (ARIMA) models convey demand of noise-free dataset for model construction. Therefore, this study proposes to overcome the noise issue by using a hybrid approach that combines the ARIMA model with fuzzy wavelet transform. In this approach, fuzzy rules are developed to categorize traffic datasets according to influencing factors such as the time of a day, the season of a year, and weather conditions. As the input of linear data series for ARIMA model needs to be converted into linear time series for traffic flow prediction, the discrete wavelet transform is applied to help separating the nonlinear and linear part of the time series along with denoised time series traffic data.

Acknowledgments

Table of Contents

List of Tables

List of Figures

List of Abbreviations/Symbols

AR	Auto Regression
MA	Moving Average
ARIMA	Autoregressive Integrated Moving Average
ACF	Autocorrelation function
PACF	Partial Autocorrelation
AIC	Bayesian information criterion
BIC	Akaike information criterion
ITS	Intelligent Transportation System
ANN	Artificial Neural Network
NN	Neural Network
SVM	Support Vector Machine
ES	Exponential Smoothing
MF	Membership Function
DOM	Degree of Membership
FT	Fourier Transform
WT	Wavelet Transform
STFT	Short-Time Fourier Transform
CWT	Continuous Wavelet Transform
DWT	Discrete Wavelet Transform
SVP	Sharp Variation Point

1. Introduction

Urbanization has revolutionized our lives in many aspects. People prefer to live in metropolitan cities rather than countryside. A higher standard of living facilitates us with an individual vehicle instead of using public transportation. It becomes a prominent cause of more vehicles appearing on the road. This problem further leads to other numerous issues such as delay in travel time, increasing traffic jams and accidents, consumption of more fuel, waste of natural resources, etc. Thus, forecasting of traffic is a significant solution for traffic congestion. Advancement in development of traffic sensors, GPS and location detectors can improve the performance of traffic prediction frameworks and the exactness of forecast techniques. Traffic forecast accuracy improves the performance of various components of transportation systems such as corridor management, transit signal priority control and traveler information system [1].

In recent years, traffic forecast has been viewed as a critical segment in the Intelligent Transportation System (ITS). The primary goal of ITS is to add all significant transportation measures to different modes of transport and traffic management and enable users to be better informed and make safer, more coordinated, and 'smarter' use of transport networks [2]. ITS involve information gathering, preparing, and investigating for a specific end goal to guarantee a compelling decision-making tool [3].

Estimating the flow situation by analyzing the traffic data until the next period with a prediction algorithm is known as traffic flow prediction. It is necessary to carry out the traffic analysis to develop better control and reduce congestion of the traffic flow, which is also accomplished by the intelligent transportation system. Local government agencies collect the historical and real-time traffic data for the study of traffic analysis. Accurate forecasting of traffic flow prediction highly depends on the quality of traffic data. As traffic data can be represented in the form of time series, such data usually carry some noise with it. Here the term "noise" can be denoted as observation error, missing data point, measurement uncertainty and many more [4]. Requiring precision in predicting systems translates to higher accuracy through collecting noise-free data [5].

In the proposed research, a fuzzy method is a crucial part to carry out data categorization as traffic is affected by various events such as a day of week, by weather conditions namely rain, fog, snow and temperature. As mentioned above, considering traffic data as time series data, Discrete Wavelet Transform (DWT) is used to correlate linear and non-linear parts of time series data along with de-noising. Subsequently, Autoregressive Integrated Moving Average (ARIMA) is used to analyze and predict the outcomes of DWT. In the representation of various time series, ARIMA models are widely adapted for precise forecasting. ARIMA considers every time series as linear; however, DWT outweighs this problem by identifying linearity and non-linearity of time series data.

In the following report, Chapter 2 represents detailed insight into the work done previously in a related area of traffic prediction and in different types of time series forecast. Moreover, the prerequisite techniques for the proposed method are discussed in Chapter 2. The proposed approach for traffic prediction is discussed in detail in Chapter 3. The results for traffic prediction achieved through the proposed method is addressed in Chapter 4, along with its experiments and analysis. Chapter 5 includes the conclusion of anticipated impact in the research area.

2. Related Work

In this section, some of the relevant background about recent work on traffic flow prediction is discussed, and technical knowledge of the ARIMA model, fuzzy logic, and discrete wavelet transform are studied in detail.

2.1. Traffic Flow Prediction

"In recent decades, some traffic forecast models have been produced to aid traffic administration and control for enhancing transportation proficiency going from route direction and vehicle navigation to signal coordination. The traffic flow prediction problem can be stated as follows. Let X_i^t denote the observed traffic flow quantity during the tth time interval at the ith observation location in a transportation network. Given a sequence $\{X_i^t\}$ of observed traffic flow data, i = 1, 2..., m, t = 1, 2..., T, the problem is to predict the traffic flow at a time interval ($t + \Delta$) for some prediction horizon Δ." [6].

Researchers explored many techniques to forecast traffic volume, Table 1 depicts essential strategies in the territory of traffic flow analysis and traffic flow prediction by posting their favorable circumstances and impediments.

Table 1: Review of prediction methods

Title	Author	Year	Advantages	Disadvantages
Traffic flow prediction using neural network	M. Jiber, I. Lamouik, Y. Ali, and M. A. Sabri	2018	Deep learning architecture to predict road traffic flow with the ability to reconstruct lost or	Unexpected conditions such as events, holidays, accidents might affect the accuracy

			destroyed data with the predicted values	
Short-Term Traffic Speed Prediction for an Urban Corridor	H. Zhu, B. Yu	2017	Short-term prediction using support vector machine model with spatial-temporal parameters with higher accuracy	single-step prediction model. When the traffic speed is higher than 35 km/h, the prediction accuracy is reduced
Traffic Flow Prediction using Kalman Filtering Technique	Kumar, S. V.	2017	Uses real time as well as historical (2 days) data for prediction	Nothing has been done for missing values.
Time series Decomposition Model for Traffic Flow Forecasting in Urban Midblock Sections	G. Omkar, S. Vasantha Kumar	2017	Multiplicative decomposition technique is used which requires fewer data compares to ARIMA. Both historical and real-time data are used	Missing data and external factors are not considered such as weather and accident.

Improving Traffic Flow Prediction with Weather Information in Connected Cars: A Deep Learning Approach	A. Koesdwiady, R. Soua, F. Karray	2016	Study of traffic flow prediction with the effect of weather data using deep belief networks.	High usage of memory for the training process.
Traffic Flow Prediction with Big Data: A Deep Learning Approach	Y. Lv	2015	Examines historical and real-time data to be inputted in a greedy approach which provides fantastic performance	Takes much time and the data used are produced synthetically
Real-time Traffic Flow Predicting Using Augmented Reality	M. Zhang	2016	Produces better accuracy	Doesn't consider the chaotic situation, data used to predict is generated synthetically
Short-term traffic flow prediction using seasonal ARIMA model with limited input data	S. V. Kumar, L. Vanajakshi	2015	Overcome issue of the sound database for ARIMA model by using past three days' data.	Using past three days data but it might not fetch the weekly or monthly patterns which can be useful information in traffic prediction.

Multi-step Prediction of Volterra Neural Network for Traffic Flow Based on Chaos Algorithm	L. Yin	2012	Considers both historical and real-time data for long-term prediction	Takes too much time as considers many data.

Early research in traffic predictions used different techniques. For example, in the year 1984, Okutani et al. proposed to apply Kalman filtering [7] with two different models to estimate traffic count. Historical traffic data is more commonly used in some prediction models [8] [9], while others relied on real-time traffic information [10]. For example, Rice et al. used present traffic conditions in Random Walk Forest [11] technique to forecast traffic flow.

On the other hand, a method like ARIMA process [23] discloses short-time traffic conditions of a location in the road network based on previous traffic flow of given area. The primary input of predictive models is real-time or historical information, but Chrobok et al. [12] emphasized combining both data to improve final prediction result. Also, Rice et al. [13] incorporated historical data with real-time data for prediction. More attention has been given to research traffic valuation and forecast. However, the restrictions of those models are remaining. For example, transportation systems might be affected by factors such as accidents and weather [14]. Specifically, extreme weather conditions may have a drastic impact on travel time and traffic flow [15] [16] [17]. Khandelwal et al. [18] used fuzzy logic to measure the influence of social factors which affects the stock price prediction. The proposed method will examine the external factors that affect the traffic count data on the road using fuzzy logic. To exemplify, the temperature – if the temperature is good, then the possibility of travellers on the road will be high, which increases traffic count. Other factors such as snow, rain, fog, type of day, the geographical position of an area, planned or unplanned road construction and accident can also affect the traffic data and make the prediction process more complicated. By capturing those

patterns in traffic data, we can compare related factors of real-time data (forecasting data) with corresponding factors of historical data and only matching a set of data should be considered for further process of prediction [19]. A concept of fuzzy logic as a method of processing data is presented by discussing partial set membership opposite to crisp set membership [20].

Traffic flow prediction methodologies can be ordered into three classes: 1) parametric approach; 2) nonparametric approach, and 3) hybrid approach [15]. The principal strategies of the parametric approach are based on models of autoregressive integrated moving average (ARIMA) [11] [21] [22] and Kalman filtering [23] [24] [25]. Traffic flow has a stochastic and nonlinear nature, but these models predict traffic without considering these characteristics which leads to a substantial error in the prediction.

In the second approach, a nonparametric regression [26] [27] is generally utilized procedure. Chang et al. [9] used a k nearest neighbour nonparametric regression model for short-term traffic prediction. The model claims to achieve better results even with historical time-series data that abruptly changes or widely fluctuates. Liu et al. [28] proposed support vector regression to establish the network traffic prediction model, whereas the global artificial fish swarm algorithm (GAFSA) was used to optimize the model's parameters. Artificial neural networks (ANNs) were proposed to predict traffic flow [6] [29]. "However, the problem of local minima still exists in ANNs appropriate procedures. What's more, ANNs, most of the time, use one concealed layer. Simulations have exhibited that one hidden layer would not be adequate to depict the convoluted association between the data sources and yields of the forecast model." [15].

Hybrid approaches [4] [27] [30] have been explored by a few examinations to deal with the obdurate nature of time series in nonparametric models. These methodologies are more versatile while consolidating various procedures. In recent eras, a combination of numerous forecasting models has pulled in consideration of researchers in several areas, especially in time series forecasting. The combination of ANN, DWT, and ARIMA is used in [30] where experiments are conducted on four different time series. Also, the inspiration for using DWT to partition

linearity and nonlinearity is taken from [31], founded by Zhang, who pointed out that time series data contains both linear and nonlinear patterns. The same approach is taken in [32] which is discussing usage of the piecewise method ARIMA model and the SVM model to predict short-term traffic flow prediction. Tang at el. constructed a fuzzy neural network to forecast travel speed for multi-step-ahead based on two-minute travel speed data [33]. Liu at el. projected use of Adaptive Fuzzy Neural Network (AFFN) for a lane changing predictor to predict steering angles [34]. An approach to tactically utilize the unique strengths of DWT, ARIMA, and ANN to improve the forecasting accuracy is mentioned in [35]. The use of DWT and Fuzzy logic to improve the accuracy of traffic flow prediction on hourly data is shown in [4]. More and more research is being conducted combining two or more methods to predict traffic flow.

Evidently, Khandelwal et al. [30] suggested that neither ARIMA nor ANN is universally suitable for all types of time series. All real-world time series contain both linear and nonlinear correlation structures among the observations. Zhang [4] has pointed out this important fact and has developed a hybrid approach that applies ARIMA and ANN separately for modelling linear and nonlinear components of a time series [35]. Moreover, Adhikari et al. [35] indicated a new hybrid approach which has used DWT to convert the time series in the linear and non-linear component. Based on this approach the proposed method will use the advantage of converting time-series data into linear time series data. Further, it will be used in final forecasting model ARIMA to predict the result.

2.1.1. Summary

Different approaches have been developed (e.g., ANN, SVM, ARIMA), but the results are not satisfying. Current approaches have not paid sufficient attention to the quality of data under investigation. ANN is not trustworthy when the training data is not descriptive of the actual pattern, and the proportions of the training data are not sufficient. SVM model is time-consuming and suffering from dimensionality during data analysis [36]. ARIMA is the most common and traditional model with a simple algorithm based on a linear analysis which is not sufficient to

predict stochastic nature of traffic series data. There has been no investigation that tries to link the count of traffic data with information from other sources, such as season, weather, the impact of events (traffic accident, planned or unplanned road construction) and time of day and the day of the week. Table 2 [37] lists the pros and cons of available methods used in time series prediction.

Table 2 Summary of advantages and challenges of classical, ANN-based, and SVR time series prediction methods

Time series prediction method	Advantages	Disadvantages
AUTOREGRESSIVE FILTER	Can be computationally efficient for low order models Convergence guaranteed Minimizes mean square error by design	Assumes linear, stationary processes Can be computationally expensive for higher models
KALMAN FILTER	Computationally efficient by design Convergence guaranteed Minimizes mean square error by design	Assumes linear, stationary processes Assumes process model is known
MULTI-LAYER PERCEPTRON	Not model dependant Not dependent on linear, stationary processes Can be computationally efficient (feed forward process)	Number of free parameters large Selection of free parameters usually calculated empirically Not guaranteed to converge to the better solution Can be computationally expensive (Training Process)
SVM/SVR	Not model dependant	Selection of free parameters usually calculated empirically

	Not dependent on linear, stationary processes	Can be computationally expensive (Training Process)
	Guaranteed to convergence to optimal solution	
	A small number of free parameters	
	Can be computationally efficient	

Researchers working on traffic prediction have used many advanced algorithms and improved accuracy in forecasting of traffic data. As traffic data is a time series, several outside factors are also responsible for final prediction of result which encourages to introduce Fuzzy Logic. No such prediction model has ever focused on removing noise from data to improve the accuracy of traffic flow prediction and DWT is the solution. Also, ARIMA is the most widely used forecasting model until now, but its nature of treating every data input as linear can add some error in final prediction of results while using on time series data. DWT is a more suitable method to denoise the time series data and partition into linear and nonlinear. Combination of Fuzzy Logic, DWT and ARIMA will help to achieve higher accuracy model for prediction.

In the proposed research, fuzzy method is a crucial part to carry out data categorization as traffic is affected by various events such as weekdays, weekends and by weather conditions such as rain, fog, snow and temperature. Traffic data carries some noisy data, and it is essential to implement the decomposition of noise from traffic data. DWT correlates linear and non-linear parts of time series along with de-noising. Then ARIMA is used to analyze and predict the outcomes of DWT. In the representation of various time series, ARIMA models are widely adapted for precise forecasting. ARIMA considers every time series as linear, but DWT outweighs this problem by identifying linearity and non-linearity of time series data.

2.2. Technical Background

2.2.1. Fuzzy Logic Control

In 1956, Professor L. A. Zadeh, University of California at Berkeley, provided the use of fuzzy logic [19]. Fuzzy thoughts and fuzzy logic even exist in our daily life. A classic example to understand fuzzy logic is the answer to a question as "How satisfied are you with their product or service"? The most likely reply would be 'Not very satisfied' or 'Very satisfied', which are likewise fuzzy or vague answers. Such types of questions can be answered easily by a human, but for a computer, these obscure or vague replies cannot be made easily. The only reply expected from computers is either '0' or '1' for 'HIGH' or 'LOW'. Such outputs are known as crisp or classic output. The fuzzy framework can be viewed as a choice to overcome this issue in the realm of Computer. "To implement a fuzzy logic technique to a real application requires the following three steps:

1. Fuzzification – convert classical data or crisp data into fuzzy data or Membership Functions (MFs)
2. Fuzzy Inference Process – combine membership functions with the control rules to derive the fuzzy output
3. Defuzzification – use different methods to calculate each associated output and put them into a table: the lookup table. Pick up the output from the lookup table based on the current input during an application" [38].
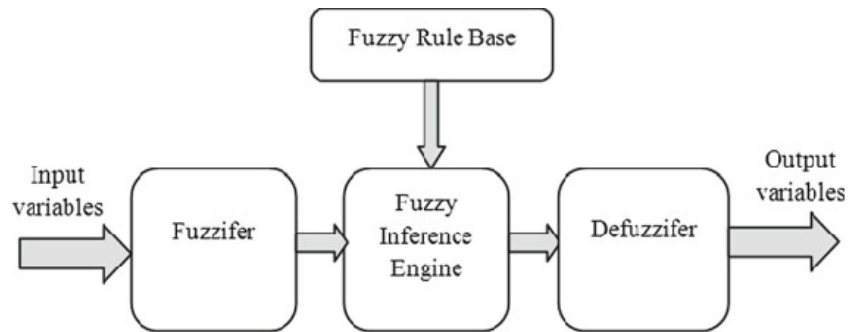
Figure 1: The typical fuzzy logic system[1]

Fuzzy Terminology

a. Fuzzy Sets

Fuzzy sets are a subset of a classic or crisp set. Difference between standard set and fuzzy sets is the boundary of separating the sets. In classical sets, we have a sharp boundary which means a member belongs to the set or not at all whereas in the fuzzy set we have a smooth boundary with members having a certain membership degree. An example [38] is used here for a better explanation. "If the temperature is defined as a crisp high, its range must be between 80 °F and higher, and it has nothing to do with 70 °F or even 60 °F. However, the fuzzy set will take care of a much broader range for this high temperature. Compared with a classical set, a fuzzy set allows members to have a smooth boundary. In other words, a fuzzy set allows a member to belong to a set to some partial degree. For instance, still using the temperature as an example,

the temperature can be divided into three categories: LOW (0 ~30 °F), MEDIUM (30 ~ 70 °F) and HIGH (70 ~ 120 °F) from the point of view of the classical set" [38], which is shown in Figure 1.

(a) Classical set   (b) Fuzzy set
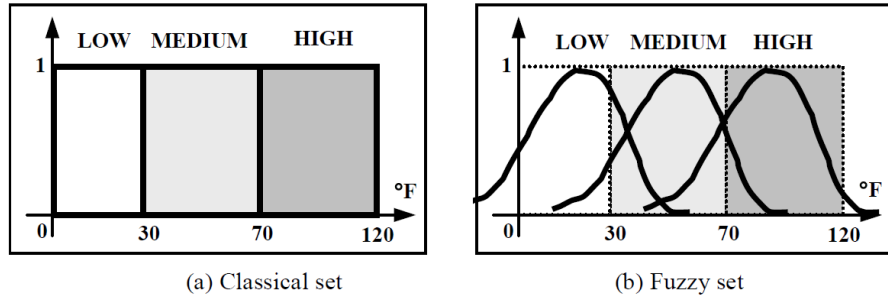
Figure 2: Representations of classical and fuzzy sets

Any temperature value in the classical set can be classified only in specific subset either in LOW, MEDIUM or HIGH. For instance, temperature value 45 °F is MEDIUM for the classical set. On the other hand, in a fuzzy set, this temperature value falls under LOW with some degree of membership (approximately 0.5 degrees). At the same time, it falls under MEDIUM with around 0.7 degrees of membership. A fuzzy set allows a member to have a partial degree of membership and this partial degree membership can be mapped into a function or a universe of membership values [38].

$$\mu_A(x) \in [0,1] \quad (A = (x, \mu_A(x)| \; x \in X)) \tag{1}$$

A fuzzy subset A with an element x has a membership function of $\mu_A(x)$ [38]. When X is finite mapping can be represented as in equation (**2**)

$$A = \frac{\mu_A(x_1)}{x_1} + \frac{\mu_A(x_2)}{x_2} + \dots = \sum_i \frac{\mu_A(x_i)}{x_i} \tag{2}$$

When X is continuing the fuzzy set can be defined as in equation (**3**)

$$A = \int \frac{\mu_A(x)}{x} \tag{3}$$

(a) Fuzzy set union   (b) Fuzzy set intersection   (c) Fuzzy set complement
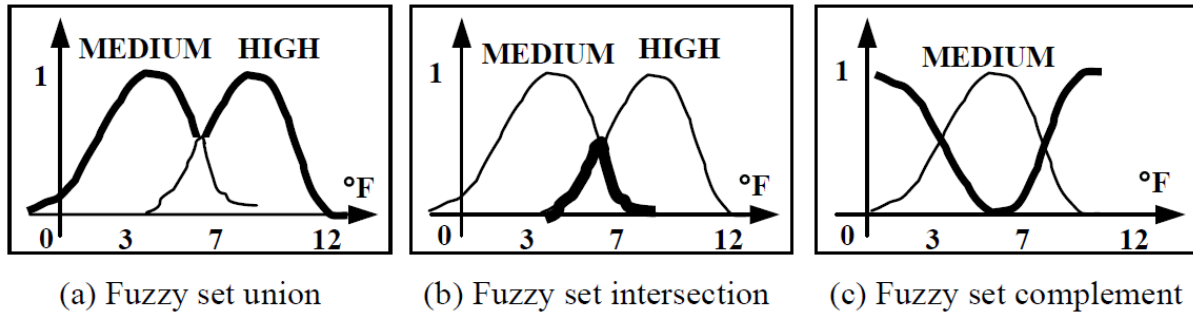
Figure 3: Fuzzy set operations

Fuzzy set operations are like classical set operations where the union is selecting the maximum of a member from the members and intersect is selecting the minimum member from the sets.

$$Union: \mu_A(x) \cup \mu_b(x) = \max(\mu_A(x), \mu_B(x))$$

$$Intersect: \mu_A(x) \cap \mu_b(x) = \min(\mu_A(x), \mu_B(x))$$

$$Complement: A^C = F \setminus A$$

b. Fuzzification and Membership Functions

Fuzzification is a process of converting classical set to a fuzzy set. This process includes calculating the degree of membership for both input variable and output variable to varying degrees and labelling its value with linguistic variables. There are some different membership functions, such as the triangular waveform, trapezoidal waveform, Gaussian waveform, bell-shaped waveform, sigmoidal waveform and S-curve waveform [38]. The exact type depends on the actual applications. For those systems that need significant dynamic variation in a short period, a triangular or trapezoidal waveform should be utilized [38]. For a system that needs very high control accuracy, a Gaussian or S-curve waveform should be selected [38].

14
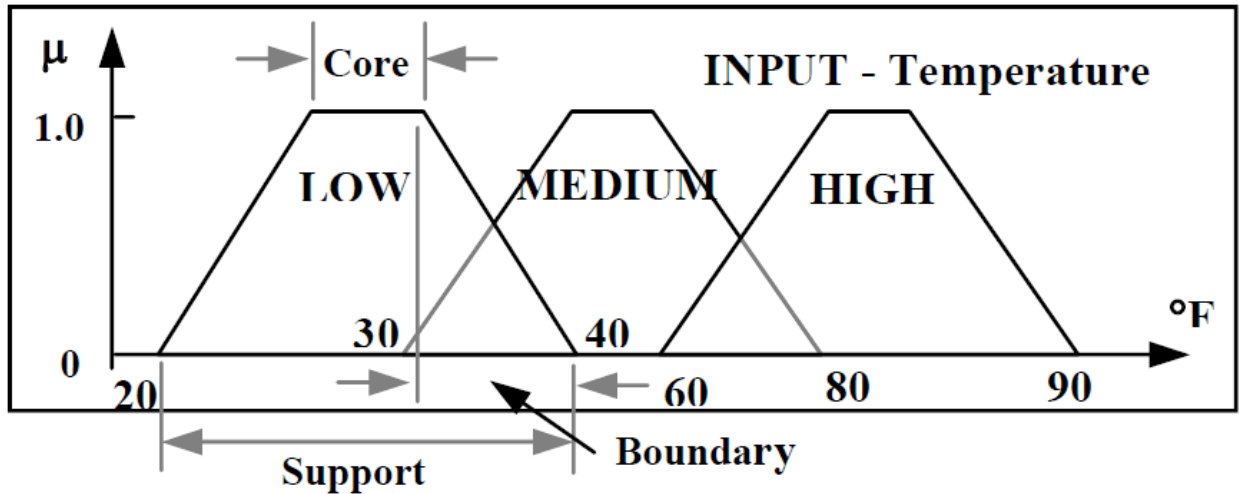
(a) Membership function of the input
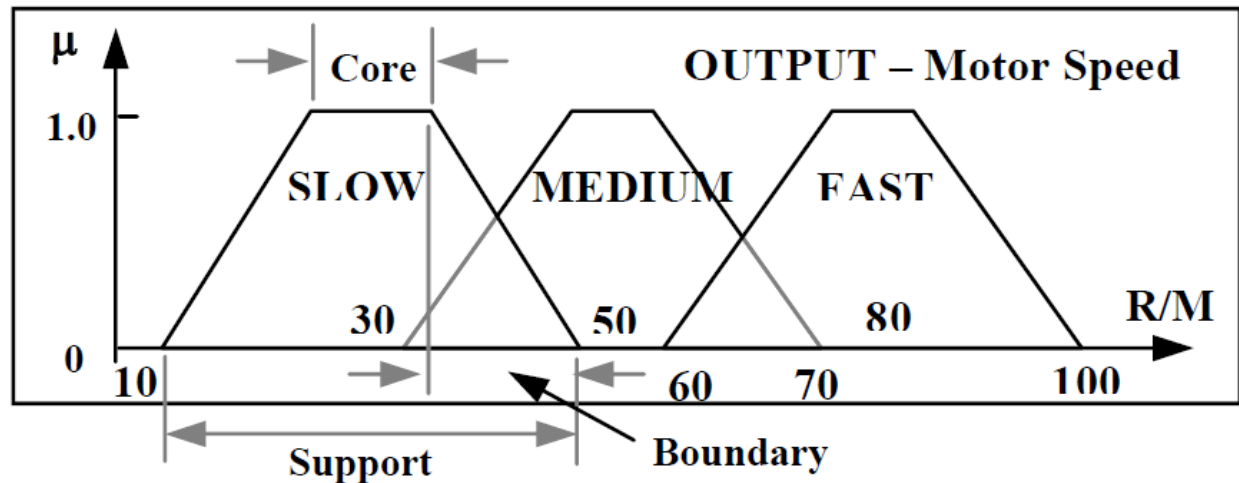
(b) Membership function of the output

Figure 4: Membership function of input and output

"Continuing the temperature example, we divide the range of temperature as follow

Low temperature:	20 °F ~ 40 °F, 30 °F is center
Medium temperature:	30 °F ~ 80 °F, 55 °F is center
High temperature:	60 °F ~ 90 °F, 75 °F is center

To understand more clearly, Figure 4 [38] shows the membership function of these temperatures. For instance, 35 °F will belong to LOW and MEDIUM to 0.5 degrees." [38]. After defining fuzzy membership functions for input and output the next step is to build a fuzzy rule set. Some terminologies displayed in Figure 4 are as follow. The term "Support" can be explained same as a classical set. For example, support for LOW is a set of elements whose degree in membership LOW is greater than 0. The term "Core" is a set of elements whose degree of membership is equal to one which can be compared to a crisp set. The term "Boundary" is a set whose degree of membership between 0 and 1.

c. Fuzzy Rules for inference

The fuzzy rules are general IF-THEN statements described by some expert or knowledgeable human being of the actual applications. A set of IF-THEN statements describing how the linguistic variable maps to a fuzzy set are introduced to derive conclusions. The antecedent of an IF-THEN statement is an elastic condition to derive the knowledge, whose following part is linguistic output. The inference engine of a fuzzy inference system uses this rule set to calculate the value degree of membership. Figure 5 [38] illustrates how the input part or if part of the rule is matched with the defined rules. Here, fuzzy condition LOW and a fuzzy input's degree of membership result in membership of 0.4.
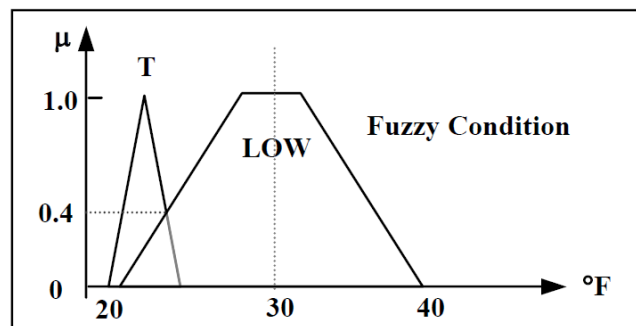
Figure 5: Matching a fuzzy input with a fuzzy condition

16

d. Defuzzification

The process of defuzzification in a fuzzy system is responsible for converting the fuzzy output to crisp value for real application to use. Remember, the fuzzy conclusion or output is still a linguistic variable, and this linguistic variable needs to be converted to the crisp variable via the defuzzification process [38]. In [38] two defuzzification techniques are discussed which are as mentioned below.

i. Mean of Maximum (MOM) Method

This method calculates the average of output with the highest degree. For example, for the linguistic output, 'the heater machine is rotated fast' graphical representation is shown in Figure 6 [38]. A shortcoming of the MOM method is that it does not consider the entire shape of the output membership function, and it only takes care of the points that have the highest degrees in that function [38].
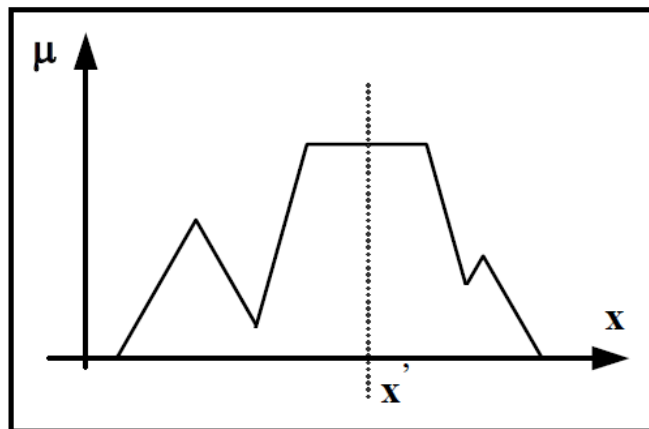
Figure 6: Graphic representation of defuzzification techniques with MOM method

ii. Center of Gravity (COG) Method

The most known and used method in the defuzzification process is COG. The weighted average of the membership function or the center of the gravity of the area bounded by the membership function curve is computed to be the crispiest value of the fuzzy quantity [38]. For example, A graphic representation of COG method for the conclusion: 'the heater motor x is rotated FAST' is shown in Figure 7 [38].
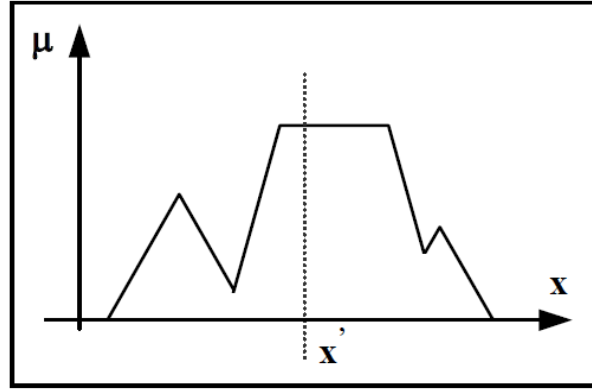
Figure 7: Graphic representation of defuzzification techniques with the COG method

An example of defuzzification with four fuzzy logic rules is described in Figure 8 [38]. The first column indicates the value of membership for temperature. The temperature value (T) of 35 °F intersect at 0.6, 0.8, 0.5 and 0.8 for LOW, MEDIUM, LOW and MEDIUM respectively. Likewise, the second column in Figure 8 shows that a temperature change rate (ΔT) of 1 °F per hour has the membership functions of 1.0, 0.4, 0.4 and 1.0 [38]. Based on equation (**4**) the calculation to get the result is min (0.6, 1.0), min (0.8, 0.4), min (0.5, 0.4) and min (0.8, 1.0), which results as 0.6, 0.4, 0.4 and 0.8, respectively [38].

After having fuzzy output from all the four defined rules the next step is to get the crisp or classical value by performing defuzzification using one of the methods defined above. For this study center of gravity is used as COG is a more reliable method than MOM.

Thus, for a temperature of 35 °F and a change rate of temperature 1 °F per hour, the fuzzy output element y for this input pair is 600 R/M [38].

$$y = \frac{0.6 \times 800 + 0.4 \times 300 + 0.4 \times 800 + 0.8 \times 500}{0.6 + 0.4 + 0.4 + 0.8} = 600 \text{ R/M} \qquad (4)$$
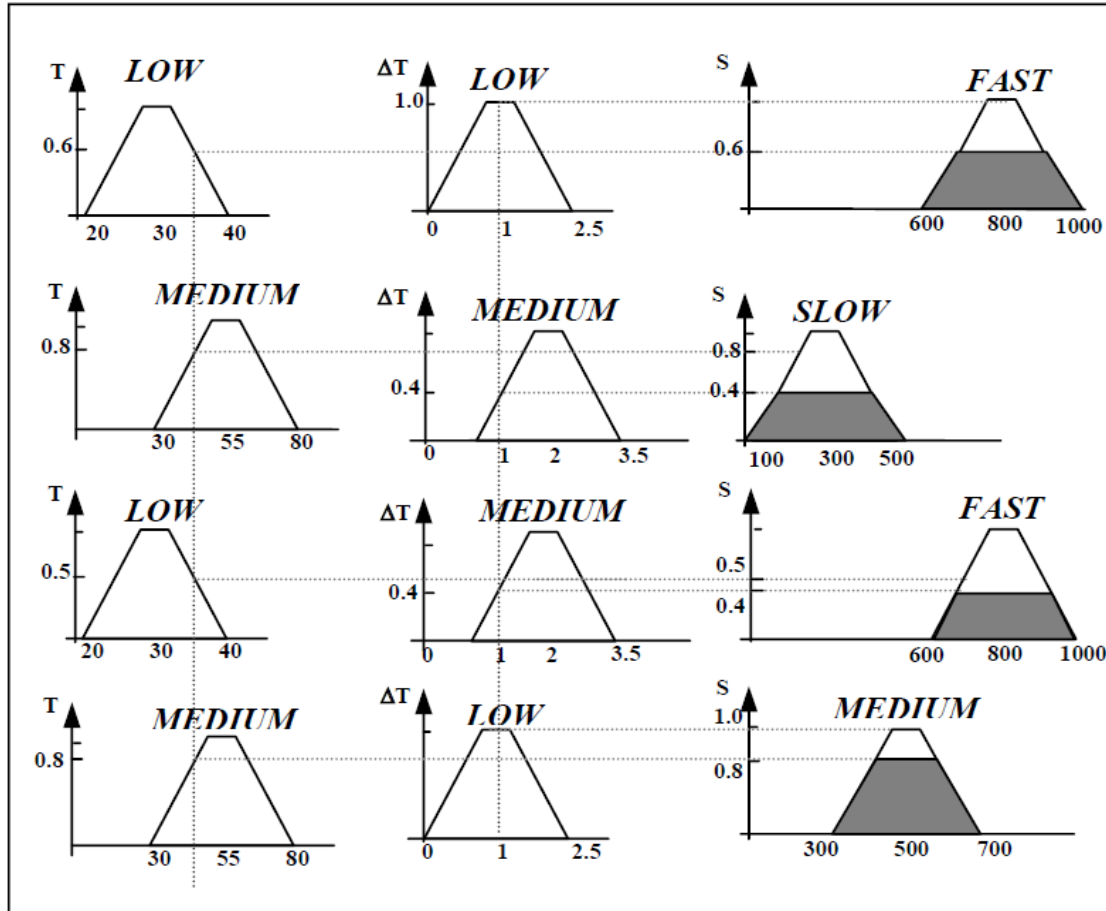
Figure 8: An illustration of fuzzy output calculation

Figure 9 [38] shows the graphical presentation of fuzzy output to crisp output calculated using COG with the help of four fuzzy rules.
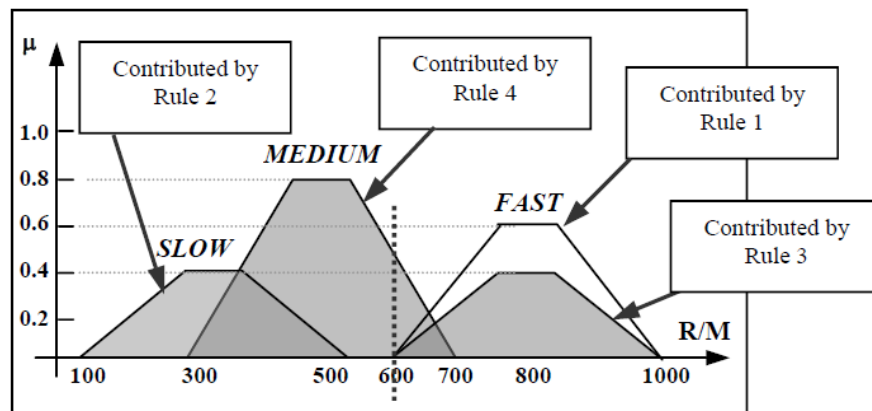
Figure 9: Determination of fuzzy output by the center of gravity method

2.2.2. Wavelet Transform

a. Signal

Any information which can be represented with respect to time can be denoted as a signal. The signal is usually represented in the time domain for most of the time. The signal in its time domain is not always helpful. There might be some information available in its frequency domain as well. To represent the signal in its frequency domain, we need to transform the signal from its time domain to frequency domain. Representation of signal in its frequency format explains what frequency components exist in that signal.

b. Frequency

The frequency is something to do with the change in the rate of something [39]. Frequency is categorized into two types, low frequency and high frequency. When the change rate is smooth, it is of low frequency; and when the change rate is high, it is of high frequency. As a general example, the daily publication has a higher frequency than monthly publication. To find the frequency of a signal mathematical transformation is applied to the signal. Figure 10 [39] shows the sine wave of 50Hz. Figure 11 [39] shows the mathematical transformation of the sine wave in Figure 10. In Figure 11 there is only one spike at 50 which means the signal in its time domain is stationary no other frequency exists in this signal.
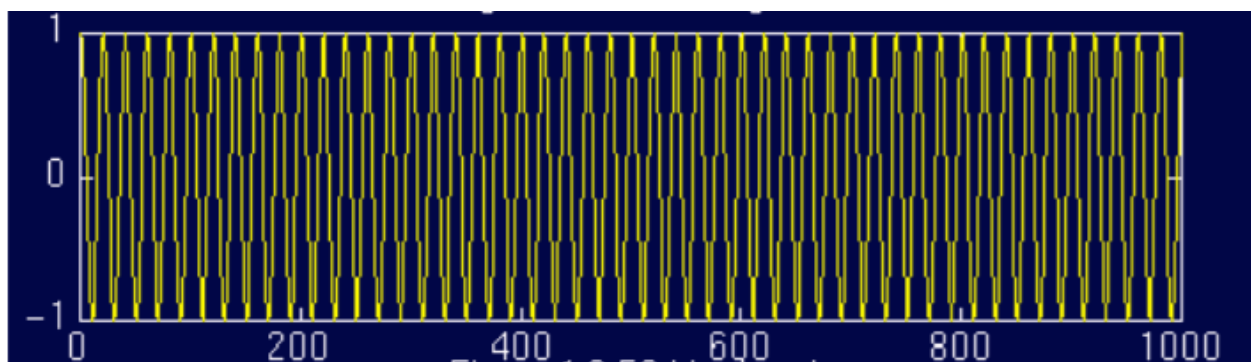
Figure 10: 50Hz signal

This example is discussed to understand the need for mathematical transformation of a signal [39]. Time domain ECG signal is analyzed by cardiologists to identify if any pathological

situation is existing or not but if the frequency component is also analyzed then the diagnosis can be made quickly.
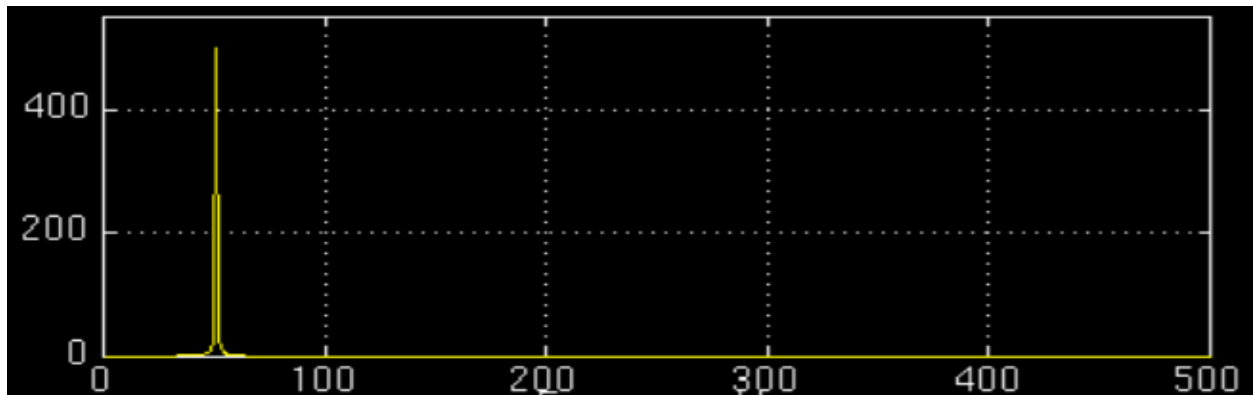
Figure 11 The FT of the 50 Hz signal given in Figure 10

c. Mathematical Transformation of the Signal

I. Fourier Transform

In the world of signal Fourier, the Fourier transform (FT) is the most utilized transformation technique. Fourier Transform (FT) processes the time domain signal to frequency domain signal which can be represented by placing frequency content on x-axis and amplitude on the y-axis. If the FT of a signal in the time domain is taken, the frequency-amplitude representation of that signal is obtained [39].

a) Stationary Signal

Signals whose frequency content do not change in time are called stationary signals [39]. That means the value of frequency content is the same throughout the existence of that signal. When the signal is stationary, generally there is no meaning of doing mathematical transformation as one already knows the frequency content of the signal. Figure 12 [39] is the time domain signal of type stationary whereas Figure 13 [39] is the frequency component of Figure 12. A conclusion can be drawn from the Figure 13 that only 10, 25, 50, and 100Hz frequency component exist for the entire duration.
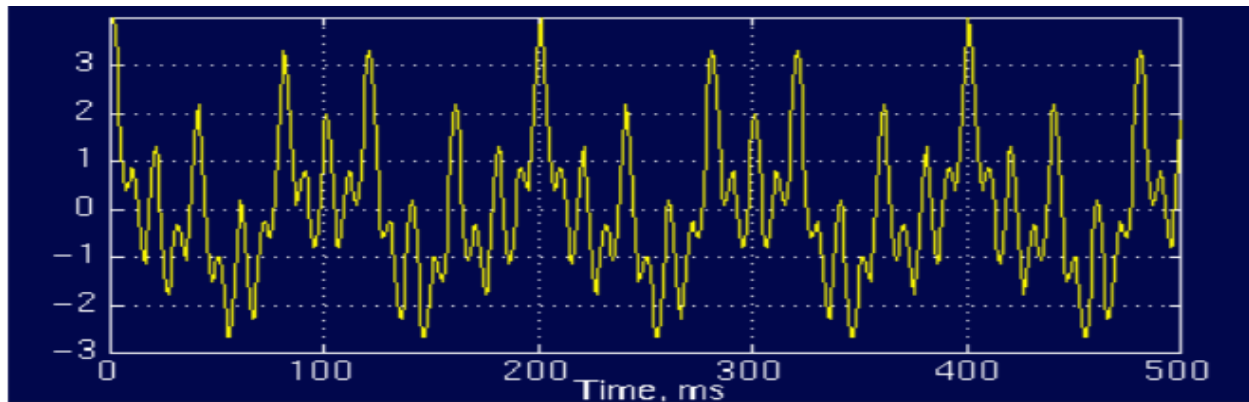
Figure 12: Stationary signal
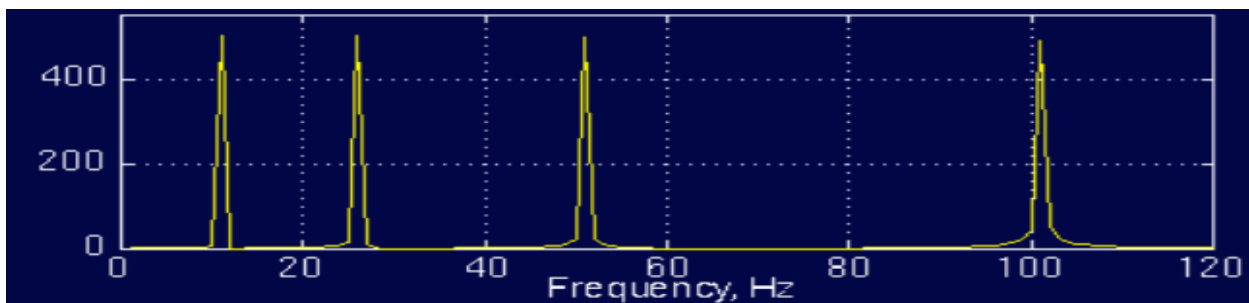
Figure 13: Frequency component of figure 12

b) Non-stationary Signal

When the signal has different frequency components irrespective of time, such signal is known as a non-stationary signal.
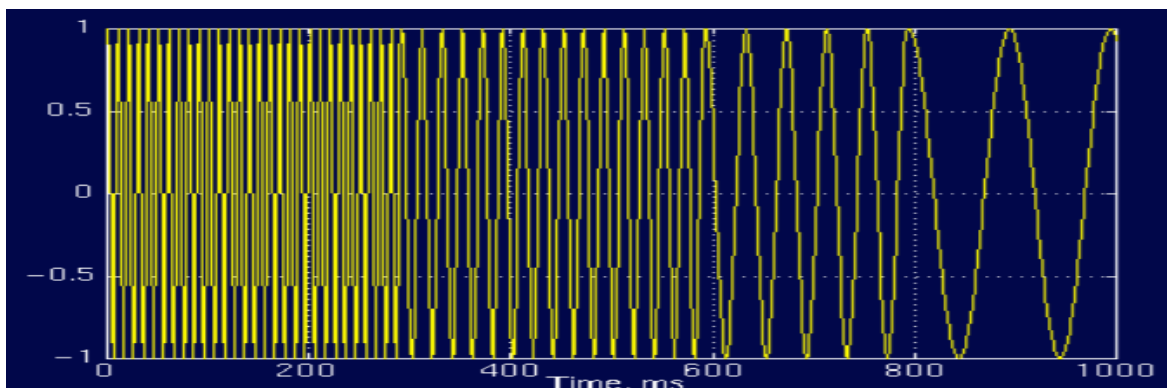
Figure 14: Non-stationary Signal

In Figure 14 [39] time interval from 0 to 300ms, 300 to 600ms, 600 to 800ms has 100Hz, 50Hz, 25Hz, and 10Hz frequency respectively. Figure 15 [39] shows the different frequencies that

exist in the signal. From the figure, there is no way to identify where in the time which frequency exists in the signal.
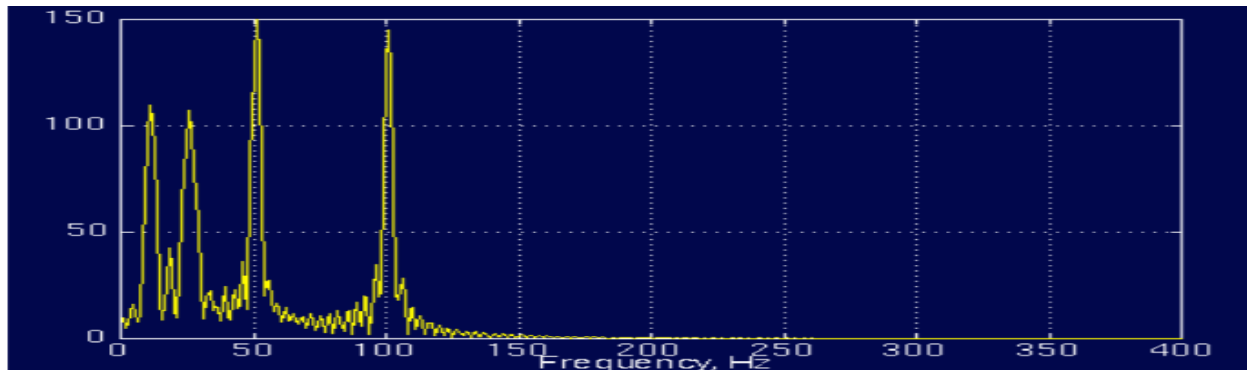
Figure 15: Frequency representation of figure 14

The drawback of Fourier Transform

Figure 13 and Figure 15 are Fourier transform of signals in Figure 12 and Figure 14. By comparing Figure 13 and Figure 15 we can see both signals have the same frequencies, i.e., at 10, 25, 50, and 100 Hz except some ripples in Figure 15. Also, there is no similarity in the time domain signal (Figure 12, Figure 14). Both time domain signals have the same frequency component, but in Figure 12 this frequency content exist all the time whereas for Figure 14 this frequency content exists for some fixed interval of time. As mentioned before FT is responsible for identifying the spectral component of any signal. However, FT is not responsible for keeping information of when in the time these frequency content exists. Hence, when the signal is of type non-stationary, FT is not a suitable transform. FT can be used for non-stationary signals, if we are only interested in what spectral components exist in the signal, but not interested where these occur [39]. However, if this information is needed, i.e., if we want to know what spectral component occurs at what time (interval), then the FT is not the right transform to use [39].

II. Short-Time Fourier Transformation

Short time fourier transform is a revised version of the fourier transform. The idea behind Short-Time Fourier Transformation (STFT) is that it treats the non-stationary signal as a stationary signal for some finite interval of time and performs the mathematical transformation on that

stationary signal. A window function is used to compute the transformation. The signal is divided into the small segments, and on each segment, the window function is applied.

$$STFT_X^W(t', f) = \int_t [x(t) \cdot W(t - t')] \cdot e^{-j2\pi ft} dt \qquad (5)$$

Here, $x(t)$ is an original signal, $W(t)$ is window function, f represents the frequency component of the signal, t' is window time interval, and t is finite time. For every t' and f a new STFT is computed.
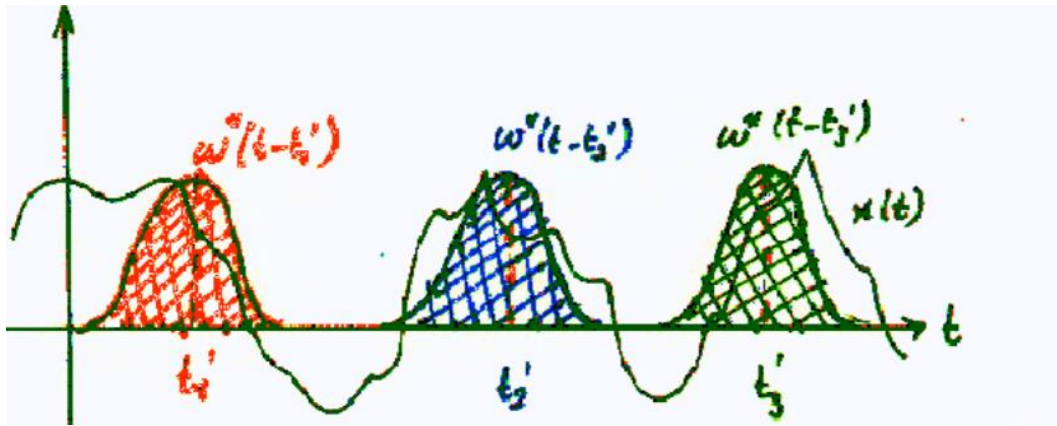
Figure 16: STFT using a window function

Figure 16 [39] will make it easy to understand how STFT works. The Gaussian-like functions are window function. The red, blue and green colored area shows the window function at time t1', t2', and t3' respectively and this represent the STFT at three different times [39]. Hence, STFT will give us both time and frequency component at the same time.

The drawback of Short-Time Fourier Transform

Which frequency component exists at what time can be seen using STFT. However, the problem with STFT is the fact whose sources go back to Heisenberg Uncertainty Principle [39]. The principle states that we can not identify both the frequency and time of a signal

24

instantaneously. "STFT provides the time intervals in which certain band of frequencies exists, which is a resolution problem. The window size performs the main role in having the time and frequency resolution problems. The narrow window gives good time resolution and poor frequency resolution where wide window gives good frequency resolution and poor time resolution." [39]

III. Wavelet Transform

The wavelet transform is used to identify the time-frequency representation of the signal. Wavelet is a small wave. Wavelets have an oscillatory property which works same as window function in STFT. Although the wavelet transform is the transformation process from the time domain to time-scale domain, these processes are known as signal decomposition because a given signal is decomposed into several other signals with different levels of resolution [40].The signal will be passed through several high pass and low pass filters to separate the high frequency and low frequency from the original time domain signal. This process will remove some portion of the signal whenever passed through filters. For instance, a signal with 500Hz frequency when passed through the filters will separate the signal into two signals one containing 0 to 250Hz frequency components of the signal which can be defined as low frequency while the other part is 250-500Hz frequency components which are of high-frequency components. By taking the either or both frequency component, the process will be continued. This process is known as decomposition of the signal. A wavelet transform can easily handle the resolution problem of FT. The WT gives better time resolution in high frequency while better frequency resolution is achieved in low-frequency components. The wavelet transform was used to adopt a wavelet prototype function, which utilized the basic function $\psi \in L_2(R)$ that is stretched and shifted to capture features that are local in time and local in frequency [35]. The transformation function is known as mother wavelet. There are different mother wavelets available in wavelet transform. Wavelet can be defined as pairs of high pass and low pass filters [35].

$$\varphi_{J,k}(t) = 2^{\frac{J}{2}}\varphi(2^J t - k); k \epsilon \mathbb{Z}. J \epsilon \mathbb{Z}^+ \tag{6}$$

25

$$\int_{-\infty}^{\infty} \varphi(t) = 1$$

And

$$\psi_{j,k}(t) = 2^{\frac{J}{2}}\varphi(2^j t - k); \; j = 1,2,3 \dots J$$

$$\int_{-\infty}^{\infty} \psi(t) = 1$$

The $\varphi_{J,k}$ is high pass filter whereas $\psi_{j,k}$ is low pass filter. Hence function $f(\cdot)$ can be defined as

$$f(t) = \sum_{k \in \mathbb{Z}} c_{J,k}\varphi_{J,k}(t) + \sum_{k \in \mathbb{Z}} d_{J,k}\psi_{J,k}(t) + \dots + \sum_{k \in \mathbb{Z}} d_{1,k}\psi_{1,k}(t) \qquad (7)$$

We can define the coefficient of both the wavelet as follow $c_{J,k} = \int_{-\infty}^{\infty} f(t)\varphi_{J,k}(t)dt$ and $d_{j,k} = \int_{-\infty}^{\infty} f(t)\psi_{J,k}(t)dt; j = 1,2,3, \dots J$ respectively.

Figure 18 [41] shows an example of the overall process of decomposition and reconstruction by wavelet transforms. These processes allow recovering the original time domain signal without losing any information, which is known as inverse DWT [40]. The reverse process of wavelet transform is known as inverse wavelet transform or signal reconstruction [42]. There are different types of wavelet transform such as continuous wavelet transform (CWT) and discrete wavelet transform (DWT).
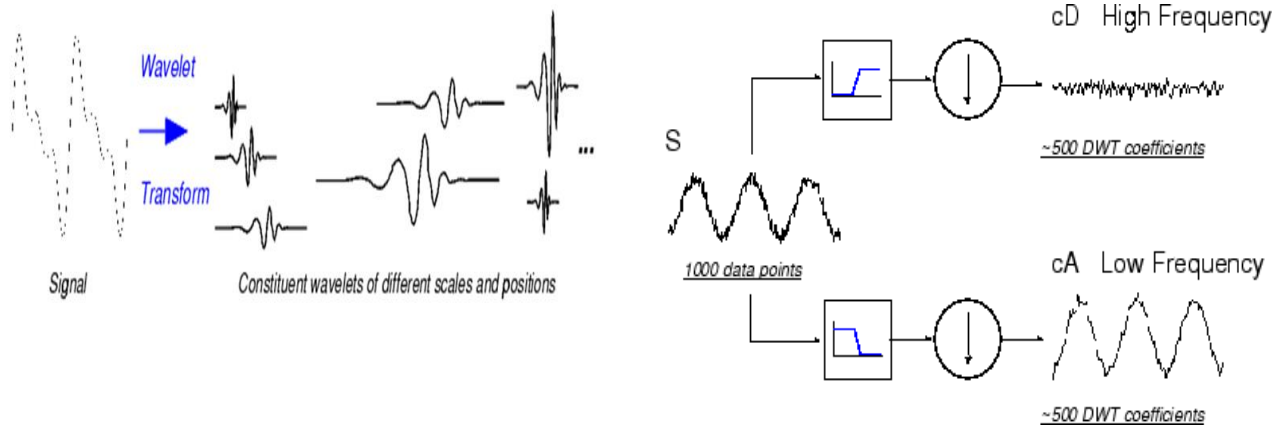
Figure 17: CWT and DWT of the time signal

Depending on the applications, regarding the continuous input signal, the time and scale parameters can be continuous, leading to the continuous wavelet transform (CWT). On the other hand, the discrete wavelet transform (DWT) can also be used for discrete time signals [42]. Selecting between CWT and DWT depends on the application.
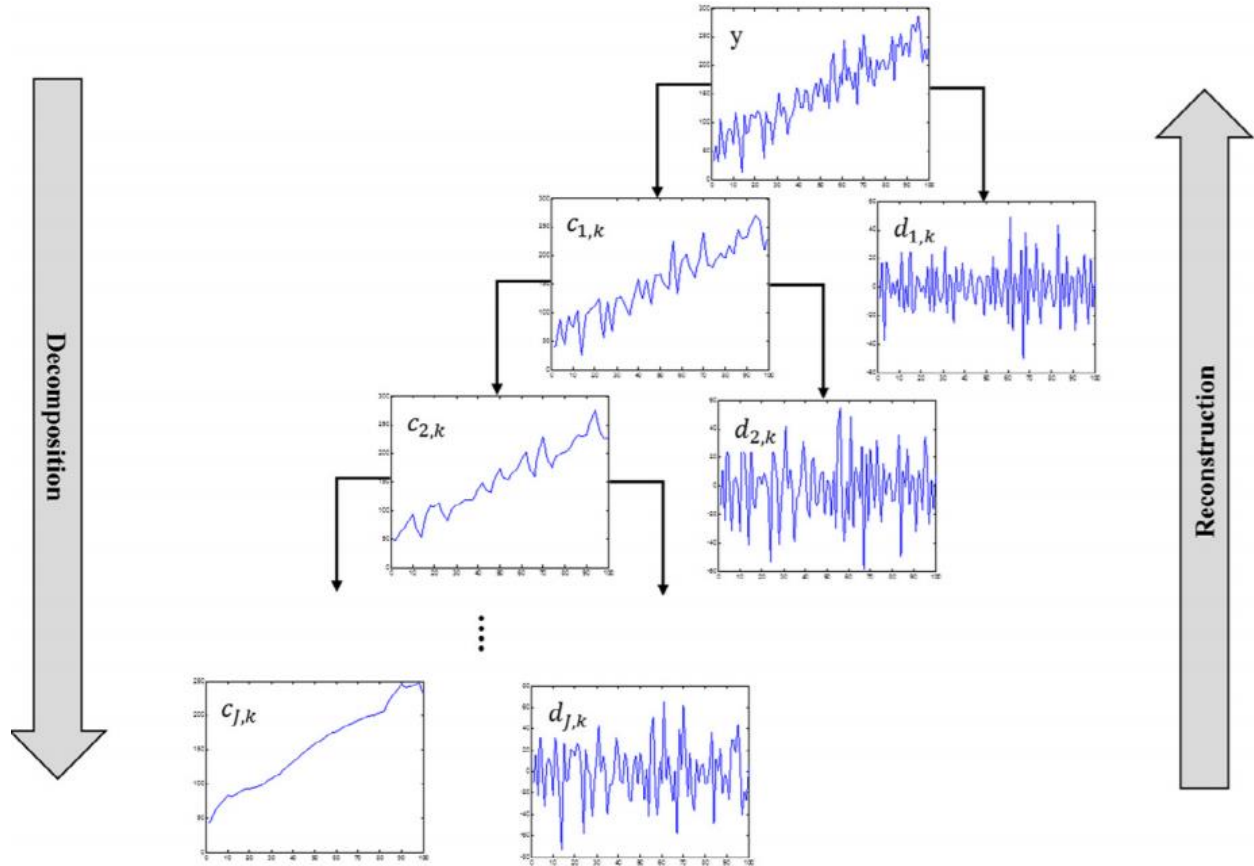
Figure 18: A process of decomposition and reconstruction by wavelet transforms

If one is interested in visualization, matching or feature detection, CWT might be more useful. If you are interested in denoising, compression, restoration, DWT is more appropriate [43]. The DWT decomposes the time series into different coefficients that represent the signal at different scales which helps to determine local abrupt and regular trend of time series in different scales [44]. There are several wavelets available such as Haar, Daubechies, Meyer, Symlet, etc. Haar wavelet transform has been applied in this research. It is a simple Daubechies wavelet, which is suitable to detect time-localized information and to increase the performance of the prediction technique [45] [46].

Sharp Variation Points by Wavelet Transform

Traffic flow time series show different fluctuation characteristics at different time scales [47] [48]. The scales in wavelet transform are divided based on some zero-crossing points from

28

wavelet transform, which is called sharp variation points [48]. The researchers proved that on a particular scale, the number of sharp variation points of the traffic flow shows the characteristic of self-similar [48] [47]. We can obtain that a wavelet function $\psi(x)$ equals to the first derivative or the second derivative of the smoothing function $\theta(x)$ [49]. When $\psi(x)$ is chosen as the second derivative of the smoothing function, the zero-crossing of the discrete dyadic wavelet transform related to the closed-contour signal indicates the position of the sharp variation point [49]. The detail coefficients represent positive and negative values. The multiplication between two consecutive coefficients becomes less than zero then those points will be revealed sharp variation points of the signal [4].

$$W_q(t, s) \cdot W_q(t + 1, s) < 0 \tag{8}$$

In equation (**8**), W_q is wavelet transformed signal at time t for scales.

2.2.3. ARIMA Model

The ARIMA model is discussed in [50] first time. It is the most used and classical model for time series analysis. The ARIMA model is based on the fundamental principle that the future values of a time series can be generated from a linear function of the past observations and white noise terms [30].

a. Model Structure

ARIMA stands for autoregressive integrated moving average. ARIMA is a combination of 3 parts:

1. AR (Auto-Regressive)
2. I (Integrated)
3. MA (Moving Average)

An ARIMA (p, d, q) is represented by three parameters: p, d, and q, where p is the degree of autoregressive, d is the degree of integration, and q is the degree of moving average. The Integrated (I) part in ARIMA subtracts time series with its lagged series to extract trends from the data. Autoregressive (AR) extracts the influence of the previous periods' values on the current period. Moving Average (MA) extracts the influence of the previous period's error terms on the current period's error.

$$x_t = \emptyset x_{t-1} + \varepsilon_t \Longrightarrow AR(1) \tag{9}$$

Equation (**9**) for AR with 1 lag is defined in [35]. Lag is a back shift of actual time series which can be any number of units.

$$x_t = \emptyset_1 x_{t-1} + \emptyset_2 x_{t-2} + \cdots + \emptyset_p x_{t-p} + \varepsilon_t \tag{10}$$

Here, ε_t is Gaussian noise and p is order of series.

The formula to calculate moving average is

$$x_t = \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \cdots + \theta_q \varepsilon_{t-q} \tag{11}$$

Moving average part depends on the noise part of current values as well as past values; it is working as a low pass filter. The combination of autoregressive (AR), moving average (MA) and integrated(I) is ARIMA model, which can be defined as [35]

$$\emptyset_p(B)(1 - B)^d (x_t - \mu) = \theta_q(B)\varepsilon_t \tag{12}$$

where B is the shift operator and μ is mean.

b. Model Identification

To determine the order of AR and MA the autocorrelation function (ACF) and partial autocorrelation (PACF) are used.

ACF plots display the correlation between a series and its lags. In addition to suggesting the order of differencing, ACF plots can help in determining the order of the q in ARIMA(p, d, q) model [51]. "The autocorrelation of a time series Y at lag 1 is the coefficient of correlation between Y_t and Y_{t-1}, which is presumably also the correlation between Y_{t-1} and Y_{t-2}. But if Y_t is correlated with Y_{t-1}, and Y_{t-1} is equally correlated with Y_{t-2}, then we should also expect to find correlation between Y_t and Y_{t-2}. In fact, the amount of correlation we should expect at lag 2 is precisely the square of the lag-1 correlation. Thus, the correlation at lag 1 'propagates' to lag 2 and presumably to higher-order lags. The partial autocorrelation at lag 2 is therefore the difference between the actual correlation at lag 2 and the expected correlation due to the propagation of correlation at lag 1" [52]. Partial autocorrelation plots (PACF), as the name suggests, display the correlation between a variable and its lags that is not explained by correlations at all lower-order-lags. PACF plots are useful when determining the order of the p in ARIMA(p, d, q) model [53].
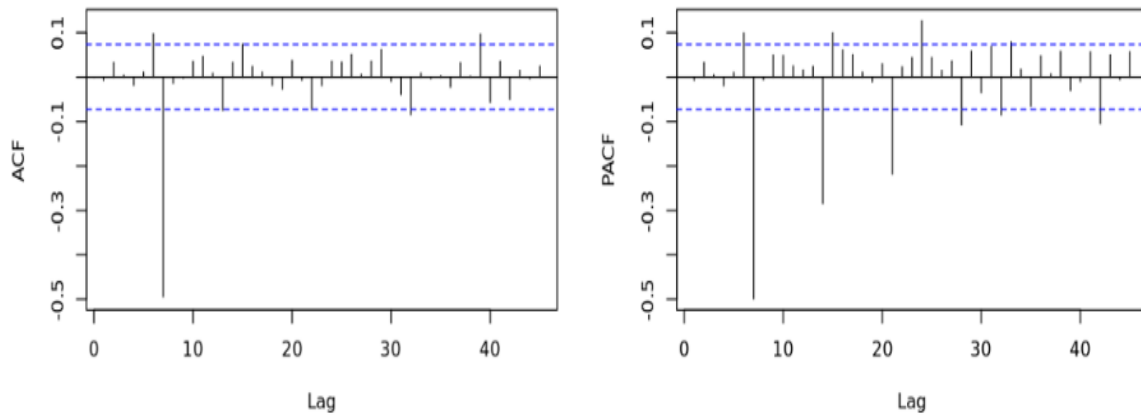
Figure 19: ACF and PACF plot

In Figure 19, an ACF and PACF plot for bike sharing demand data has been shown from [53]. For ACF/PACF there is some significant correlation at lag 7. Hence, the model might be of AR=7 or MA =7. After fitting the model for MA (7), we can tell that in Figure 20 there is no significant correlation available in the series.
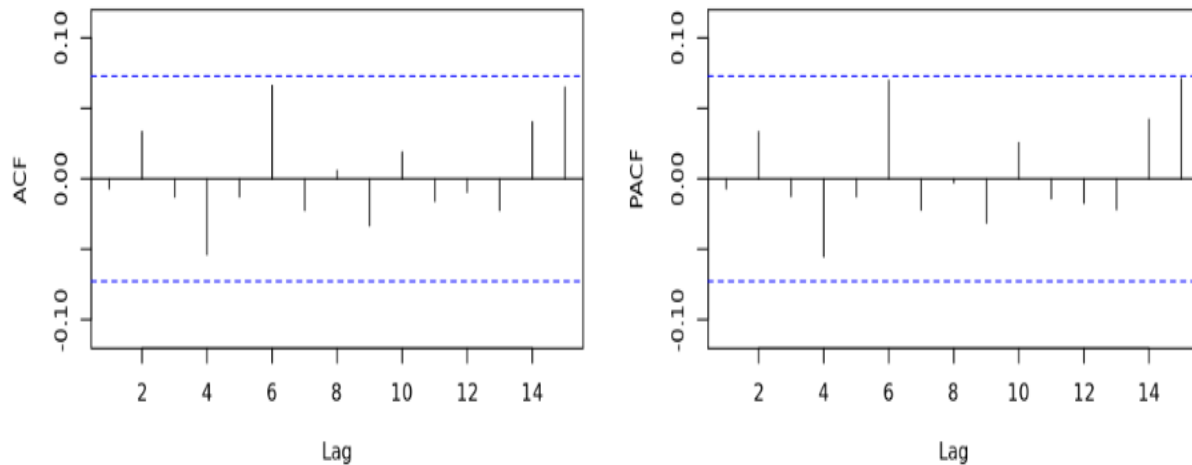
Figure 20: ACF and PACF plot after MA (7)

Table 3 represents the idea of selecting the best-fitted model for the forecast.

Table 3 Properties of ACF, PACF of AR, MA and ARMA series

Model	Autocorrelation	Partial Autocorrelation
AR(p)	Tails off gradually	Cuts off after p lags
MA(q)	Cuts off after q lags	Tails off gradually
ARMA(p,q)	Tails off gradually	Tails off gradually

In addition to ACF and PACF, more techniques are available like AIC (Akaike information criterion) and BIC (Bayesian information criterion) to measure the degree for ARIMA parameters and find the best optimum model.

Akaike Information Criterion: "A model fit statistic considers goodness-of-fit and parsimony. Select models that minimize AIC. When comparing multiple model fits, additional model parameters often yield larger, optimized loglikelihood values. Unlike the optimized loglikelihood value, AIC penalizes for more complex models, i.e., models with additional parameters. The

formula for AIC, which provides insight into its relationship to the optimized loglikelihood and its penalty for complexity, is" [54]:

$$AIC = -2(logL) + 2(number\ of\ parameters) \qquad (13)$$

Bayesian Information Criterion: "Like AIC, BIC uses the optimal loglikelihood function value and penalizes for more complex models, i.e., models with additional parameters. The penalty of BIC is a function of the sample size, and so is typically more severe than that of AIC. The formula for BIC is" [54]:

$$BIC = -2(logL) + (number\ of\ parameter) \times \log(number\ of\ parameter) \quad (14)$$

3. Proposed Method

This chapter discusses the characteristics of data for time series analysis, the architecture of the algorithm and steps involved by flowcharts and block diagrams of the proposed hybrid algorithm.

3.1. Motivations

Before delving into the study of traffic prediction, the findings of the related work discussed in previous chapters gave solid reasons which need to be considered while predicting the time series. These issues mentioned in 2.1.1 must be considered while forecasting the traffic flow. A significant part of the current work focuses on the short-term forecasting which helps to improve the decision-making process of traffic lights or online map system to guide the user to take a less time-consuming road [21]. On the other hand, the long-term traffic prediction is helpful to those who are planning a journey and can anticipate their travel time in advance to reach their destination in time [21].

In the proposed method some of the pitfalls of previous work have been addressed in the following two aspects.

- **Model:**

   Prerequisite of a model which is trustful under all circumstances. The model should be able to work for adjustable time spans. It should be able to handle missing information, capture the changing trend of information and work with stochastic data.

- **Accuracy:**

   The prediction is considered accurate if accuracy is stable for the different time span. Also, the accuracy should be maintained for the same condition.

3.2. A new hybrid approach using the ARIMA model with Fuzzy wavelet transform

In time series forecasting, identifying the abnormal behavior of traffic flow from the vast dataset and finding the reason behind irregular traffic pattern can help to identify the changing trend. The reason behind such an abnormal traffic count can be any special events such as accidents, severe weather conditions, roadwork, or vehicle breakdown. We can find such a trend when time series data starts to behave irregular from usual daily traffic patterns.

The characteristic of the traffic flow forecast is hard to capture. From Figure 21, for 25th July and 26th July the flow of traffic is almost similar. For 27th July significant fluctuation from other two-time series can be noticed. There were multiple events happened which have affected the normal flow of traffic. Initial time of 1 p.m. to 1:15 p.m., there was congestion which decreased the traffic volume steeply. After that between 1:45 p.m. to 2:00 p.m., there is an increase in traffic because an accident happened at 1:09 p.m. The next event happened at 2:20 p.m. when a car accident happened which has increased the traffic flow from 3:15 p.m. to 3:45 p.m. until the clean up took place. The proposed forecasting model captures such influencing events as features impacting the forecasting accuracy. The source of comparison is taken from [55] and the original work used the nearest neighbor algorithm to detect such an event and to identify the affected regions and calculate the time of delay. In the proposed method, fuzzy logic detects such an event and find all the traffic data that has a similar event with respect to forecasting duration.
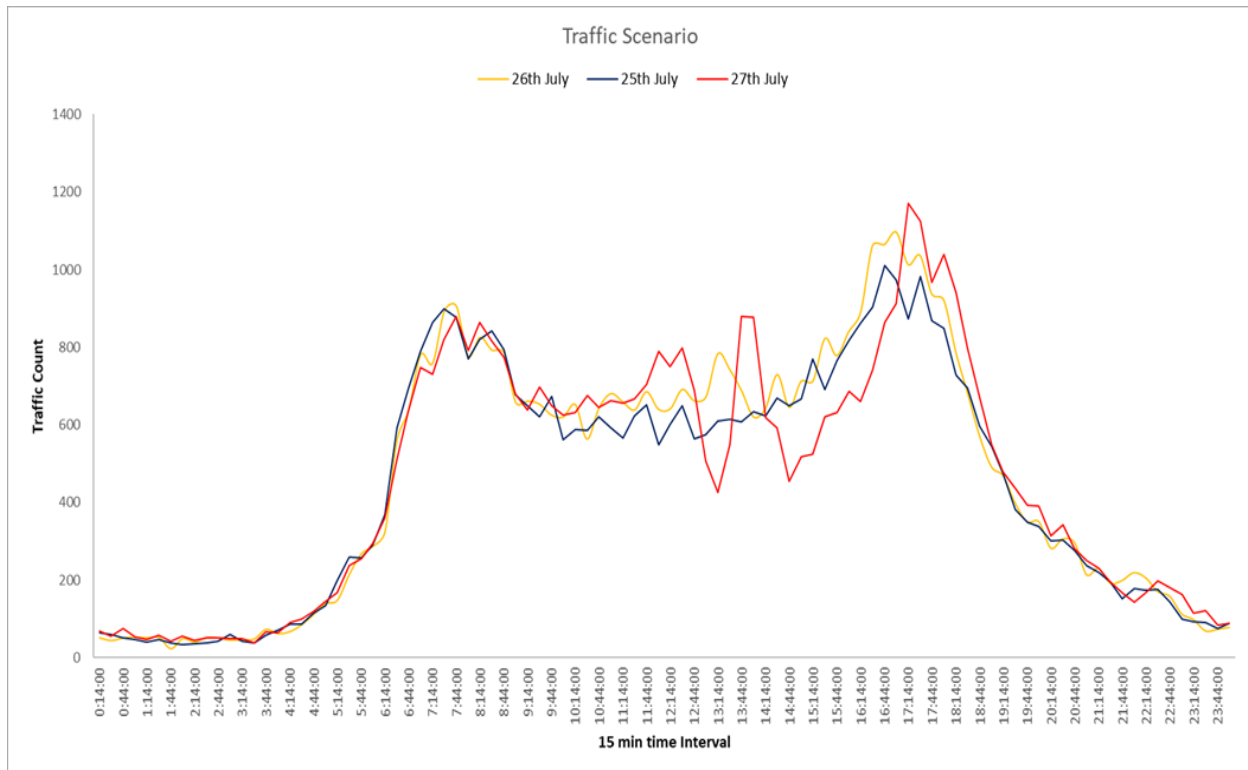
Figure 21: Traffic count comparison of regular days against special event day

A fuzzy classifier is used to compare real data with stored data to the degree of membership. From the definition of the degree of membership denoted by μ, first the level of preciseness is defined. Three categories of preciseness as described in Table 4 which depends on the value obtained from the degree of membership. As these values range from 0 to 1, the degree of membership can be defined as μ [0,1]. The degree of membership of any historical traffic dataset will be determined based on similarities between real-time dataset and historical dataset [4].

Table 4 Fuzzy membership Categories

Highly Similar Dataset (HSD)	Having at least 8 similar data
Medium Similar Dataset (MSD)	Having at least 4 to 7 similar data
Less Similar Dataset (LSD)	Having no or at least 3 similar data

Based on the number of factors N_C (number of data components) this method classifies dataset as HSD for having 8 similar data out of 11, as MSD for having at least 4 similar data , or as LSD for having at least 3 similar data out of 11 [3]. Here N_C can vary based on used dataset and similarity will be adjusted accordingly. Based on the calculated membership degree, days with the highest membership degree will be used only for the further process of the forecast. If the selected featured dataset is not matching with HSD, the system looks for a matched MSD with minimal discrete distance. And lastly, the system looks for an LSD dataset if there is no MSD with minimal discrete distance exist.

Evidently, Khandelwal et al. [30] suggest that neither ARIMA nor ANN is universally suitable for all types of time series. Almost all real-world time series contains both linear and nonlinear correlation structures among the observations [30].
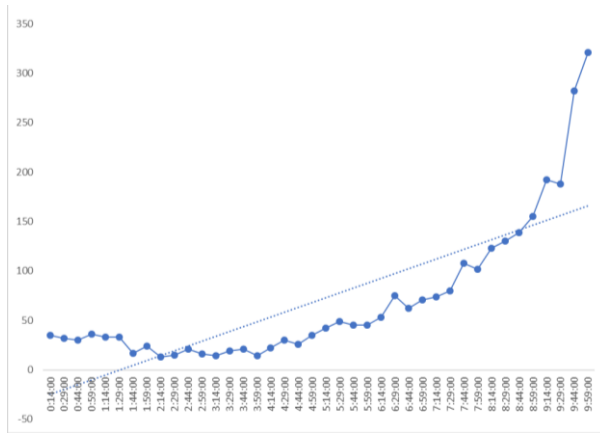
Figure 22a: From 6:00 am to 10 am
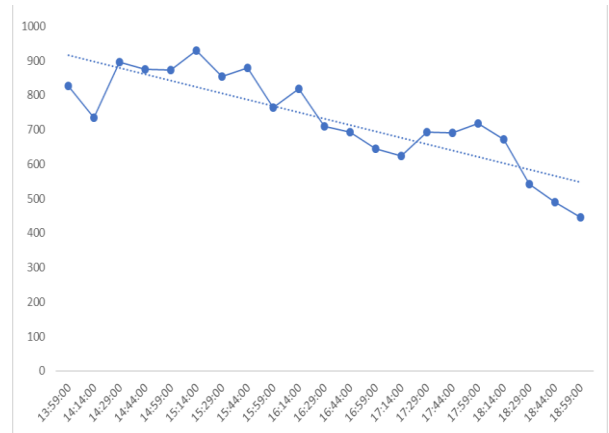
Figure 22b: From 2:00 pm to 6 pm

Figure 22: Trend in Time Series

Figure 22a and Figure 22b show a comparable situation. In Figure 22a the trend is upward, and for Figure 22b trend is going downward. It can be easily compared with a real-life situation. In specific time-period when people travel more often, and the traffic condition becomes more complicated. Such type of non-linearity is not handled by ARIMA, which affects the forecast results. In [35] a hybrid approach uses DWT to convert the time series in the linear

and non-linear component. The proposed method will use the advantage of converting time-series data into linear time series data. Another advantage of using DWT as discussed in [56] is to apply the transformation to approximation coefficients in each layer. The filter is used to recognize local dependencies in the data, which are subsequently combined to represent more global features until the output of interest is computed in the final layer. These coefficients will be further used in final forecasting model ARIMA to predict the result.

3.2.1. Design of Algorithm
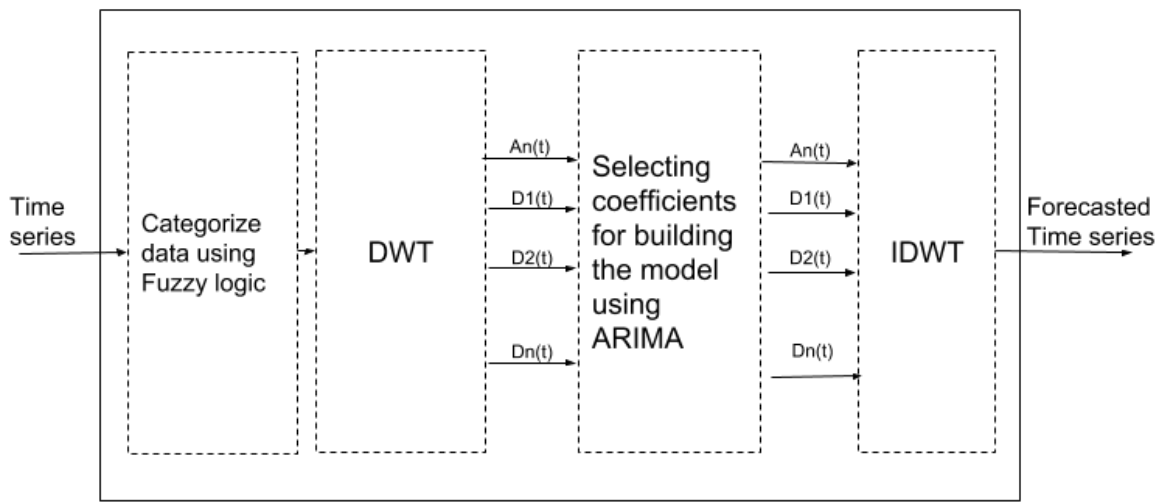
Figure 23: Proposed Hybrid Algorithm

Given a time series $X = \{x_t, t = 1, 2, \ldots, n\}$, where x_t is a value at discrete time t, we need to predict the value at time $t + i$. As shown in Figure 23 the proposed forecasting model is defined in four sections. First, it will do the categorization of the dataset with affected external factors and create the featured dataset. Next, DWT is applied to the dataset which will decompose the non-linear time series data into linear time series and generate the approximation (general trend component) and detail (high-frequency component) part of the time series at different time scale. In addition, DWT will eliminate random component or denoise the time series data which will improve the quality of data for producing a more accurate result. Further ARIMA model will use the output from DWT to estimate the parameter (p, d, q) for

ARIMA model. The one with an optimal (p, d, q) will be used. After getting an optimal linear time series from ARIMA inverse DWT will be applied to construct the predicted traffic time series.

Algorithm 1: ARIMA MODEL WITH FUZZY WAVELET TRANSFORM

INPUT: $X = \{x_t, t = 1,2, \dots, n\}$, where x_t is a value at discrete time t

OUTPUT: Predicted traffic from $t + i$ as $(x_t + i)$ where i is the forecasting period

---

1. train_dataset ← 70% of the dataset

2. real_time_dataset ← 30% of dataset

3. CategorizModels ← FuzzySystem(train_dataset)

4. foreach(x in CategorizModels)

    4.1. trained_coeffs ← dwt(x, filterlevel, filtertype)

    4.2. SVP(trained_coeffs)

5. trained_arima_model ← arima(trained_coeffs)

6. foreach(X_t in real_time_data)

    6.1. DOM ← FuzzySystem(X_t)

    6.2. Model ← CategorizModels(DOM)

    6.3. coeffs ← dwt(Model, filterlevel, filtertype)

        6.3.1. foreach($\hat{X}_t$ in Model)

            6.3.1.1.   distance ← $\sqrt{\sum_1^T(X_t - \hat{X}_t)^2/T}$

        6.3.2. data ← $Day of Minimum$(distance)

    6.4. i ← Find distance between two nearest consequent SVPs from data

    6.5. arima.forecast(coeffs, trained_arima_model, i) for next $(x_t + i)$

7. Go for next $(x_t + i) + 1$

8. End

In the 1st and 2nd steps, a dataset will be divided into training and testing datasets. Training dataset will be categorized into HSD, MSD, and LSD using the fuzzy system. In the HSD the data with highest matching features with the prediction day will be placed. MSD will have less similar data, and LSD will have almost non-similar data. First HSD will be used for the further process if HSD is null. Otherwise, if no data with a similar feature of prediction day found then, we will use MSD which has few numbers of matching data. If MSD is blank, then we will use LSD. The selected dataset will be returned from the fuzzy system as CategorizModels. In the next step, it will be passed to the discrete wavelet transform to denoise and linearize the data. The DWT will return lists of coefficients of different scales which help to track even the minor trend in time series. The denoised data will be passed to ARIMA to get the optimized model for the forecast. In step 6, the degree of membership (DOM) is calculated for the prediction day. DOM will be compared against CategorizModels's degree of membership and matching dataset will be selected. In this step, we calculate the sharp variation points of each day in training data and store it in the table. Hence, the algorithm starts processing data from the beginning of the day. The data of prediction day along with time frame of prediction defined as variable i in the algorithm is passed to an optimized model which will calculate next prediction value(s) up to i. The variable i represent the duration in which the trend is constant. Sharp variation points usually indicate either the beginning of the trend or end of the trend. By finding the distance between two consecutive SVP points, we can assume that for that period the trend is constant hence the same ARIMA model can be used to forecast up to the variable i.

Algorithm 2: Fuzzy System

INPUT: train_dataset

OUTPUT: CategorizModels

1. Determine the required features as variables of the fuzzy system to partition datasets in

     HSD, MSD, and LSD

2. Define fuzzy rules to map the variables to one of the categories (HSD, MSD, LSD)

3. Calculate the degree of membership for each rule

4. Divide the dataset into (HSD, MSD, LSD)as per the specified boundary of membership

     for each partition

5. return HSD, MSD, LSD

6. End

4. Implementation and Experiments

4.1. Experiment Data

There are three datasets used in the experiment of the proposed method. Among them, one is constructed to include all the features listed in Table 5 and the other two consist of only weather data as a feature for analysis purpose. The first data set is taken from Highways England for site M1/4771B (MIDAS site at M1/4771B priority 1 on link 116019002; GPS Ref: 432323;405646; Southbound) [57]. For the same geographical position, the event data is taken from Traffar: UK Current traffic information where all the event related to M1 southbound junction 37 is available. The weather data for the same location is taken from timeanddate.com for nearby city Barnsley, England, United Kingdom. All the data of weather and events are considered as feature data. Table 5 is a list of features considered in the experiment. The time interval is 15 minutes.

Table 5: Feature List used in Traffic analysis

Feature Type	Feature Name	Feature Number
Weather	Snow	1
	Wind	2
	Humidity	3
	Fog	4
	Temperature	5
	Precipitation	6
	Duration	7
Holiday	0 or 1 (0 is holiday, 1 no holiday)	8
Day after holiday	0 or 1 (0 is the day after the holiday, 1 regular)	9

Event	Accident or Roadwork or vehicle breakdowns	10
Geographic Location	Highway or Intercity	11

The second dataset provides a minute by minute traffic flow for Junctions 25 to 30 of the M62. Traffic data is recorded via inductive loops by the MIDAS Subsystem (Motorway Incident Detection and Automatic Signalling) for those sites that have been enabled as traffic counting sites [58]. The weather data for nearby city Brighouse HD6 4HG, UK is taken. The third dataset is traffic data provided by Washington State Department of Transportation (WSDOT), USA. WSDOT determines traffic count data from most of the roads in the city road network [4]. This research uses traffic count for the road at Rhododendron, Coupeville on milepost 20:02 point, Eastbound [4].

4.2. Performance Evaluation measurements

We utilize three performance measures to identify the appropriateness of the proposed method which are the mean relative error (MRE), root-mean-square error (RMSE) and mean absolute percentage error (MAPE) as described in [6]

$$MSE = \frac{1}{n}\sum_{i=1}^{n}|f_i - \widehat{f_i}| \tag{15}$$

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\frac{|f_i - \widehat{f_i}|}{f_i} \times 100 \tag{16}$$

$$RMSE = \left[\frac{1}{n}\sum_{i=1}^{n}(|f_i - \widehat{f_i}|)^2\right]^{\frac{1}{2}} \tag{17}$$

Here, f_i stands for observed traffic count and $\widehat{f_i}$ stands for predicted traffic count.

4.3. Using Proposed Hybrid Model for Traffic Flow Forecast

The dataset of the 15-minute time interval is for almost one year. In Figure 24 the original time series has fluctuation which is an indication of noise. Therefore, after using the discrete wavelet transform, we have eliminated such noise as shown in Figure 25. Also, an analysis can be made from the Figure 25 that a regular recurrence at a particular time is present in the time series. The traffic flow starts to increase from 5 a.m. to 6 a.m. and keep increasing until 10 a.m. in the morning. After 10 a.m. to 4 p.m. almost decreases and then again between 5 p.m. to 6 p.m. increases in traffic can be seen in Figure 25. By identifying such trends, we can quickly build a model that follows them. The trend detection proposed in [50] used zero crossing detection to identify the trend in the time series. In Figure 26, DWT shows the changing trend in the topmost plot. After applying DWT, we can even trace the minor details available in time series. Also notice here that the 2nd and 3rd and 4th spikes are higher than other spikes. It is a holiday on 25th and
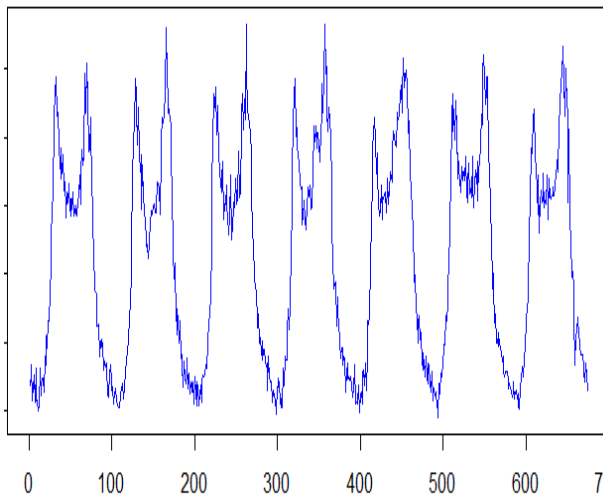
Figure 24: Original Time Series from 24th December to 30th December
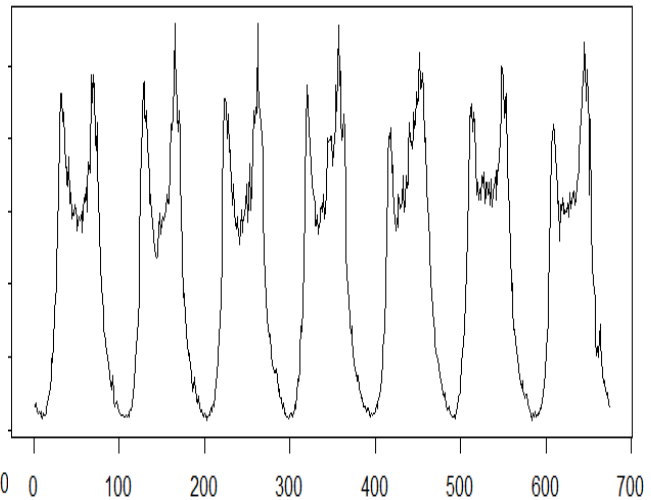
Figure 25: Denoised Time Series from 24th December to 30th December

26th December. As we all know in the UK, these days are the seasonal holidays of Christmas and Boxing day. Probably the busiest week as well. The analysis can be made here that the time series has a similar trend in the same time span. The proposed method used zero crossing detection in

discrete wavelet transform to spot such trends and for long-term traffic prediction as described in the previous section.
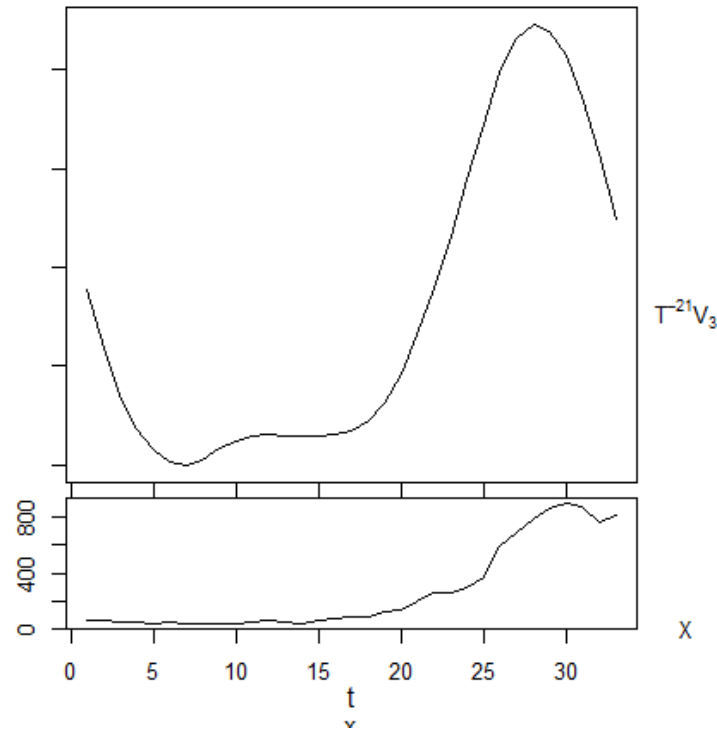
Figure 26: DWT of known data for 26th December

As per step 6.1 and 6.2 in the algorithm, all the datasets for similar features will be derived using Fuzzy system rules.

Table 6: Sample list of Zero crossing points for matching days for 26th December

01/01/2017	33	52	71	86
14/04/2017	33	51	70	84
17/04/2017	30	57	74	85
01/05/2017	35	52	78	86
29/05/2017	31	45	69	86
28/08/2017	33	53	64	88
25/12/2017	31	52	67	86

Then using the distance calculation formula $\sqrt{\sum_1^T (X_t - \hat{X}_t)^2 / T}$, where X_t is prediction day and $\hat{X}_t$ is matching day, a distance of each day with the prediction day will be calculated and stored. Then the data with minimum distance will be retrived which is a list of trained coefficients of matching day. In step 6.3 find the distance between the nearest two zero crossing points of matching day and pass it along with trained coefficients and the calculated optimized model to forecast upto the distance between two crosing points. For an example, Suppose prediction day is 26th december as shown in Figure 26 start from the initial time of 12 a.m. for the forecast. Based on the distance calculation formula the closest trend that matched for similar duration of day 26th December 2017 will be calculated, which is 1st January 2017. Next two sharp variation point for the matched day will be retrieved. From Table 6, the first two sharp variation points are 33 and 52 which is from 8 a.m. to 1 p.m. So, the first prediction for 26th December can be done using 1st January's data from 8 a.m.(t=33) to 1 p.m. (t=52) which is plotted in Figure 27. Then repeat the same procedure for the second prediction. The second closest match found for the same day is 14th April 2017. The next two sharp variation point from 14th April is 2 p.m. (t=51) to 5 p.m.(t=70) which is shown in Figure 28. The third closest matching trend found for the duration of 5 p.m.(t=69) to 11 p.m.(t=86) is 29th April 2017 and the prediction for that time is depicted in Figure 29. Repeat the above process to get prediction results till the end of the day.
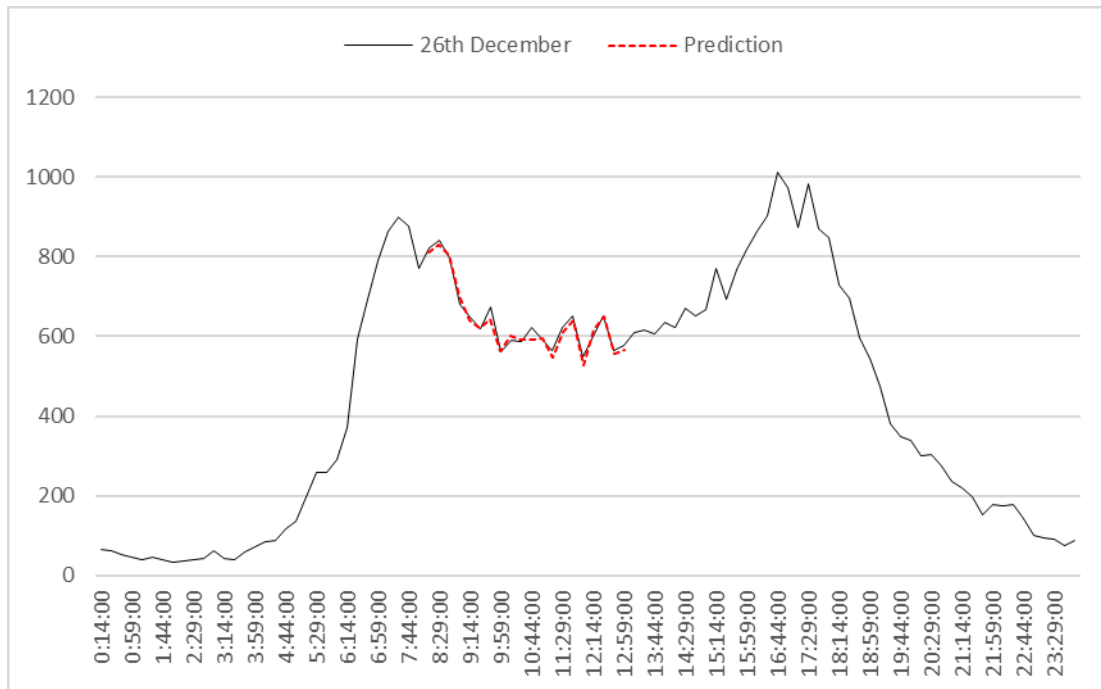
Figure 27: The first prediction of 26th December
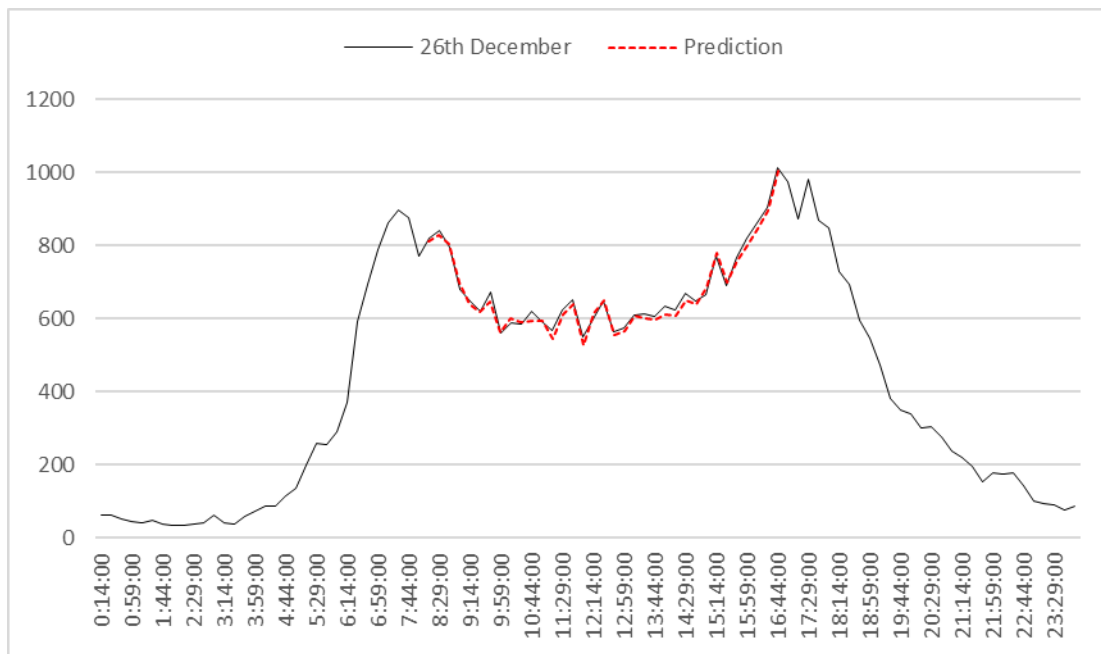
Figure 28: The second prediction of 26th December
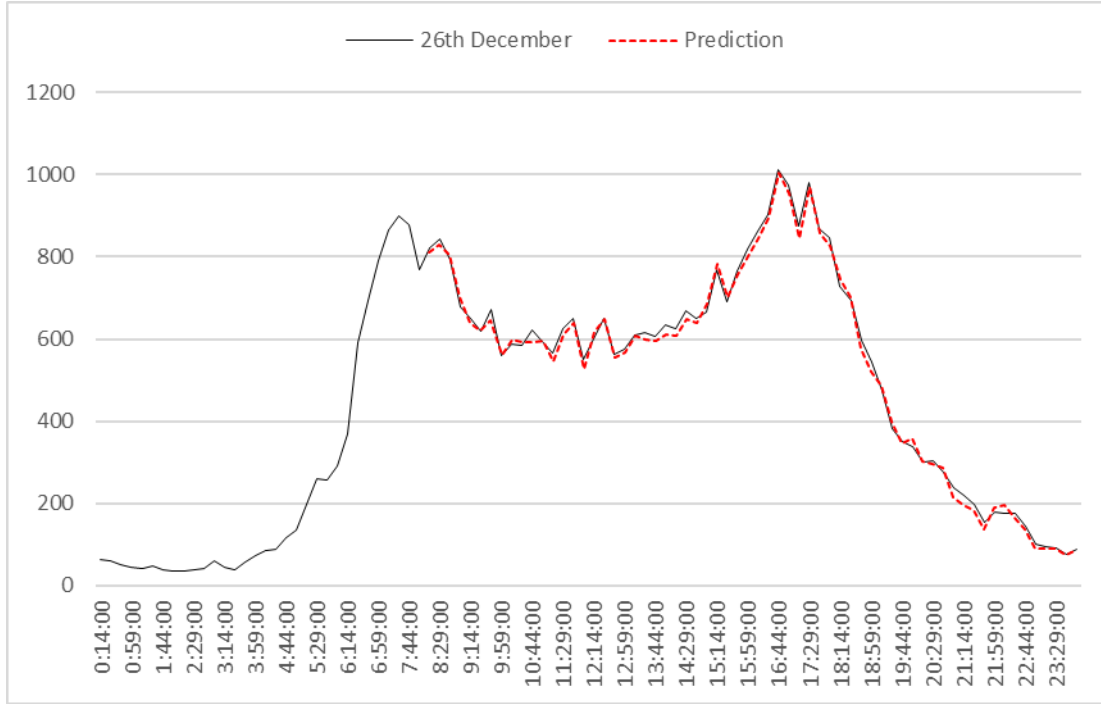
Figure 29: The third prediction of 26th December

4.3.1. Feature Analysis

From the discussion of the previous section, we can see that traffic flow is affected by some special events, weather, seasonal holidays. We will identify the selection of features in further discussion. Wang et al. [32] and Hossain et al. [4] discussed selecting certain features. The feature listed in Table 5 are selected based on the observations, and by analyzing experiments. Some of the observations are discussed here.
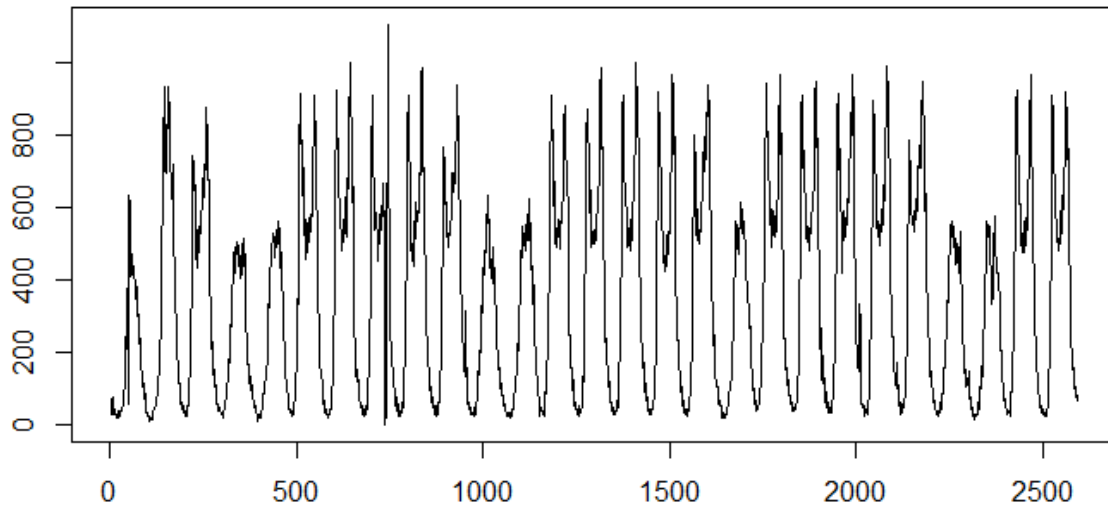
Figure 30: Time series for January 2017

A fascinating perception made here is that the preferred location is busy during weekdays and less busy during weekends. Though for the 1st December 2017 Sunday, the traffic seems high than other Sundays. The reason behind this rush can be the new year' day which inspires to select the seasonal holiday as a feature. Traffic rush difference between weekdays and weekend is also an important feature that affects the traffic count. In Figure 21 we compare the regular days against some days where special events like roadwork or accident happened which encourages to include those events in our feature list for traffic count dataset. Figure 31 shows the effect of fog on traffic count. Weather data suggest that on 6th and 8th February there was fog in morning pick hours which had increased the traffic count than usual day. As discussed in [32] and as per experiment suggest that holiday other than weekend may lead to overfitting. Hence, while training the model using ARIMA, holiday is not considered as a feature. Also, the rain feature is always combined with the duration of rain because the analysis suggests that traffic data is not affected while the rain was in the forecast for short duration.
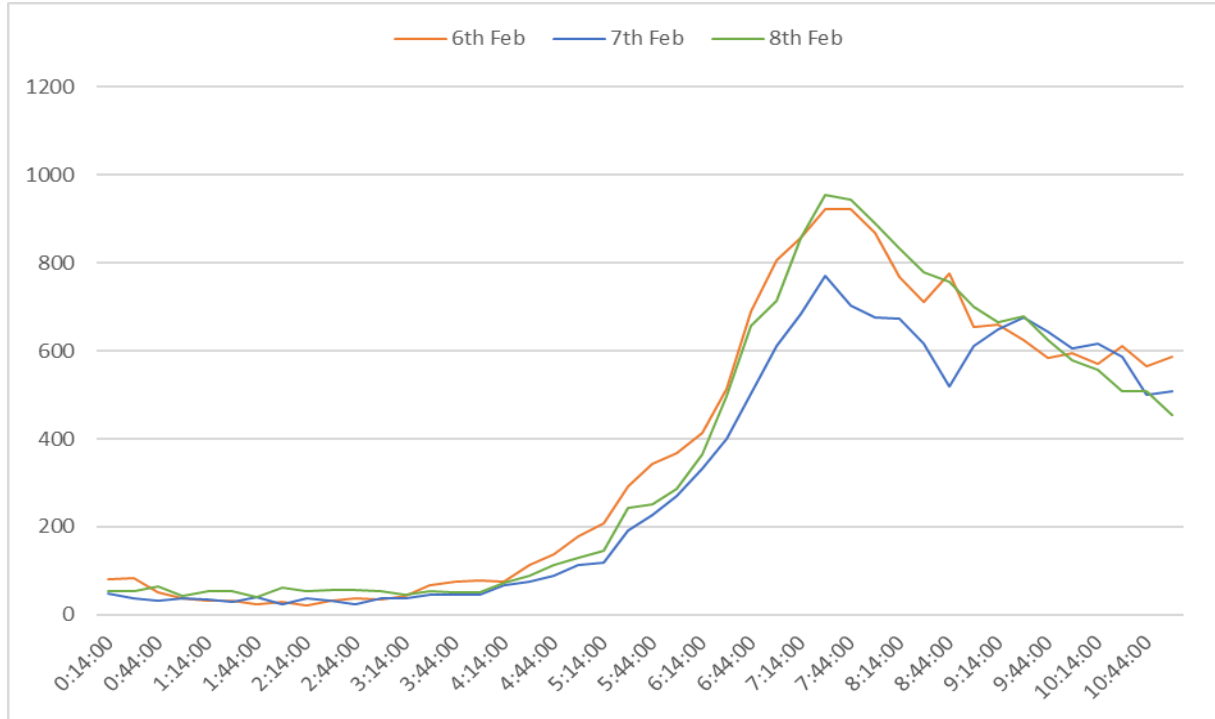
Figure 31: Fog as a feature

4.4. Experiments and Results

Above feature analysis leads to two types of traffic prediction cases. One is a special day while the other is an average day with no special events.

Case 1: The normal day for traffic prediction. The prediction day has no special event or weather condition or holiday that can affect the traffic count. The experiment will be performed on all the three datasets along with a short-term and long-term prediction. For the long-term prediction, we are going to use sharp variation points to decide the maximum period of forecast. On the other hand, the short-term prediction will predict only the next value of time series. There will be no feature analysis for this scenario. We will use 2 months data as historical data for training ARIMA model to find the optimal parameters and the day with no special event as a test day.

Table 7 shows the proposed algorithm is giving better accuracy compared to other traditional models. As the interval of time series is increasing, the accuracy is decreasing, but the proposed approach outperformed other models for long-term as well as short-term predictions. We can say that our model is more reliable with small interval time series.

Table 7: Forecast Accuracy Results for 15 min dataset

	Short-Term Traffic Prediction				Long-Term Traffic Prediction			
	10 Min	20 Min	30 Min	60 Min	2 hours	4 hours	8 hours	16 hours
CATA	95.0	94.3	94.0	87.0				
CATB	87.3	87.4	87.3	88.6				
Slip-road	92.2	91.5	92.1	84.7				
Velorra					92.18	92.07		
RBFNN					94.79	89.96		
Hossain J					95.13	93.86	94.96	80.69
Hybrid ARIMA	**99.89**	**99.45**	**98.49**	**97.7**	**96.25**	**95.49**	**95.10**	**83.49**

Case 2: Rainy day, a day when the accident happened and holidays for traffic prediction. The steps of prediction will be the same as discussed in case 1. The only difference between these two cases is that we will add features to a model dataset which will be further used by ARIMA for final predictions.

Table 8: Forecast Accuracy Results for Traffic Count Interval (15-minute dataset)

	Hybrid ARIMA			ARIMA		
	MAE	MAPE	RMSE	MAE	MAPE	RMSE
Rainy Day (Fri, Mar 3)	0.6417607	1.523321	0.83977	11.19758	13.310527	14.19007

Accident (Thus, July 27)	0.9878688	2.186649	1.307614	15.14474	16.20278	18.1965
Holiday (Tue Dec 26)	0.5264768	1.667271	0.6604433	12.8223	19.3160679	22.241

Table 8 and Table 9 shows that including features data improve the accuracy of prediction. The forecast result of Table 8 for a rainy day, holiday and day after a holiday is 98.48%, 97.82%, and 98.33% MAPE, respectively while for the same days if features are excluded then accuracy measure MAPE drops to 86.69%, 83.80%, and 80.68%. That depicts the external factors directly relate to traffic count.

Table 9: Forecast Accuracy Results for Traffic Count Interval (1-hour dataset)

	Hybrid ARIMA			ARIMA		
	MAE	MAPE	RMSE	MAE	MAPE	RMSE
Rainy Day (Thu, Nov 30)	2.839904	2.942747	3.474441	14.97356	17.20757	18.23457
Holiday (Thu Nov 23)	1.797659	1.306445	2.292871	17.30464	18.339838	19.30214
Day after Holiday (Thu Dec 26)	0.842441	1.211623	2.504301	17.97094	17.315016	19.46874

The same behavior is noticed for 1-hour dataset as well. From Table 9 for a rainy day, holiday and day after a holiday is 97.06%, 98.7%, and 98.79% MAPE, while MAPE drops to 82.02%, 81.67%, and 82.68 % for the same days without features information.

Table 10: Forecast Accuracy Results for Long-Term Prediction

Long-Term Prediction (MAPE)	1st prediction	2nd Prediction	3rd Prediction
Rainy Day (Thu, Nov 30)	96.02%	95.45%	93.48%
Holiday (Thu Nov 23)	93.56%	92.76%	93.49%

Table 10 represents the results when prediction duration is derived from zero crossing points. The prediction day 30th Nov with featured data is used in 1st prediction, the trend from 8 a.m. to 2 p.m. of day on 8th November is closely matching based on distance calculation formula. Hence, the first prediction is up to 2 p.m. on that day. The second prediction trend is closely matched with 15th November from 2 p.m. to 7 p.m. The last prediction trend is closely following the 29th November. The average prediction accuracy achieved with this method is 94.98%.
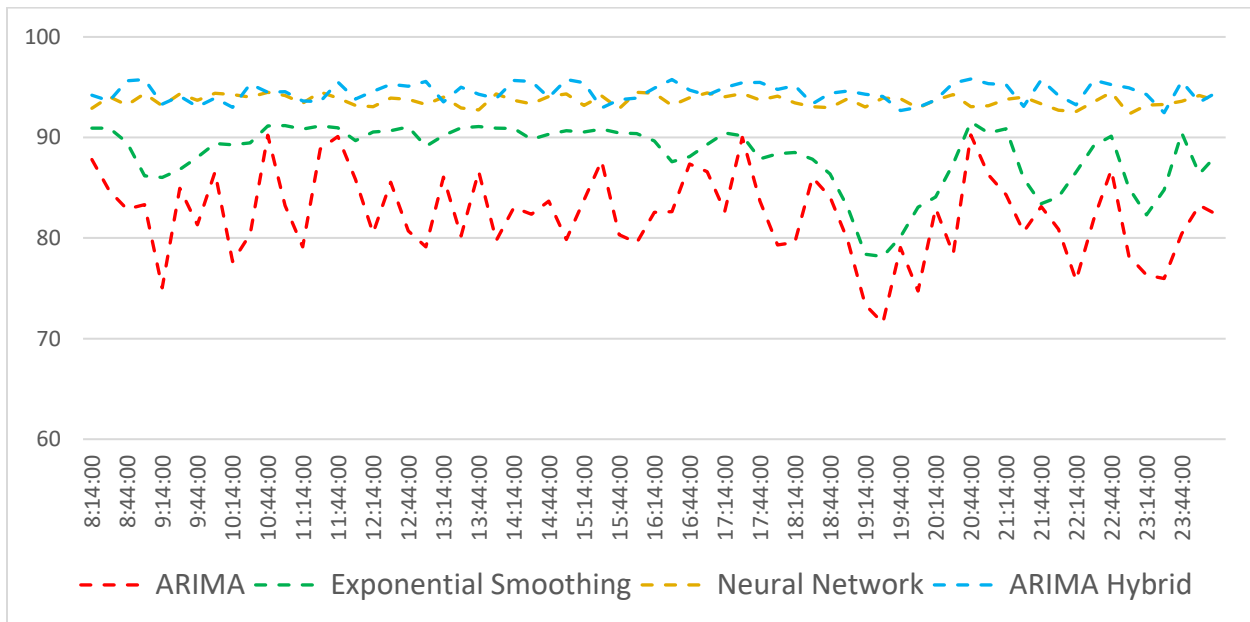
Figure 32: Comparison with different Forecast Models

The forecasting result in Figure 32 is of short-term results for a single day from different time series prediction models. It can be noticed that the proposed model outperformed among all. The classic ARIMA model has a lowest average accuracy of 82.33%. The Exponential Smoothing model has 88.27% average accuracy. The neural network model has better performance result than any other model with an accuracy of 93.67%. The neural network exhibits similar results as Hybrid ARIMA. However, the proposed model of prediction yields better results with the highest accuracy of 95.49%.

Table 11: Short Term Forecast Accuracy for one day

Prediction Models	MAPE (%)
ARIMA	82.33
Exponential Smoothing	88.27
Neural Network	93.67
Hybrid ARIMA	**95.49**

The results are obtained under some of the constraints. The optimal ARIMA model achieved in this paper does not use the validation in the process. The fuzzy rules defined in this algorithm used binary value to capture the presence of features instead of actual value. For example, the effect of snow is captured using 1 or 0 instead of using amount of snowfall. Though the neural network in experiment used only one hidden layer, very good accuracy was achieved when it was used to replace ARIMA in the hybrid approach for traffic flow prediction.

5. Conclusion

As the civilization is developing the problem of traffic on the road is increasing on daily basis. To help people tackle with traffic problems the best option is to know the situation of traffic in advance, so people can manage their work and save their time by adjusting their daily schedule of traveling. In thesis proposal, we presented a time series forecasting based on the combination of Fuzzy logic, wavelet analysis, and ARIMA. We have shown how the fuzzy logic is built to categorize data to provide more accurate data in the prediction model. The results state that if data is classified then better accuracy can be achieved. Usually time series ARIMA forecasting model uses past data with lagged value. Fuzzy logic in this model is used to build the dataset by considering only those past data which are most relevant to the prediction duration. For example, if the day is Monday and season is winter, then only those data with higher frequency matching will be selected which helps to get the more fitted model in ARIMA process. The multi-scaling property of the DWT enhances the prediction of non-linear and denoised traffic time series, and finally, ARIMA is used as a prediction model for long-term as well as short-term traffic prediction. For long-term traffic prediction, we have used sharp variation points to examine the detail level of traffic fluctuation to forecast for a certain period.

- Model:

The model builds in this thesis using the hybrid approach is trustful under all circumstances. The model can work for adjustable time spans, short-term as well as long-term. The use of DWT handles the missing information, capture the changing trend of information and work with stochastic data. The ARIMA being classical forecasting method along with fuzzy and DWT shows the more improved results.

- Accuracy:

The use of feature analysis helps to predict the traffic by looking at similar features related to it and improves the accuracy of forecasting. If the availability of data is high, then the

classification can be even more accurate. This helps to build the model for more accurate prediction.

In future, this method can be improved by using more features that affect the traffic counts. Also, from the result, we can see as the time span increase the accuracy is decreasing. Recent machine learning approach can help to improve the accuracy for long-term prediction such as neural network and a deep learning approach. By adding more recent work of data mining to identify the impact of external factors on traffic data can help to build the fuzzy logic rules more accurately. More work can be done to relate the features internally and selecting those features dynamically that are affecting the traffic count for that duration. The fuzzy rules are defined manually, we can introduce other machine learning techniques to learn the effect of features and build the rules set dynamically. No visualization tool has been added in this study. Introduction of visualization can help the user to understand the forecast more easily. ARIMA only handles the linear part of time series another method can be used to deal with non- linear part of time series.

References

[1] L. LI, W.-H. Lin and H. Liu, "Type-2 Fuzzy Logic Approach for Short-term Traffic Forecasting," in *IEE Proceedings-Intelligent Transport Systems*, vol. 153, IET, 2006, pp. 33--40.

[2] "Intelligent Transportation Systems," People.ece.umn.edu, [Online]. Available: http://people.ece.umn.edu/~mhong/Project_ITS.html. [Accessed 17 09 2018].

[3] M. Jiber, I. Lamouik, Y. Ali and A. Sabri, "Traffic flow prediction using neural network," in *Intelligent Systems and Computer Vision (ISCV), 2018 International Conference on*, IEEE, 2018, pp. 1--4.

[4] J. Hossain, "A Hybrid Approach of Traffic Flow Prediction Using Wavelet Transform and Fuzzy Logic," University of Windsor, Windsor, 2017.

[5] T. J. Ross, Fuzzy logic with engineering applications, John Wiley & Sons, 2009.

[6] Y. Lv, Y. Duan, W. Kang, Z. Li, F.-Y. Wang and others, "Traffic flow prediction with big data: A deep learning approach," *IEEE Trans. Intelligent Transportation Systems,* vol. 16, no. 2, pp. 865--873, 2015.

[7] I. Okutani and Y. . J. Stephanedes, "Dynamic prediction of traffic volume through Kalman filtering theory," *Transportation Research Part B: Methodological,* vol. 18, no. 1, pp. 1--11, 1984.

[8] M. o. T. Ontario, Advanced Traffic Management System, 2007.

[9] H. Chang, Y. Lee, B. Yoon and S. Baek, "Dynamic near-term traffic flow prediction: system-oriented approach based on past experiences," *IET intelligent transport systems,* vol. 6, no. 3, pp. 292--305, 2012.

[10] F. G. Habtemichael and M. Cetin, "Short-term traffic flow rate forecasting based on identifying similar traffic patterns," *Transportation Research Part C: Emerging Technologies,* vol. 66, pp. 61--78, 2016.

[11] B. M. Williams and L. A. Hoel, "Modeling and forecasting vehicular traffic flow as a seasonal ARIMA process: Theoretical basis and empirical results," *Journal of transportation engineering,* vol. 129, no. 6, pp. 664--672, 2003.

[12] R. Chrobok, J. Wahle and M. Schreckenberg, "Traffic forecast using simulations of large scale networks," in *Intelligent Transportation Systems, 2001. Proceedings. 2001 IEEE*, IEEE, 2001, pp. 434--439.

[13] J. Rice and E. v. Zwet, "A simple and effective method for predicting travel times on freeways," in *Intelligent Transportation Systems, 2001. Proceedings. 2001 IEEE*, IEEE, 2001, pp. 227--232.

[14] B. Pan, C. Shahabi and U. Demiryurek, *Traffic prediction using real-world transportation data,* Google Patents, 2018.

[15] A. Koesdwiady, R. Soua and F. Karray, "Improving traffic flow prediction with weather information in connected cars: a deep learning approach," *IEEE Transactions on Vehicular Technology,* vol. 65, no. 12, pp. 9508--9517, 2016.

[16] S. Dunne and B. Ghosh, "Weather adaptive traffic prediction using neurowavelet models," *IEEE Transactions on Intelligent Transportation Systems,* vol. 14, no. 1, pp. 370--379, 2013.

[17] M. Agarwal, T. H. Maze and R. Souleyrette, "Impacts of weather on urban freeway traffic flow characteristics and facility capacity," in *Center Transp. Res. Educ., Ames, IA*, Integration of Road Weather Information With Traffic Data—Aurora Project, 2005, pp. 18--19.

[18] Khandelwal, Ina, R. Adhikari and G. Verma, "Time series forecasting using hybrid ARIMA and ANN models based on DWT decomposition," *Procedia Computer Science,* vol. 48, pp. 173-179, 2015.

[19] L. A. Zedeh, "Fuzzy Sets," *Information and control,* vol. 8, no. 3, pp. 338--353, 1965.

[20] "Fuzzy Logic Tutorial," Encoder- The Newsletter of the Seattle Robotics Society, [Online]. Available: http://www.seattlerobotics.org. [Accessed 14 07 2018].

[21] H. Peng, S. U. Bobade, M. E. Cotterell and J. A. Miller, "Forecasting Traffic Flow: Short Term, Long Term, and When It Rains," in *International Conference on Big Data*, Springer, 2018, pp. 57--71.

[22] W. Huang, W. Jia and J. Guo, "Real-time prediction of seasonal heteroscedasticity in vehicular traffic flow series," *IEEE Transactions on Intelligent Transportation Systems,* vol. PP, no. 99, pp. 1--11, 2017.

[23] N. Barimani, A. R. Kian and B. Moshiri, "Real time adaptive non-linear estimator/predictor design for traffic systems with inadequate detectors," *IET Intelligent Transport Systems,* vol. 8, no. 3, pp. 308--321, 2014.

[24] S. Jin, D.-h. Wang, C. Xu and D.-f. Ma, "Short-term traffic safety forecasting using Gaussian mixture model and Kalman filter," *Journal of Zhejiang University SCIENCE A,* vol. 14, no. 4, pp. 231--243, 2013.

[25] Z. Jiang, C. Zhang and Y. Xia, "Travel time prediction model for urban road network based on multi-source data," *Procedia-Social and Behavioral Sciences,* vol. 138, pp. 811--818, 2014.

[26] B. Yoon and H. Chang, "Potentialities of data-driven nonparametric regression in urban signalized traffic flow forecasting," *Journal of Transportation Engineering,* vol. 140, no. 7, p. 04014027, 2014.

[27] Z. Diao, D. Zhang, X. Wang, K. Xie, S. He, X. Lu and Y. Li, "A Hybrid Model For Short-Term Traffic Volume Prediction In Massive Transportation Systems," *IEEE Transactions on Intelligent Transportation Systems,* vol. PP, no. 99, pp. 1--12, 2018.

[28] Y. Liu and R. Wang, "Study on network traffic forecast model of SVR optimized by GAFSA," *Chaos, Solitons & Fractals,* vol. 89, pp. 153--159, 2016.

[29] Y. Tian and . L. Pan, "Predicting short-term traffic flow by long short-term memory recurrent neural network," *Smart City/SocialCom/SustainCom (SmartCity), 2015 IEEE International Conference on,* pp. 153--158, 2015.

[30] I. Khandelwal, R. Adhikari and G. Verma, "Time series forecasting using hybrid ARIMA and ANN models based on DWT decomposition," *Procedia Computer Science,* vol. 48, pp. 173--179, 2015.

[31] G. P. Zhang, "Time series forecasting using a hybrid ARIMA and neural network model," *Neurocomputing,* vol. 50, pp. 159--175, 2003.

[32] Y. Wang, L. Li and X. Xu, "A Piecewise Hybrid of ARIMA and SVMs for Short-Term Traffic Flow Prediction," in *International Conference on Neural Information Processing*, Springer, 2017, pp. 493--502.

[33] J. Tang, F. Liu, Y. Zou, W. Zhang and Y. Wang, "An improved fuzzy neural network for traffic speed prediction considering periodic characteristic," *IEEE Transactions on Intelligent Transportation Systems,* vol. 18, no. 9, pp. 2340--2350, 2017.

[34] J. Tang, F. Liu, W. Zhang, R. Ke and Y. Zou, "Lane-changes prediction based on adaptive fuzzy neural network," *Expert Systems with Applications,* vol. 91, pp. 452--463, 2018.

[35] I. Khandelwal, R. Adhikari and G. Verma, "Forecasting Of Time Series Data Using Hybrid ARIMA Model With The Wavelet Transform," *International Research Journal of Engineering and Technology,* vol. 05, no. 6, p. 1048–1055, 2018.

[36] N. Zaini, . M. A. Malek, M. Yusoff, N. H. Mardi and . S. Norhisham, "Daily River Flow Forecasting with Hybrid Support Vector," in *IOP Conference Series: Earth and Environmental Science*, IOP Publishing, 2018, p. 012035.

[37] N. I. Sapankevych and R. Sankar, "Time series prediction using support vector machines: a survey," *IEEE Computational Intelligence Magazine,* vol. 4, no. 2, pp. 24-38, 2009.

[38] Y. Bai and D. Wang, "Fundamentals of fuzzy logic control—fuzzy sets, fuzzy rules and defuzzifications," in *Advanced Fuzzy Logic Technologies in Industrial Applications*, Springer, 2006, pp. 17--36.

[39] R. Polikar, *The wavelet tutorial,* Rowan University, College of Engineering Web Servers, 1996.

[40] F. Ababneh, S. A. Wadi and M. T. Ismail , "Haar and Daubechies Wavelet Methods in Modeling Banking Sector," in *International Mathematical Forum*, Citeseer, 2013, pp. 551--566.

[41] T. W. Joo and S. B. Kim, "Time series forecasting based on wavelet filtering," *Expert Systems with Applications,* vol. 42, no. 8, pp. 3868--3874, 2015.

[42] M. A. Wadia and M. Tahir Ismail, "Selecting wavelet transforms model in forecasting financial time series data based on ARIMA model," *Applied Mathematical Sciences,* vol. 5, no. 7, pp. 315--326, 2011.

[43] L. Duval, "Should I ever pick the continuous wavelet transform over the discrete one? DWT vs CWT vs STFT," 25 05 2017. [Online]. Available: https://dsp.stackexchange.com/questions/41240/should-i-ever-pick-the-continuous-wavelet-transform-over-the-discrete-one-dwt-v. [Accessed 23 7 2018].

[44] P. CHAOVALIT, A. GANGOPADHYAY, G. KARABATIS and Z. CHEN, "Discrete wavelet transform-based time series analysis and mining," *ACM Computing Surveys (CSUR),* vol. 43, no. 2, p. 6, 2011.

[45] C. Stolojescu, I. Railean, S. Moga, P. Lenca and A. Isar, "A wavelet based prediction method for time series," in *Proceedings of Stochastic Modeling Techniques and Data Analysis (SMTDA2010) International Conference*, Chania, Greece, 2010.

[46] P. Sugiartawan, R. Pulungan and A. K. Sari, "Prediction by a hybrid of wavelet transform and long-short-term-memory neural network," *International Journal of Advanced Computer Science and Applications,* vol. 8, no. 2, pp. 326--332, 2017.

[47] Z. Yong and L. Shi-Gao, "Analysis of scale-free characteristic on sharp variation point of traffic flow," *Acta Physica Sinica,* vol. 63, no. 24, pp. 1--6, 2014.

[48] M. Zhang, "Real-time traffic flow prediction using augmented reality," University of Windsor, Windsor, 2016.

[49] X. ZHAO and P. XIAO, "Wavelet-based the character recognition in map," *IAPRS,* vol. 34, no. 2, pp. 605--608, 2002.

[50] G. M. JENKINS and G. E. BOX, Time Series Analysis Forecasting and Control, WISCONSIN UNIV MADISON DEPT OF STATISTICS, 1970.

[51] "Multicolinearity and Categorical variables in regression models," [Online]. Available: https://modelovanie.online/10/. [Accessed 11 09 2018].

[52] R. Nau, "Identifying the orders of AR and MA terms in an ARIMA model," People.duke.edu, [Online]. Available: https://people.duke.edu/~rnau/411arim3.htm. [Accessed 17 09 2018].

[53] R. Dalinina and O. DataScience.com, "Introduction to K-means Clustering," Datascience.com, 1 10 2017. [Online]. Available: https://www.datascience.com/learn-data-science/tutorials/introduction-to-forecasting-with-arima-r-data-science. [Accessed 11 9 2018].

[54] "Akaike or Bayesian information criteria - MATLAB aicbic," Mathworks.com, [Online]. Available: http://www.mathworks.com/help/econ/aicbic.html#btvzxaa. [Accessed 16 09 2018].

[55] J. Kwon, M. Mauch and P. Varaiya, "Components of congestion: Delay from incidents, special events, lane closures, weather, potential ramp metering gain, and excess demand," *Transportation Research Record,* vol. 1959, no. 1, pp. 84--91, 2006.

[56] A. Borovykh, S. Bohte and C. W. Oosterlee, "Conditional Time Series Forecasting with Convolutional Neural Networks," Cornell University Library, Ithaca, 2017.

[57] "Highways England - WebTRIS - Map View," Webtris.highwaysengland.co.uk, [Online]. Available: http://webtris.highwaysengland.co.uk/. [Accessed 12 08 2018].

[58] "Datamillnorth.org," Datamillnorth.org, [Online]. Available: https://datamillnorth.org/dataset/m62-junction-25-to-30-traffic-flow. [Accessed 11 09 2018].

[59] B. Yao, C. Chen, Q. Cao, . L. Jin, M. Zhang, H. Zhu and . B. Yu, "Short-term traffic speed prediction for an urban corridor," *Computer-Aided Civil and Infrastructure Engineering,* vol. 32, no. 2, pp. 154--169, 2017.

[60] S. V. Kumar, "Traffic flow prediction using kalman filtering technique," *Procedia Engineering,* vol. 187, pp. 582--587, 2017.

[61] Y. Li-Sheng, H. Yi-Gang, D. Xue-Ping and L. Zhao-Quan, "Research on the Multi-step Prediction of Volterra Neural Network for Traffic Flow," *Acta Automatica Sinica,* vol. 9, p. 029, 2014.

[62] G. Omkar and S. V. Kumar, "Time series decomposition model for traffic flow forecasting in urban midblock sections," *Smart Technologies For Smart Nation (SmartTechCon), 2017 International Conference On,* pp. 720--723, 2017.

[63] S. V. Kumar and L. Vanajakshi, "Short-term traffic flow prediction using seasonal ARIMA model with limited input data," *European Transport Research Review,* vol. 7, no. 3, p. 21, 2015.

[64] T. Ozaki, "On the order determination of ARIMA models," *Journal of the Royal Statistical Society. Series C (Applied Statistics),* vol. 26, no. 3, pp. 290--301, 1977.

[65] B. Sharma, V. K. Katiyar and A. K. Gupta, "Fuzzy logic model for the prediction of traffic volume in week days," *International Journal of Computer Applications,* vol. 107, no. 17, p. 0975 – 8887, 2014.

Vita Auctoris

NAME:	Sukruti Vaghasia
PLACE OF BIRTH:	Surat, Gujarat, India
YEAR OF BIRTH:	1992
EDUCATION:	Bachelor of Science in Information Technology, 2010-2013
	Veer Narmad South Gujarat University, Surat, Gujarat, India
	Master of Science in Information Technology, 2013-2015
	Veer Narmad South Gujarat University, Surat, Gujarat, India
	Master of Science in Computer Science, 2017-2018
	University of Windsor, Windsor, ON