Uncertainty-Aware Traffic Prediction using Attention-based Deep Hybrid Network with Bayesian Inference

Md. Moshiur Rahman[1], Abu Rafe Md Jamil[2], Naushin Nower[3]
Institute of Information Technology, University of Dhaka, Dhaka, Bangladesh[1,3]
Department of Computer Science and Engineering, Jashore University of Science and Technology, Jashore, Bangladesh[2]

***Abstract*—Traffic congestion has an adverse impact on the economy and quality of life and thus accurate traffic flow forecasting is critical for reducing congestion and enhancing transportation management. Recently, hybrid deep-learning approaches show promising contributions in prediction by handling various dynamic traffic features. Existing methods, however, frequently neglect the uncertainty associated with traffic estimates, resulting in inefficient decision-making and planning. To overcome these issues, this research presents an attention-based deep hybrid network with Bayesian inference. The suggested approach assesses the uncertainty associated with traffic projections and gives probabilistic estimates by applying Bayesian inference. The attention mechanism improves the ability of the model to detect unexpected situations that disrupt traffic flow. The proposed method is tested using real-world traffic data from Dhaka city, and the findings show that it outperforms than other cutting-edge approaches when used with real-world traffic statistics.**

Keywords*—*Traffic flow prediction; uncertainty; deep learning; Bayesian inference; Dhaka city

I. Introduction

Traffic congestion has a significant negative impact on the global economy as it decreases productivity, and increases waiting time, fuel consumption, air pollution, etc. According to INRIX, a major transportation analytics provider, estimates that the typical commuter in the top 1000 cities in the world will spend 99 hours stuck in traffic congestion in the year 2020 [1]. As a consequence of this, an estimated economic cost of $1,036 per commuter was incurred as a result of wasted time as well as fuel [2]. More specifically, Dhaka city, the capital of Bangladesh lost Tk56,000 crore (6.5 billion US Doller) in 2020 due to traffic congestion [3]. Thus for improving the country's economy and for better people's daily life, is it needed to improve the traffic condition. For example, reducing congestion in only Dhaka city gets to benefit from the massive economic growth of 35 percent of the country's GDP [3]. One promising way to mitigate traffic congestion is to accurately predict traffic flow since it helps users to make better route planning and city planners to effectively manage traffic throughout the city.

Extensive research has been conducted on traffic flow prediction, and several strategies have been developed over the years. Among them, deep hybrid approaches have recently gained attention and popularity because of their capacity to capture the complex dynamic nature of traffic data [4]. Traffic flow has complex dynamics as it depends on both space and time dimensions since it has spatiotemporal features. Recent approaches make use of the power of deep learning algorithms to identify complex spatiotemporal patterns and dependencies in the traffic flow. Besides these, traffic flow is also affected by the many sudden incidents such as accidents, rain, VIP movement, social event, and many more. The attention mechanism is incorporated with a deep hybrid approach to tackle this dynamic and sudden traffic nature. Although the deep hybrid methods can capture the spatiotemporal traffic pattern, it only generates a point prediction and neglects the inherently uncertain nature of the traffic data [5]. Uncertainty is associated with every outcome of any prediction algorithm and most of the existing prediction methods do not consider uncertainty of their predicted outcome. However, uncertainty is accompanied by predicted outcomes due to the overfitting nature of learning, lack of model knowledge, and also the dynamic nature of the traffic flow [6].

Prediction models can generate overly optimistic or gloomy projections when uncertainty is not appropriately addressed in traffic flow prediction. This can lead to poor decision-making and planning since they are providing deterministic predictions. If a forecast model, for instance, does not take into account the uncertainty associated with unexpected traffic events, then it is possible that it would underestimate or overestimate the levels of congestion.

To solve this issue, this paper proposes a prediction approach where Bayesian inference is incorporated with the attention-based deep hybrid network. The proposed method considers the network-traffic flow rather than a single road and considers the connectivity among the links to make better predictions. This method captures the spatiotemporal features of the traffic flow by using a deep hybrid network and handles the sudden incident using an attention mechanism. To handle uncertainty, we mainly incorporate Bayesian inference with attention-based deep hybrid network for prediction. Bayesian inference reduces uncertainty in predicted outcomes made by deep hybrid networks by providing probability. Probabilistic prediction can provide a better outcome to measure the uncertainty and risks of the predicted outcome since it provides an interval consisting of the lower and upper bound in which the future estimation should lie [7]. Thus we have incorporated Bayesian inference in our prediction model. The main contribution of this paper is three folds:

- Proposed model takes into account linked road information, which enables a more comprehensive and precise forecast of traffic flow. The utilization of this technique allows for the model to effectively apprehend the spatiotemporal interconnections and interdependencies among distinct road segments.

- To handle uncertainty, Bayesian inference is incorporated in the proposed method to measure the probability of the traffic flow propagating to a certain direction. Then, prediction is made by combining the traffic features and reducing uncertainty.

- The proposed approach is validated on the real-life traffic data of Dhaka city collected from Google Maps. To the best of our understanding, this is the first study that uses deep learning with Bayesian estimation to estimate traffic flow in Dhaka. The outcomes indicate that the proposed methodology outperforms all of the standard techniques in terms of prediction accuracy.

The remaining sections of the paper are structured as follows: Section II provides a comprehensive literature review, discussing existing works in the field. In Section III, we present essential background knowledge that underpins our proposed method. Section IV elaborates on our proposed methodology. The results and interpretations of our studies are presented in Section V. In Section VI, we delve into a related discussion surrounding our proposed method. Finally, Section VII offers a concise conclusion summarizing our findings.

II. Literature Review

Accurate prediction of traffic patterns plays a vital role in mitigating congestion and improving traffic flow. Many existing research has focused on traffic flow prediction, however, only a few of them have taken the uncertainty associated with these projections into account. Traffic flow is naturally unpredictable and it is critical to take this into account when making predictions. In this section, some research works that concentrate on uncertainty within prediction are addressed here.

Researchers aim to improve the precision and dependability of traffic forecast models by including uncertainty quantification. Ying Wu et al. [8], for example, develop a Bayesian deep learning model for traffic speed prediction with uncertainty quantification. This model uses ChebNet to capture the spatial feature and uses gated linear units (GLU) for temporal prediction. The model is designed to be a universal traffic forecasting framework and perform better in traffic flow and speed forecasting tasks both in prediction accuracy and handling uncertainty. However, this model can not be able to capture all the factors that affect traffic speed, such as weather conditions and sudden incidents.

Another recent work proposed by Genwang Liu et. al. [9] focuses on the problem of incident detection on freeways and addresses the challenge of uncertainty quantification. The proposed method utilizes a variant of convolutional neural networks (CNN) within a Bayesian framework. The weight of the model is updated using mechanisms such as Bayes by backpropagation and local reparameterization techniques. By integrating the aleatoric uncertainty (uncertainty in the data) and epistemic uncertainty (uncertainty in the model), the method models the predictive uncertainty comprehensively. The results of the experiments indicate that the aleatoric uncertainty of the model remains relatively stable under different noise levels.

Mundher Seger et al. [10] presents a Monte Carlo simulation-based method for quantifying uncertainty in traffic assignments, as well as insights into the unpredictability and bias of expected traffic flows. The authors created an approach that utilizes Monte Carlo simulation to evaluate uncertainty in traffic flows. The values of the origin-destination (OD) matrix were handled as stochastic variables with a specified probability distribution. The methodology calculated values for every link by simulating traffic patterns on the transportation network. This work focuses on four scenarios that can occur when uncertainty exists: Case 1: low prediction uncertainty, Case 2: medium prediction uncertainty, Case 3: large prediction uncertainty with ensemble agreement, and Case 4: severe prediction uncertainty with divergence estimates. They also discovered that traffic flow uncertainty occurs on all transportation network links, but to varying degrees, depending on the scenario's specifications and actual traffic flow.

To deal with the issue of low fitting between projected and real values in existing research methodologies, Lingmin Yang [11] proposes a traffic flow uncertainty prediction approach based on the K-nearest neighbor (KNN) algorithm. To generate the necessary database for the prediction process, the suggested method uses numerous databases, comprising the original database, classification center database, k-nearest neighbor database, and intermediate search database. The method employs multivariate linear regression to assign weights to state variables, taking into account the uncertainties of traffic flow. The K-nearest neighbor algorithm and Kalman filter are then utilized to update the weights iteratively, adapting them to the evolving uncertainties of traffic flow. Through this iterative process, the predicted values of traffic flow uncertainties are obtained. In their paper, they mainly handle the uncertainity by considering linked road information. When predicting a road segment they consider the other connected road traffic condition. Their experimental result shows that the model achieves good accuracy but it still suffers from uncertainity and the sudden incident can not be handled by the method.

For traffic prediction, Jun Fu et al. [12] suggests a Bayesian Spatio-Temporal Graph Convolutional Network (BSTGCN). This method learns the graph structure using both the physical road network topology and the traffic data. Graph convolutional networks (GCN) are utilized for expressing traffic data as well as the physical structure of road networks as graphs. This enables a more accurate depiction of the intricate interactions between traffic flows to be captured. Furthermore, they provide a probabilistic generative model for expressing the graph structure, which improves the generalization capability of GCNs and handles uncertainty.

III. Background

In this paper, we use ConvLSTM to handle spatiotemporal data, attention mechanism to handle sudden incidents and Bayesian Inference to handle uncertainity. In the following section, the background of this models is discussed.

A. Convolutional Long Short-Term Memory

ConvLSTM (Convolutional Long Short-Term Memory) is a type of recurrent neural network (RNN) that is intended to handle spatiotemporal data. It uses the strengths of both convolutional neural networks (CNNs) and long short-term memory (LSTMs) to make a better model. This makes ConvLSTM very good at jobs that involve sequential data with a spatial structure, like analyzing videos, predicting the weather, and predicting traffic.

In ConvLSTM, the input data is in a matrix format, where the measurements are width, height, and time steps.

$$X_t^s = \begin{bmatrix} V_{t-m}^s \\ . \\ . \\ V_{t-1}^s \\ V_t^s \end{bmatrix} = \begin{bmatrix} V_{t-m}^1 & V_{t-m}^2 & .. & V_{t-m}^n \\ . & . & . & . \\ V_{t-1}^1 & V_{t-1}^2 & .. & V_{t-1}^n \\ V_t^1 & V_t^2 & .. & V_t^n \end{bmatrix} \quad (1)$$

where X_t^s represents the road networks state, including $m+1$ time periods and n road segments. $V_t^s = [V_t^1, V_t^2, ..., V_t^n]$ denotes traffic intensity of all road segments in the road networks at time t. The convolutional layer and pooling layer of the first CNN takes the input data and convert multi-dimensional data V_t^s into one-dimensional data for LSTM. The main idea behind ConvLSTM is to record spatial dependencies by using convolutional operations instead of fully connected layers in the LSTM cell. This lets the model take advantage of the local connectedness and parameter-sharing features of CNNs, which are good for dealing with spatial data. The equations for ConvLSTM can be expressed as follows:

Input Gate:

$$i_t = \sigma(W_{xi} * X_t + W_{hi} * H_{t-1} + W_{ci} \circ C_{t-1} + b_i) \quad (2)$$

Forget Gate:

$$f_t = \sigma(W_{gf} * O_t^s + W_{hf} * H_{t-1}^s + W_{cf} \circ C_{t-1} + b_f) \quad (3)$$

Cell Update:

$$C_t = f_t \circ C_{t-1} + i_t \circ tanh(W_{gc} * O_t^s + W_{hc} * H_{t-1}^s + W_{hc} * H_{t-1}^s + b_c) \quad (4)$$

Output Gate:

$$O_t = \sigma(W_{go} * O_t^s + W_{ho} * H_{t-1}^s + W_{co} \circ C_t + b_o) \quad (5)$$

In these equations, X_t represents the input tensor at time step t, H_{t1} represents the hidden state tensor from the previous time step, and C_{t1} represents the cell state tensor from the previous time step. The $*$ operator denotes the convolution operation, $\circ$ represents the Hadamard product (element-wise multiplication), and σ denotes the sigmoid activation function. The ConvLSTM cell consists of input gates (i_t), forget gates (f_t), and output gates (o_t) that control the flow of information within the cell. The cell state (C_t) is updated based on the input, previous cell state, and the gates. The hidden state (H_t) is computed by applying the output gate to the cell state passed through the hyperbolic tangent activation function. By using convolutional operations within the LSTM cell, ConvLSTM can describe complex spatiotemporal patterns in the data. It does this by capturing spatial dependencies across different time steps. This makes ConvLSTM a powerful tool for analyzing and predicting spatiotemporal data.

B. Attention Mechanism

The attention mechanism is crucial for enhancing the performance and effectiveness of traffic flow prediction systems. It enables the model to focus on important spatial and temporal information, focusing on specific regions or time steps that are more useful for the prediction aim [13]. Incorporating attention processes can help collect complicated trends, dependence, and changes in traffic data, leading to more precise and interpretable forecasts. To gain the benefits of the attention mechanism in traffic flow prediction, a combination of two ConvLSTM layers can be used. ConvLSTM layers are RNN layers that combine convolutional layers with LSTM (Long Short-Term Memory) units. These layers are quite helpful in mimicking spatiotemporal dependencies in traffic data. By combining spatial information from the first ConvLSTM layer with temporal information from the second ConvLSTM layer, the model can successfully capture both local spatial patterns and long-term temporal correlations in traffic data. This combination enables the attention mechanism to prioritize important spatial and temporal regions, resulting in more accurate predictions from the model.

C. Bayesian Inference

Bayesian inference is an approach to statistics that uses Bayes' theorem to revise our assumptions or probabilities depending on observed data. Bayesian inference represents the uncertainty related to predictions in a probabilistic manner [14]. Bayesian inference, rather than offering a single-point projection, provides a posterior probability range that defines the range of alternative outcomes and associated likelihoods. This distribution represents the prediction's uncertainty, providing decision-makers with a full picture of the probable outcomes. The Bayes theorem can be represented mathematically as:

$$P(H|D) = \frac{P(D|H).P(H)}{P(D)} \quad (6)$$

where, $P(H|D)$ is the posterior probability of hypothesis H given the observed data D.
$P(D|H)$ is the likelihood of observing data given hypothesis H.
$P(H)$ is the prior probability of hypothesis H before observing the data.
$P(D)$ is the probability of observing the data D.

The Bayesian inference posterior distribution provides a quantifiable measure of uncertainty. It can be used to compute statistics such as confidence intervals or reasonable intervals, which specify the range of values that the real value of a parameter or variable is expected to fall. These intervals indicate the prediction's uncertainty and serve as a measure of the confidence or dependability associated with the estimations.

IV. Proposed Method

In this section we have presented our proposed prediction approach with Bayesian Inference which incorporated linked road traffic information using graph. We start with a road network graph $G(V, E)$, where V is the set of N roads and E is the set of edges. We also collect historical information about the road segments. $X_{1:t} = \{X_1, X_2, X_3, ...X_t\}$, where X_t is a member of $\mathbb{R}^{N \times n}$ and n is the size of the traffic data. Our

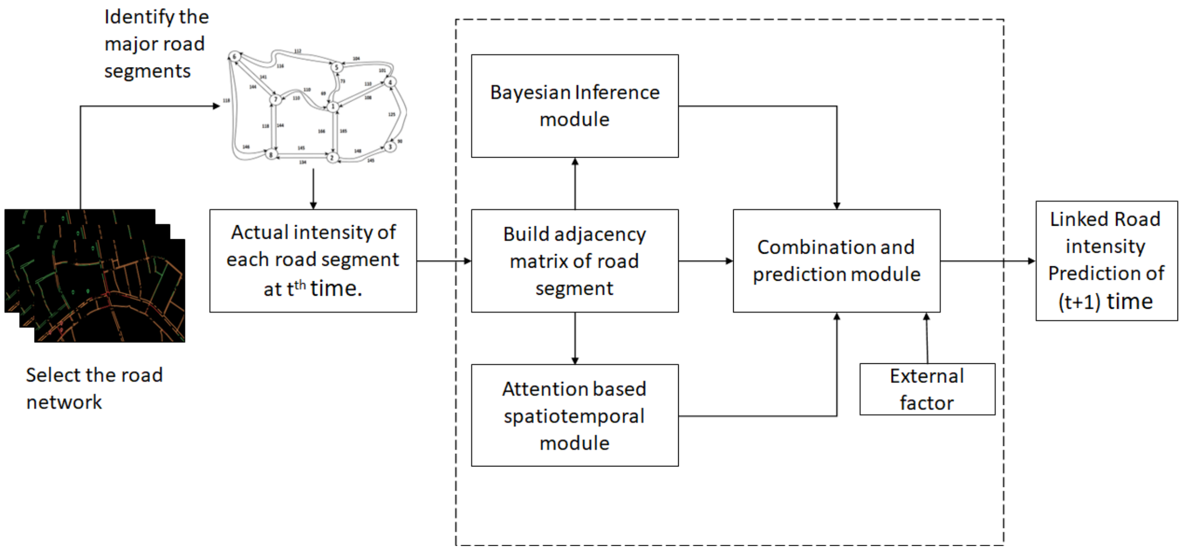


Fig. 1. Overall process of our proposed attention-based deep hybrid network with bayesian inference.

main goal is to predict the traffic flow for the next T timesteps, from X_{t+1} to X_{t+T}, based on collected historical data. For accurate traffic flow prediction, we proposed a technique that incorporates three modules, an attention-based spatiotemporal module, a Bayesian inference module, and a prediction module. An attention-based spatiotemporal module (H) can handle the spatiotemporal nature of traffic data and also handle sudden incidents. The Bayesian inference module (B) handles the uncertainty of traffic prediction. Lastly, the prediction module (P) combines the other modules and produces the next timesteps traffic flow prediction X_{t+T}. So our objective function is used to minimize the prediction error by reducing uncertainty which can be written as

$$min(\|Y_i - P(X_{t+1:t+T} : X_{1:t}, G(V, E), H, B, P)\|) \quad (7)$$

Our proposed model's main objective is to produce a more accurate prediction and the difference should be minimized from the actual value. In equation 7 Y_i is the actual value of traffic flow. Fig. 1 represents the proposed attention-based deep hybrid network with Bayesian inference. In the following subsection, we describe the procedures of the modules.

A. Attention-based Spatiotemporal Module

The Attention-based spatiotemporal module is a key component of the proposed model, designed to calculate the spatiotemporal weight (STW). This module utilizes two ConvLSTM layers to capture the temporal dependencies in the data. The input to the module consists of the data from each road segment, along with external factors such as temperature, holidays, or any other relevant information. These inputs are processed by the ConvLSTM layers, which enable the model to learn and capture the temporal patterns and dependencies in the data. By incorporating the external factors, the module can account for their influence on the spatiotemporal weight calculation. This allows the model to adapt its predictions based on specific contextual information, such as the impact of temperature or holidays on traffic patterns. The internal structure of the module is shown in Fig. 2.

B. Bayesian Inference Module

Most existing works only predict traffic flow using spatiotemporal weights without considering the data's uncertainty. Those predictions neglect the diversity and uncertainty of data and network parameters and provide a deterministic forecast. Indeed, traffic conditions can vary between different days and times, such as the difference between a Sunday at 8:00 AM and a Monday or the following Sunday. This variation can lead to differing levels of congestion. However, if a deep learning model overfits the data, it may struggle to capture this uncertainty accurately, resulting in uncertain predictions. To overcome this problem, the proposed framework incorporated Bayesian inference with an attentive spatiotemporal convolutional network to handle uncertainty within the prediction. Bayesian inference determines the probability associated with certain predictions which defines the probability of a particular prediction.

Incorporating Bayesian inference with the transitional probability (TP) calculation can help to handle uncertainty and provide a more accurate estimate of the probability of traffic flow propagation. We used the Bayesian theorem to calculate the posterior probability of a particular road segment given the collected traffic data, which is represented as $P(i|D(t))$. Then, using the adjacency matrix of the graph, we calculate the prior probability of being on that road segment at the next time step, represented as $P(i|t+1, D(t))$. Using the prior and posterior probabilities, the transitional probability of propagating traffic from road segment i to road segment j at time t, represented as $P(i \rightarrow j|t)$ is calculated.

$$P(i \rightarrow j|t) = P(j|i, t) \cdot P(i|t + 1, D(t)) \quad (8)$$

where $P(j|i, t)$ is the conditional probability of propagating from road segment i to road segment j within a given time period, and $D(t)$ refers to the traffic data collected at a particular time step t. By combining the Bayesian inference with the transitional probability calculation, we can obtain the Bayesian transitional probability of propagating traffic from road segment i to road segment j at time t, which is

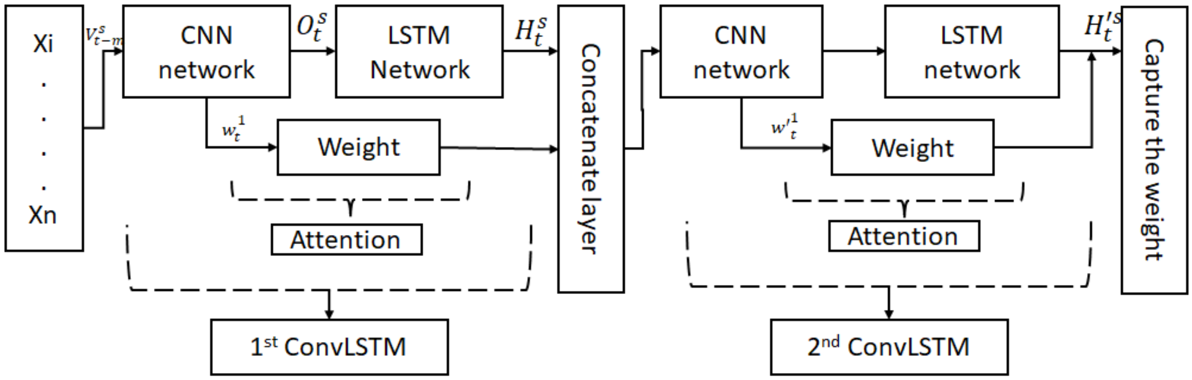


Fig. 2. Internal structure of attention-based spatiotemporal module.

represented as $P(i \to j|D(t))$.

$$P(i \to j|D(t)) = P(j|i,t) \cdot P(i|D(t)) \tag{9}$$

This probability represents the likelihood of traffic flow transitioning from segment i to segment j given the historical traffic data. We calculate the posterior probability of each road segment using the collected traffic data, then calculate the prior probability using the adjacency matrix of the graph. Finally, we use the prior and posterior probabilities to obtain the Bayesian transitional probability of traffic flow transitions between road segments in the network. Bayesian inference with the transitional probability calculation using the last two-time adjacency matrix, we can obtain the Bayesian transitional probability of traffic flow transitions between road segments in the network. This probability takes into account the uncertainty in the data and network parameters, providing a more accurate estimate of the likelihood of traffic flow transitions.

C. Combination and Prediction Module

After finding the matrices such as the transitional probability matrix represented as (TP) and the spatiotemporal weight matrix as (STW). The process of finding these values is described above. To predict the intensity of the next time step (t), we combine spatiotemporal weights (STW), transitional probability (TP), and actual intensity matrix (A) of the previous time step $(t-1)$. We combine the spatiotemporal and transitional probability of the previous time step $(t-1)$ using equation 10 and find the combined weight as (SPW), which handles spatiotemporal traffic features and uncertainty. Then, the prediction is made by multiplying previous time steps combined weight (SPW^{t-1}) with the actual intensity matrix (A^{t-1}) as shown in equation 11. Where (A^t) represents the next timesteps prediction outcomes. Algorithm 1 shows the overall process of the proposed prediction architecture where three dynamic characteristics of traffic flow are incorporated.

$$P^{t-1} = STW^{t-1} + TP^{t-1} \tag{10}$$

$$A^t = A^{t-1} \times P^{t-1} \tag{11}$$

Algorithm 1 Proposed Prediction Method: Attention based deep hybrid prediction network with Bayesian inference

procedure PREDICTNEXTTIMESTEP$(X, D(t), I)$
Attention-based spatiotemporal module
  $n \leftarrow$ number of road segments
  $m \leftarrow$ number of time steps
  $X' \leftarrow$ input data from CSV file, reshaped to $n \times m$
  $h_1, c_1 \leftarrow$ initial hidden and cell states for ConvLSTM 1
  $h_2, c_2 \leftarrow$ initial hidden and cell states for ConvLSTM 2
  for $t \leftarrow 1$ to m **do**
    $X_t \leftarrow$ input tensor of shape $n \times 1 \times 1$
    $H_1, c_1 \leftarrow$ ConvLSTM 1(X_t, h_1, c_1)
    $H_2, c_2 \leftarrow$ ConvLSTM 2(H_1, h_2, c_2)
    $I_t \leftarrow$ attention weights from H_2
    $H_t \leftarrow$ weighted sum of H_2 using A_t
  end for
Bayesian Inference module
  $W \leftarrow$ BayesianTransitional$(D(t), A^{t-1}, A^t)$
Combination and prediction module
  $A' \leftarrow$ last time step intensity matrix from I
  $P \leftarrow$ empty matrix of size $n \times n$
  for $i \leftarrow 1$ to n **do**
    for $j \leftarrow 1$ to n **do**
      $P_{i,j} \leftarrow W_{i,j}$
    end for
  end for
  $A^t \leftarrow P \times A^{t-1}$
  return A^{t-1}
end procedure

V. SIMULATION AND RESULT ANALYSIS

To evaluate our proposed approach we have compared it with that of the states of the art method and this section provides the details of the simulation setup and result analysis.

A. Data Description

To evaluate our proposed model we have chosen Dhaka city's traffic data as a case study since it is the fifth most congested city in the world. The real-life traffic data of Dhaka city is collected from Google map images using the tool [15, 16] available at [17].

We have worked on road network topology rather than a single road, thus it is needed to collect traffic data from individual road segments. To achieve this, we first select a zone and identify the major road segments. Then collect the starting and ending latitude and longitude of each road segment and collect traffic data from every road segment. For our simulation, we have selected the Shahbag region of Dhaka city and selected twelve main road segments, each with two lanes thus a total of twenty-four roads. Fig. 3(a) represents the Shahbag region road network with eight-road intersections and Fig. 3 (b) represents the graph that is built considering the selected road network. We need to collect each road segment data separately and then combine it according to the graph network. For simulation, we collect one-month data. One month of data provides a sufficient amount of data for short-term traffic flow modeling and analysis [18]. Our collected data set comprises a total of 3720 data instances for each road segment. Thus we have a total of 3720*24 = 89280 instances for twenty-four road segments. Since we considered a total of eight intersections (nodes), thus the size of the adjacency matrix is 8x8. We split the collected data into two sets: 80% for training and 20% for testing our model. Table I represent the learning parameter of the Attention-based Spatiotemporal Module.

TABLE I. ATTENTION-BASED SPATIOTEMPORAL MODULE LEARNING PARAMETER

Parameter	Value
Learning Rate	0.01
Number of Epochs	1000
Batch Size	32
Loss Function	RMSE
Optimizer	Adam
Regularization Techniques	L2 regularization

B. Performance Metrics

Three performance indices the mean absolute error (MAE), the mean absolute percentage error (MAPE), and the root mean square error (RMSE) are used to evaluate the effectiveness of the proposed model. This is carried out to assess how accurately the model's predictions were made. The suggested model's efficacy is evaluated using three performance indices: MAE, MAPE, and RMSE. RMSE is frequently used to assess the effectiveness of traffic forecast models [19, 20]. The RMSE gives a general notion of the typical discrepancy between the values of the observed and forecasted data. The model and its predictions are better when the RMSE value is lower. We can calculate the value of RMSE by using equation 12, where f_i is the predicted value and $\hat{f}_i$ is the observed value.

$$RMSE = \left[\frac{1}{n}\sum_{i=1}^{n}(|f_i - \hat{f}_i|)^2\right]^{\frac{1}{2}} \tag{12}$$

A statistical indicator of a forecast system's accuracy is the MAPE. It is easy to understand since it measures as a percentage. MAPE calculation is represented in equation 13, where f_i is the predicted value and $\hat{f}_i$ is the observed value.

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\frac{\left|f_i - \widehat{f}_i\right|}{f_i} \tag{13}$$

A negative number becomes positive through a mathematical operation known as the absolute. As a result, when calculating the MAE, the difference between an expected value and a predicted value is always positive. We can use equation 14 for calculating MAE, where f_i is the predicted value and $\hat{f}_i$ is the observed value.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|fi - \widehat{f}i\right| \tag{14}$$

MAE is calculate using equation (14) where f_i is the actual traffic flow and $\widehat{f}_i$ is the predicted traffic flow. In general, a good prediction model should have lower values of MAE, MAPE, and RMSE, indicating that the predicted values are closer to the actual values. On the other hand, a bad prediction model will have higher values of these metrics, indicating that the model's predictions are further from the actual values. It is essential to consider these metrics when evaluating the effectiveness of a prediction model as they provide insights into the accuracy and reliability of the model's predictions.

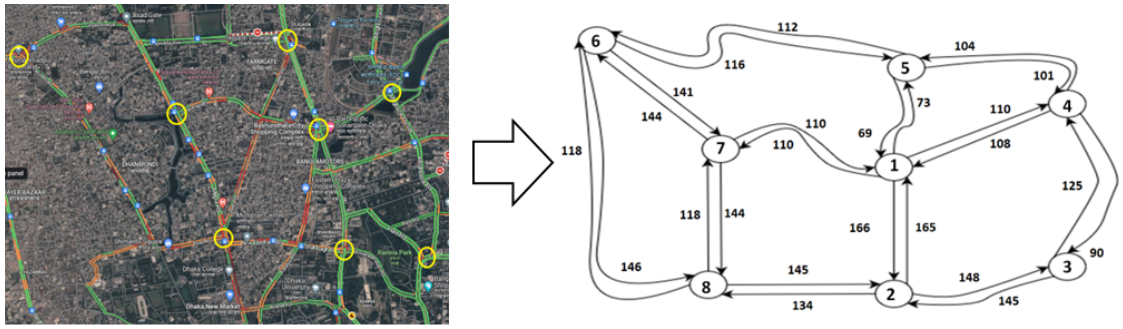
C. Compared Methods

We evaluated our proposed model's performance against the following widely used models for traffic flow prediction. Here we choose five widely used methods from deep learning, and conventional time-series prediction techniques which include ARIMA, SVR, LSTM, GRU, and DNN.

a) ARIMA (Autoregressive Integrated Moving Average): is a conventional statistical method that models the temporal dependence in the data using autoregression, differencing, and moving average techniques. Several research has made use of modeling that is based on ARIMA for the purpose of predicting traffic flow [24, 26, 27]. For instance, an ARIMA model was proposed to develop a short-term time series traffic flow forecast model[26].

b) SVR (Support Vector Regression): makes use of a hyperplane to capture the relationships that exist between the input variables and the output variables. It is efficient in dealing with nonlinear relationships present in the data and has been applied in a number of studies for the purpose of predicting traffic flow [25]. For instance, SVR was applied to the problem of predicting trip times, and it was found to be applicable and to perform well when applied to the study of traffic data [28]. For the purpose of traffic forecasting, LS-SVMs, also known as Least Squares Support Vector Machines, were used. These machines demonstrated benefits such as rapid convergence, high accuracy, and little computational effort [29]. When used in conjunction with other methodologies, SVR can accurately forecast changes in traffic flow as well as accidents. The capability of SVR to handle high-dimensional data, its resistance to noise, and its adaptability to non-linear relationships in the data are just some of its many advantages. SVR, on the other hand, has a number of drawbacks, including the fact that it is very dependent on the kernel function that is used and that it has difficulties managing huge datasets [30].

c) LSTM (Long Short-Term Memory): has been successful at identifying long-term dependencies in sequential data. Due to its capacity for handling input and output of varying lengths, it is frequently employed in time series prediction problems. LSTM is frequently used to estimate traffic flow and has been demonstrated to perform better than other machine learning techniques [23, 31, 32]. An example



(a) (b)

Fig. 3. Road network selection from Google map and graph representation of the selected road network.

TABLE II. PREDICTION COMPARISON FOR OUR PROPOSED ATTENTION-BASED DEEP HYBRID NETWORK WITH BAYESIAN INFERENCE, ARIMA, SVR, LSTM, GRU, AND DNN

	Time	15 min	30 min	45 min	60 min	75 min	90 min	105 min	120 min
Proposed method	RMSE	**15.785**	**16.60**	28.35	17.408	19.90	**14.79**	**17.43**	16.897
	MAE	**7.462**	**7.68**	15.013	**7.03**	7.90	**7.88**	**6.575**	**6.303**
	MAPE	14.055	**13.75**	**13.75**	15.206	17.42	**14.453**	**14.83**	12.912
Proposed method without Bayesian Inference (BI)	RMSE	15.94	16.74	16.74	29.209	17.466	19.87	17.69	17.091
	MAE	7.625	7.725	**7.88**	15.56	7.095	8.055	6.862	6.633
	MAPE	14.329	14.329	14.079	35.90	15.435	17.875	15.467	13.660
DNN [21]	RMSE	20.155	23.066	34.357	17.842	**9.690**	28.944	21.368	19.209
	MAE	15.386	17.514	27.897	15.348	**7.3035**	23.753	16.658	13.864
	MAPE	9.577	13.800	21.683	**9.882**	**6.479**	18.805	13.180	**11.134**
GRU [22]	RMSE	17.856	18.602	21.527	24.216	24.784	19.829	18.064	14.543
	MAE	15.890	12.772	16.792	15.054	17.475	12.981	12.350	13.107
	MAPE	15.168	12.906	**14.016**	15.875	19.340	19.978	18.266	14.427
LSTM [23]	RMSE	16.186	16.828	**13.661**	**16.832**	16.093	15.013	15.446	**14.518**
	MAE	13.041	12.383	11.507	13.776	13.200	12.490	13.341	11.937
	MAPE	15.548	14.277	12.764	10.114	14.697	16.094	13.164	15.249
ARIMA [24]	RMSE	32.295	29.449	29.123	28.751	30.305	27.163	26.830	26.777
	MAE	27.047	24.493	24.204	23.680	25.501	22.335	21.811	22.240
	MAPE	22.800	20.733	20.476	19.962	20.916	18.208	17.824	18.101
SVR [25]	RMSE	17.840	21.008	21.201	22.339	11.991	22.073	18.812	21.183
	MAE	14.708	16.27	17.253	17.861	9.778	17.557	15.147	16.753
	MAPE	**9.303**	13.843	14.281	10.910	8.700	14.237	14.973	13.533

of a Recurrent Neural Network (RNN) that uses memory cells to preserve significant information over time is the LSTM. Long-term memory storage technology (LSTM) is capable of learning long-term dependencies and non-linear traffic flow data. To increase the accuracy of traffic flow prediction [23], LSTM has also been integrated with other techniques, such as multiple linear regression (MLR) [31].

d) GRU (Gated Recurrent Unit): deep learning model, has been effective at identifying dependencies in sequential data. Its capacity to accommodate variable-length inputs and outputs makes it a popular choice for time series prediction jobs. In contrast to the assertion, GRU is frequently employed in traffic flow prediction and has been proven to be successful in doing so [22]. GRU is a variant of recurrent neural networks that work well for predicting traffic flow and can memorize data from the prior sequence [33]. To increase the accuracy of traffic flow prediction, GRU has also been integrated with other approaches like graph convolution networks [34].

e) DNN (Deep Neural Network): is a class of machine learning models that use multiple layers of artificial neurons to make predictions. DNNs have been widely used in various applications, including regression and classification tasks, due to their capacity to learn complex nonlinear correlations in data. Keras is a popular Python library for building DNNs and CNNs (Convolutional Neural Networks) [21]. Keras provides a high-level API for building and training DNNs, making it easy to create and experiment with different architectures [35]. Keras also supports various optimization algorithms, activation functions, and loss functions, making it a versatile tool for building and training DNNs.

D. Performance Evaluation

The simulation result of our proposed traffic flow prediction models is presented in this section. Table II presents the forecasting performance comparison of the proposed model and other baseline methods for different forecasting horizons ranging from 15 minutes to 120 minutes on our collected data set. The findings demonstrate that over the majority of the forecasting horizons, the proposed model performs better than other models in terms of several evaluation measures. Due to its inability to handle complicated spatiotemporal data, the ARIMA model scores the lowest. Because models based on LSTM have the ability to capture dependence over time and nonlinear interactions in the data, LSTM forecasts are sometimes better. In comparison to previous models, the DNN model performs substantially better when predicting traffic flow for 75 minutes, but it performs poorly in all other situations.

Further, we try to measure the statistical significance of our result. We perform a one-way ANOVA test to determine the statistical significance of the differences in the RMSE values among the models. F-statistic is 8.055 and the p-value is 4.496×10^{-6}, which is very small. This indicates that there is a significant difference in the RMSE values among the models. The confidence interval for the proposed method, which is the model of interest, is 15.04 to 21.75. This means that with 95% confidence, the true population means of RMSE values for the proposed method is between 15.04 and 21.75. For measurement of uncertainty we calculate the confidence interval, Table III represents the confidence interval range for different models. Comparing the proposed method's confidence interval to the other models' confidence intervals, it can indeed measure the uncertainty associated with a prediction [36]. A wider confidence level suggests greater uncertainty since it contains a wider spectrum of possible values. A narrower confidence interval, on the other hand, indicates less uncertainty because it gives an estimate that is more accurate [37]. We can see that the proposed method's interval does not overlap with some of the other models, such as ARIMA, SVR, LSTM, the proposed method without BI, and the proposed method. This means that the proposed method has significantly different RMSE values compared to these models. Additionally, the proposed method's confidence interval is narrower than some of the other models, such as DNN and GRU, indicating that the proposed method has less uncertainty in its predictions. Results suggest that the proposed method performs better in handling the uncertainty in traffic flow prediction compared to other models. The narrower confidence interval for the proposed method suggests that it provides more accurate predictions with less uncertainty.

TABLE III. CONFIDENCE INTERVAL OF RMSE OF DIFFERENT MODEL

Model	Start	End	Difference
Proposed method	15.0437	21.746	6.702
Proposed method without BI	14.446	30.333	15.887
DNN	16.050	27.607	11.557
SVR	16.874	27.236	10.362
ARIMA	17.339	30.313	12.974
GRU	17.233	26.621	9.388
LSTM	14.690	23.454	8.764

VI. PERFORMANCE ANALYSIS

Our suggested model takes into account the effect of linked roads on traffic flow forecast. This model captures the influence of neighboring roads on the target road segment by including the road network topology and analyzing the dependencies between road segments. This allows for more precise predictions by accounting for traffic dynamics and flow patterns over the whole road network. It employs Bayesian interference for addressing prediction uncertainty. It can calculate the uncertainty related to its predictions by adding probabilistic modeling. This is especially useful when the prediction outcomes may fall outside of a given range or demonstrate greater variability. This model uses Bayesian inference to assist measure and managing uncertainty, resulting in more trustworthy and robust predictions. It also handles the spatiotemporal patterns and relationships in the traffic flow data like other deep learning models. By giving more importance to relevant spatial and temporal features, this model can make more precise predictions. This model also used an attention mechanism that allows it to focus on the most informative features and road segments.

VII. CONCLUSION

We proposed an attention-based deep hybrid network with Bayesian inference for traffic flow forecasting to address the problem of uncertainty. The attention mechanism in our suggested methodology enhanced the model's ability to recognize unexpected scenarios that impede traffic flow. By collecting complicated spatiotemporal trends in traffic data, our deep hybrid network effectively identified patterns and dependencies, improving the accuracy of traffic flow predictions. Most significantly, by applying Bayesian inference, we successfully evaluated and minimized uncertainty in the projected outcomes. In future, the suggested method's transferability and scalability will be evaluated by testing it in various parts of cities or regions with diverse traffic patterns and features. To prove its generalizability, it would be beneficial to assess its performance in other metropolitan areas also.

ACKNOWLEDGMENT

This work is also partially supported by a grant for the Research Fellowship (2022-2023) funded by the Information and Communication Technology Division, Ministry of Telecommunications and Information Technology, Government of Bangladesh. ICT Fellowship No 56.00.0000.052.33.004.22-15.

REFERENCES

[1] P. Releases. (2020) Inrix scorecard 2020: After a year of lockdowns, uk city centre congestion down 52%. [Online]. Available: https://inrix.com/press-releases/2020-traffic-scorecard-uk/

[2] L. Aratani. (2020) Sitting in traffic costs d.c.-area residents an average of $1,761 per year, study finds. [Online]. Available: https://www.washingtonpost.com/transportation/2020/03/09/sitting-traffic-costs-dc-area-residents-an-average-1761-per-year-study-finds/

[3] M. Z. Haider and R. S. Papri, "Cost of traffic congestion in dhaka metropolitan city," *Public Transport*, vol. 13, no. 2, pp. 287–299, 2021.

[4] Y. Wu, H. Tan, L. Qin, B. Ran, and Z. Jiang, "A hybrid deep learning based traffic flow prediction method and its understanding," *Transportation Research Part C: Emerging Technologies*, vol. 90, pp. 166–180, 2018.

[5] S. Du, T. Li, X. Gong, and S.-J. Horng, "A hybrid method for traffic flow forecasting using multimodal deep learning," *arXiv preprint arXiv:1803.02099*, 2018.

[6] W. Li, Y. Ji, and T. Wang, "Adaptive real-time prediction model for short-term traffic flow uncertainty," *Journal of Transportation Engineering, Part A: Systems*, vol. 146, no. 8, p. 04020075, 2020.

[7] Y. Li, S. Chai, G. Wang, X. Zhang, and J. Qiu, "Quantifying the uncertainty in long-term traffic prediction based on pi-convlstm network," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 11, pp. 20 429–20 441, 2022.

[8] Y. Wu and J. James, "A bayesian learning network for traffic speed forecasting with uncertainty quantification," in *2021 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2021, pp. 1–7.

[9] G. Liu, H. Jin, J. Li, X. Hu, and J. Li, "A bayesian deep learning method for freeway incident detection with uncertainty quantification," *Accident Analysis & Prevention*, vol. 176, p. 106796, 2022.

[10] M. Seger and L. Kisgyörgy, "Predicting and visualizing the uncertainty propagations in traffic assignments model using monte carlo simulation method." *Journal of advanced transportation*, 2018.

[11] L. Yang, "Uncertainty prediction method for traffic flow based on k-nearest neighbor algorithm," *Journal of intelligent & fuzzy systems*, vol. 39, no. 2, pp. 1489–1499, 2020.

[12] J. Fu, W. Zhou, and Z. Chen, "Bayesian spatio-temporal graph convolutional network for traffic forecasting," *arXiv preprint arXiv:2010.07498*, 2020.

[13] N. Adaloglou. (2020) How attention works in deep learning: understanding the attention mechanism in sequence models. [Online]. Available: https://theaisummer.com/attention/

[14] G. E. Box and G. C. Tiao, *Bayesian inference in statistical analysis*. John Wiley & Sons, 2011.

[15] I. Hossain and N. Nower, "Traffic data collection and visualization tool for knowledge discovery using google maps," *International Journal of Software Innovation (IJSI)*, vol. 10, no. 1, pp. 1–12, 2022. [Online]. Available: https://doi.org/10.4018/IJSI.293270

[16] N. N. Md. Moshiur Rahman, "Attention-based deep hybrid networks for traffic flow prediction using google maps data," in *8th International Conference on Machine Learning Technologies (ICMLT 2023)*. ACM, 2023. [Online]. Available: https://doi.org/10.1145/3589883.3589894

[17] M. M. Rahman, "trafficdatacollectiontool," https://github.com/Moshiurcse13/trafficDataCollectionTool, 2022.

[18] Z. Abbas, A. Al-Shishtawy, S. Girdzijauskas, and V. Vlassov, "Short-term traffic prediction using long short-term memory neural networks," in *2018 IEEE International Congress on Big Data (BigData Congress)*. IEEE, 2018, pp. 57–65.

[19] Y. Rong, X. Zhang, X. Feng, T.-k. Ho, W. Wei, and D. Xu, "Comparative analysis for traffic flow forecasting models with real-life data in beijing," *Advances in mechanical engineering*, vol. 7, no. 12, p. 1687814015620324, 2015.

[20] L. Chen, Q. Ren, J. Zeng, F. Zou, S. Luo, J. Tian, and Y. Xing, "Csfpre: Expressway key sections based on ceemdan-stsgcn-fcm during the holidays for traffic flow prediction," *Plos one*, vol. 18, no. 4, p. e0283898, 2023.

[21] J. Brownlee. (June 18, 2022) Your first deep learning project in python with keras step-by-step. [Online]. Available: https://machinelearningmastery.com/tutorial-first-neural-network-python-keras/

[22] B. Hussain, M. K. Afzal, S. Ahmad, and A. M. Mostafa, "Intelligent traffic flow prediction using optimized gru model," *IEEE Access*, vol. 9, pp. 100 736–100 746, 2021.

[23] P. Poonia and V. Jain, "Short-term traffic flow prediction: using lstm," in *2020 International Conference on Emerging Trends in Communication, Control and Computing (ICONC3)*. IEEE, 2020, pp. 1–4.

[24] T. Alghamdi, K. Elgazzar, M. Bayoumi, T. Sharaf, and S. Shah, "Forecasting traffic congestion using arima modeling," in *2019 15th international wireless communications & mobile computing conference (IWCMC)*. IEEE, 2019, pp. 1227–1232.

[25] G. Lin, A. Lin, and D. Gu, "Using support vector regression and k-nearest neighbors for short-term traffic flow prediction based on maximal information coefficient," *Information Sciences*, vol. 608, pp. 517–531, 2022.

[26] X. Lin and Y. Huang, "Short-term high-speed traffic flow prediction based on arima-garch-m model," *Wireless Personal Communications*, vol. 117, no. 4, pp. 3421–3430, 2021.

[27] 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, pp. 1–9, 2015.

[28] C.-H. Wu, J.-M. Ho, and D.-T. Lee, "Travel-time prediction with support vector regression," *IEEE transactions on intelligent transportation systems*, vol. 5, no. 4, pp. 276–281, 2004.

[29] Y. Zhang and Y. Liu, "Traffic forecasting using least squares support vector machines," *Transportmetrica*, vol. 5, no. 3, pp. 193–213, 2009.

[30] T. D. Toan and V.-H. Truong, "Support vector machine for short-term traffic flow prediction and improvement of its model training using nearest neighbor approach," *Transportation research record*, vol. 2675, no. 4, pp. 362–373, 2021.

[31] R. Shi and L. Du, "Multi-section traffic flow prediction based on mlr-lstm neural network," *Sensors*, vol. 22, no. 19, p. 7517, 2022.

[32] Y. Tian and L. Pan, "Predicting short-term traffic flow by long short-term memory recurrent neural network," in *2015 IEEE international conference on smart city/SocialCom/SustainCom (SmartCity)*. IEEE, 2015, pp. 153–158.

[33] P. Sun, A. Boukerche, and Y. Tao, "Ssgru: A novel hybrid stacked gru-based traffic volume prediction approach in a road network," *Computer Communications*, vol. 160, pp. 502–511, 2020.

[34] R. Fu, Z. Zhang, and L. Li, "Using lstm and gru neural network methods for traffic flow prediction," in *2016 31st Youth Academic Annual Conference of Chinese Association of Automation (YAC)*. IEEE, 2016, pp. 324–328.

[35] S. Tanwar. (June 26, 2019) Building our first neural network in keras. [Online]. Available: https://towardsdatascience.com/building-our-first-neural-network-in-keras-bdc8abbc17f5

[36] A. Khosravi, S. Nahavandi, and D. Creighton, "Prediction intervals for short-term wind farm power generation forecasts," *IEEE Transactions on sustainable energy*, vol. 4, no. 3, pp. 602–610, 2013.

[37] T. J. Sullivan, *Introduction to uncertainty quantification*. Springer, 2015, vol. 63.