Short-Term Bus Passenger Flow Prediction Using Multi-Component Graph Attention Nerual Network Model

Mohamed Chaaben
Engineering Student at the Higher School of Communication of Tunis
Email: mohamed.chaaben@supcom.tn

Asiye Baghbani, Corresponding Author
PhD Student, Concordia Institute for Information Systems Engineering
Computer Science and Visual Arts Integrated Complex, 1515 St. Catherine W., Montreal, Canada
Artificial Intelligence PhD Researcher at Buspas Inc., Montreal, Canada
Email: asiye.baghbani@mail.concordia.ca

Nizar Bouguila
Professor, Concordia Institute for Information Systems Engineering
Computer Science and Visual Arts Integrated Complex, 1515 St. Catherine W., Montreal, Canada
Tel: 514-48-2424 ext. 5663, Email: nizar.bouguila@concordia.ca

Zachary Patterson
Associate Professor, Concordia Institute for Information Systems Engineering
Computer Science and Visual Arts Integrated Complex, 1515 St. Catherine W., Montreal, Canada
Tel: 514-848-2424 ext. 8830, Email: zachary.patterson@concordia.ca

Word Count: 6749 words + 1 table(s) × 250 = 6999 words

Submission Date: August 1, 2022

ABSTRACT

Predicting passenger flow is an important part of transit operation planning. It is nonetheless subject to a variety of influencing factors. Emerging deep learning models are increasingly being used to improve passenger flow prediction precision. Nevertheless, most of these methods predict the future passenger flow of a stop based on its historic passenger flow alone, without taking its neighbors into account, or, if they do, they treat their spatial-temporal dependencies equally, which is not valid in real life. In this vein, we propose a graph deep learning model entitled Multi-Component Spatial Temporal Graph Attention Convolution (Mutli-STGAC) in order to predict short-term bus passenger flow. The proposed framework consists of a Graph Attention Network, as well as a Graph Convolution Neural Network that are specifically designed to account for spatial correlations in passenger flow data in the context of a bus network. In order to capture long sequences of historical time intervals, a gated dilated convolution layer was also used in the temporal modeling process part of the Multi-STGAC model. Furthermore, the model contains a layer of fusion in which short-, mid-, and long-term temporal dependencies are taken into account simultaneously. Moreover, with real-world data from the Laval, Canada bus network, an analysis of the proposed model is performed to compare its MAE, RMSE, and accuracy with other deep learning models. We find that the proposed model outperforms the LSTM, ConvLSTM, and different versions of Multi-STGAC that do not take into account one of its assumptions.

Keywords: Short-Term Passenger Flow Forecasting, Deep learning, Graph Neural Network, Bus Network, Graph Attention

INTRODUCTION

As the world's urban population grows, the demand for transportation is increasing. This has concomitantly led to many problems, such as traffic congestion, atmospheric pollution, energy consumption, etc. Public transportation is a critical aspect of well functioning transportation systems and can contribute to the livability of cities, less congestion, faster commutes, more convenience, and a reduction in carbon emissions. Public bus transportation is an essential part of public transportation systems more broadly. Numerous studies demonstrate that encouraging people to prioritize taking the bus is an effective strategy to cope with the latter issues (*1*). As part of Intelligent Transportation Systems (ITS), the prediction of bus passenger flows is a major component that can help with the better management of bus systems. Additionally, it facilitates the allocation of resources, the management of infrastructure, and the determination of service frequencies. It remains challenging, however, to accurately describe and capture the high variability and nonlinearity that characterizes bus passenger flows.

Previous approaches to passenger flow forecasting can be categorized into short-term and long-term prediction. Long-term prediction studies typically have a forecast horizon of one to several days. These forecasts are generally operator-based as they provide strategic level information for planning and are thus fundamental for planning and network development purposes (*2*). In contrast to long-term prediction studies, short-term projections are based on forecast horizons from the next few minutes to the next few hours ahead. They are usually client-based meaning they are often used for consumer applications (like mobile application) or bus stop board notifications. They also have a vital role in the real-time administration and monitoring of a bus transportation system. Short-term prediction approaches have recently received increasing attention from transport planners and researchers due to their important role (*3, 4*) making them a prerequisite for proactive operations and management of bus transport services (*5*).

There have been a variety of approaches used to predict short-term passenger flows, including linear or nonlinear methods, such as deep learning models. However, many methods focus only on the historical passenger flows of a target stop when predicting its future value, not on network-wide spatio-temporal dependencies. Graph family neural networks are becoming increasingly popular in the field of transportation (*6*). Graph Neural Networks (GNNs) capture network-wide spatio-temporal dependencies by transmitting messages between nodes. The bus transportation network can be viewed as a graph, since it is composed of nodes and links. Although these methods can predict passenger flow to different degrees, they often treat neighbors' spatio-temporal dependencies equally, resulting in a failure to select the most important features for further accuracy improvement. Moreover, most studies on passenger flow forecasting have focused on metro systems, and even when they were applied to bus systems, the system size was not adequate to project a large-scale network comparable to metropolises.

To address these issues, we develop a multi-component spatial-temporal graph attention convolution (Multi-STGAC) model for forecasting short-term passenger flow in the bus network as part of this study. Three specific contributions are made by this study: 1) Combining Graph Attention Networks with graph convolution networks, which have been specifically developed for bus networks, for capturing spatial features; the model takes into account how the neighborhood and traffic affect the predicted passenger flow value. 2) A Temporal Gated Dilated Convolution Layer is designed specifically to capture the long-term time series sequences as well as a fusion layer to consider short-term, mid-term, and long-term historical temporal dependencies. 3) It evaluates the performance of the suggested framework on a real network with 467 stops in Laval, Canada.

To contextualize the subject of this study, we begin by reviewing the literature on short-term prediction of passenger flows in public transportation systems. Then, we present the methodology, which involves a description of the model's architecture and the necessary context for comprehending the proposed model. After describing the dataset, we exhibit the results alongside the performance of other reference models. Finally, we conclude with a discussion of potential future research topics.

LITERATURE REVIEW

Short-term passenger flow has been predicted using a variety of approaches in different studies. In the beginning, several attempts have been made to predict passenger flow using linear models (also known as parametric models or statistical models). For instance, Zhang et al. utilized a basic Kalman filter algorithm in an effort to predict short-term bus passenger flow at bus stops (*7*). Yang et al. also predicted the short-term passenger flow using a generalized linear regression model (*8*). Gu et al. used an autoregressive moving average model (ARMA) to forecast the passenger flow at bus stops of Shanghai in China (*9*). Xue et al. proposed an interactive multiple model (IMM) that implements different methods to forecast passenger demand during the day. They applied ARMA, autoregressive integrated moving average (ARIMA), and seasonal ARIMA for weekly, hourly and daily time-series analyses, respectively (*10*).

Most studies using linear approaches for forecasting passenger flow used simple networks and small datasets. Real-world passenger flow observations are complex and highly variable, making linear approaches less applicable for modeling passenger flow variation accurately. Consequently, machine learning and deep learning models are becoming increasingly popular in passenger flow forecasting studies. For instance, Yang and Liu has developed a Support Vector Regression (SVR) based on Affinity Propagation for bus stop passenger flow prediction (*11*). Chen et al. also established a short-term passenger flow prediction model based on Least Squares Support Vector Machine Regression (LS-SVMR) (*12*). Likewise, Lv et al. proposed a passenger flow prediction model based on bus Integrated Circuit (IC) card data. They collected points-of-interest (POI) data around bus stops, and an extreme gradient boosting (XGBoost) approach was implemented to train the model for each bus line (*13*).

Since deep learning models can better account for nonlinearity and complexity observed in transit data, these models have achieved state-of-the-art results in the passenger flow prediction domain. A variety of deep learning models have been used to predict passenger flow. One of the most commonly used deep learning methods to predict passenger flows is Convolutional Neural Networks (CNN). For instance, Shen et al. used CNN for building a metro passenger flow forecasting system (*14*). In another study, Zhang et al. used a CNN-based framework for short-term forecasting of passenger demand on a multi-zone basis (*15*). The Recurrent Neural Network (RNN) model is another type of deep learning model, mostly used for exploring temporal dependencies in passenger flow prediction (*16, 17*). Long short-term memory (LSTM) model is one of the popular approaches in RNN which has been used a lot in the concept of passenger flow forecasting in metro networks (*18*) and in bus networks (*19, 20*).

Neither CNNs nor RNNs are capable of capturing both spatial and temporal features at the same time. Consequently, in recent years, hybrid traffic prediction models have become increasingly popular because they incorporate both spatial and temporal interdependencies. For example, in a recent paper, Wang et al. have combined LSTMs and CNNs to analyze spatial and temporal data simultaneously for predicting passenger flows in Shanghai, China's subway network (*21*).

Xiao et al. also predicted passenger flow and travel times using LSTM, CNN-LSTM, and ConvLSTM (*22*). In another study, using both convolutional neural networks (CNNs) and gated recurrent units (GRUs), Zaho et al. proposed a method for classifying bus passenger flows. Taking into account both working days and non-working days scenarios, the model predicted the classified passenger flows of routes and cross-sections of a bus line (*23*).

Despite the combination of CNNs and RNNs solving the problem of simultaneously extracting and fitting spatio-temporal features, these techniques are not sufficient for modeling transit networks, which are non-Euclidean in nature. Research on machine learning tasks involving graphs has recently caught the attention of researchers, and a new class of neural networks known as Graph Neural Networks (GNNs) was developed and adopted in a variety of applications. Comparatively to CNNs that are limited to Euclidean data structures, GNNs are able to capture spatial information concealed in non-Euclidean structured data (*24*). This type of information is commonly encountered in transportation applications.

The use of GNNs opens up new avenues for addressing passenger flow forecasting challenges, and several studies have used this model in this area. For example, Kong et al. proposed a framework for the prediction of passenger flow based on graph convolutional networks (GCNs). In this study, a novel sharing-stop network was constructed based on bus records in order to model the relationship between passengers (*25*). Similarly, Han et al. proposed a GCN-based model to forecast the inflow and outflow passenger flow volumes at each metro station in Shanghai, China (*26*). In another study, Arabghalizi and Jia forecasted the short-term flow of passengers at bus stops using a combination of GCN, CNN, and LSTM for extracting spatial and temporal data patterns. They analyzed the proposed method based on historical data collected from 87 bus stations along one of Pittsburgh's busiest routes (*27*). Ma et al. also proposed a model for estimating short-term subway passenger flow based on GCN and Bidirectional LSTM (BiLSTM) (*28*).

Public transportation networks in the real world are dynamic, unlike graph neural networks where each node's effects on its neighbors are static and the same. The impact of two stops on each other may vary, for example, based on their proximity or distance and whether they are connected or not. In general, these impacts are highly dynamic and change over time. In an increasing number of studies, attention mechanisms are used in conjunction with GNNs to overcome this limitation. As a result of an attention mechanism, deep neural network models are able to make predictions based on the most relevant parts of the data, from a spatial or temporal perspective. It can be achieved by dynamically assigning weights to edges/nodes in the graph or to different time intervals, which has been explored in several studies (*29*). Han et al., for example, presented a unique deep learning architecture that integrates graph attention networks (GAT) with LSTM to predict passenger flow. In this work, GATs learned spatial dependency, while LSTM learned temporal dependency (*30*). In another study, Li et al. also combined LSTM, GCN, and attention mechanism to have a short-term metro passenger flow prediction considering only adjacent stations (*31*). Chen et al. also employed a GAT network to capture the spatial correlation of multiple bus stops and a BiLSTM with attention to extract the temporal correlation of historical ridership, in order to forecast passenger flow (*32*).

The GAT model has been used in several studies to predict passenger flow, but further studies are needed for a number of reasons. First, it has mainly been used in the studies with the application of metro systems, and even when it has been used for bus systems applications, the systems have been very limited in size and complexity which is not sufficient to take advantage of complex graph-based approaches. Furthermore, the attention mechanism described in the passenger flow predictions so far is not able to account for the interdependencies caused by traffic and other factors, which clearly have a significant impact on it. As this is still a relatively new model, combining it with another deep learning model would allow it to accommodate more spatial and temporal variables, making it more like a public transportation system in the real world. Therefore, in this study, an end-to-end graph attention framework is being developed called Multi-Component Spatial Temporal Graph Attention Convolution (Multi-STGAC) to predict passenger flow on a bus network in Laval, Canada. For spatial modeling, this framework implements a GAT and a specific GCN tailored specifically to the bus network. In addition, for the temporal modeling, a gated dilated convolution layer was used, which is capable of capturing long sequences of time intervals. Furthermore, the model also contains a fusion layer which allows it to take into account short-term, mid-term, and long-term temporal dependencies simultaneously.

METHODOLOGY

Figure 1 illustrates the structure of the proposed Multi-STGAC framework in this study, consisting of three independent components that share the same structure for extracting spatio-temporal features over different time horizons. Specifically, they are intended to determine the dependencies of historical data on recent, daily, and weekly periods. In the beginning of each component, there is a convolution layer, which is responsible for generating various feature maps and capturing correlations between the input features. This convolution layer provides input to the next four spatio-temporal Temporal Spatial Modeling (TSM) modules, which are stacked on top of one another. Each TSM module is containing two layers of what we call Temporal Gated Dilated Convolution Layers (Temporal-GDCN) and Spatial Graph Attention Convolution Layers (Spatial-GACN), follow by a Batch Normalization (BN) layer. As a final step, the outputs from each of the three components, Y_R, Y_D, and Y_W, are combined via a fusion layer to produce the final prediction. The details of each layer will be discussed in the following paragraphs.

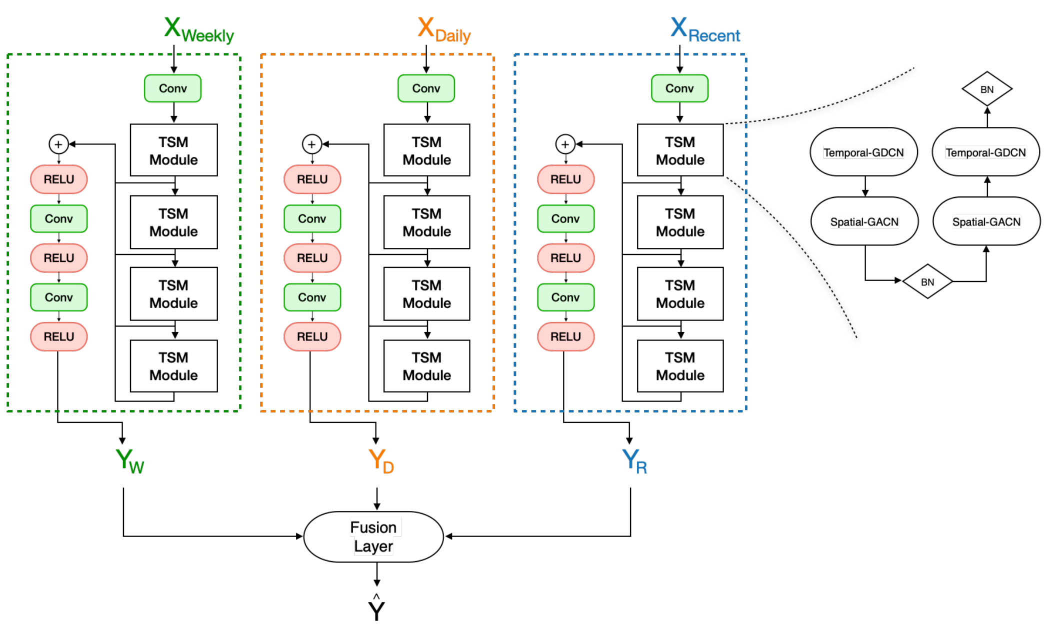

Figure 1 The Structure of the Multi-Component Spatial Temporal Graph Attention Convolution (Multi-STGAC) Model

Model Input Definition

In this section, the different inputs for the model are described. We describe first how to generate the different historical passenger flow datasets that will be used as inputs for each time series segment of the model, means X_{Recent}, X_{Daily}, and X_{Weekly}. Next, the bus network graph will be described along with its adjacency matrix and bus proximity matrix.

Historical Passenger Flow Time Series Inputs

Suppose that passenger flow data is aggregated every $\Delta t_{interval}$ minutes and a sample frequency for a day is p times mean $\Delta t_{interval} \times p = 1440$ minutes, with the index for the prediction time interval being t. As also shown in **Figure 1**, three time series inputs are considered for the recent, daily, and weekly spatio-temporal components in the framework whose lengths are determined by T_r, T_d, and T_w respectively. The following paragraphs provide details regarding each of these input data:

- **The recent segment:** Bus stop passenger flow gradually extends to its neighboring stops over time, so the near-neighboring historical time series to the predicted time interval can have a significant effect on how much predicted passenger flow is observed at that particular stop and its neighboring stops. As a result, a segment adjacent to the interval of prediction has been defined for this time series. This time series data can be defined by $X_{Recent} = [X_{t-T_r+1}, X_{t-T_r+2}, \dots, X_{t-1}]$.
- **The daily-periodic segment:** This segment consists of time intervals that occurred during the past several days at the same time of day as the prediction time interval. Due to the fact that people's daily habits, such as morning and evening peaks, are almost always the same throughout the workday, traffic and passenger flow statistics may display recurring patterns. The objective of this component is to create a model that illustrates the daily periodicity of passenger flow data and it is representing by $X_{Daily} = [X_{t-T_d \times p}, X_{t-(T_d-1)\times p}, \dots, X_{t-p}]$.
- **The weekly-periodic segment:** This segment consists of the same periods that occurred over the last few weeks at the same time and day as the prediction interval. The traffic patterns of a particular day are typically similar to those of the previous week. The purpose of the weekly-periodic component is to record the weekly patterns that occur frequently in the passenger flow data, and it is representing by $X_{Weekly} = [X_{t-T_w \times 7 \times p}, X_{t-(T_w-1)\times 7 \times p}, \dots, X_{t-7\times p}]$

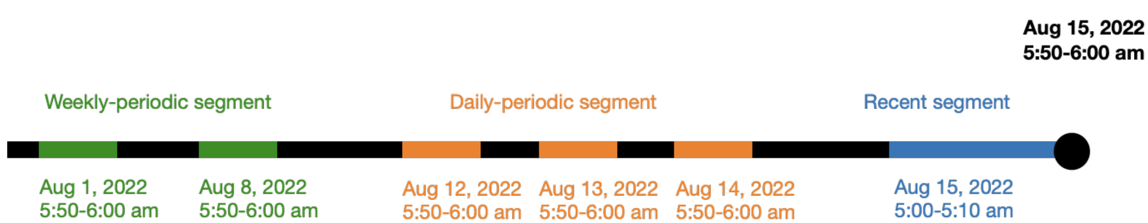

Figure 2 Time series dataset segments for weekly, daily, and recent input data examples

Bus Network Graph and Adjacency Matrix

In general, a graph is defined as $G = (V, E)$ where V is the set of vertices (or nodes) whose cardinality is noted by N and E is the set of edges between the nodes. Practically, we consider stops as nodes, and bus routes between stops are defined as the edges of the graph. The adjacency matrix is seen as the key to capturing spatial dependency in traffic forecasting (*33*). It is built by assigning to each couple of nodes (bus stops) the value one if they are connected, otherwise zero.

Bus Network Proximity Matrix

The effect of each stop on each other are different in real-world bus networks. For example, most of the time, the impact of physically close stops is typically greater than that of stops that are physically farther apart. A Bus Network Proximity Matrix (BNP) has been created to account for this property of the bus network. In this matrix, cells reflect the possibility that a given destination node can be reached from a given origin node in a specified amount of time. As a prerequisite to defining the BNP matrix, we must first define the distance matrix. As a function of the length of each road in the network, a distance matrix $Dist \in R^{N\times N}$ is defined, in which each element $Dist_{ij}$ represents the real distance between nodes i and j ($Dist_{ii} = 0$). We then define the Bus Network Proximity Matrix in **Equation 1** based on the distance matrix defined above and the average bus speed in each edge:

$$BNP_{ij} = \begin{cases} 1, & \text{if } S_{ij}\Delta t - Dist_{ij} \geq 0 \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

In this equation, S_{ij} is the average speed of the bus between the stops i and j, and Δt is the time interval. Each element BNP_{ij} equals one if the passenger using the bus can reach from stop i to stop j under a specific time interval Δt , and $BNP_{ij} = 0$ otherwise. Also, each stop is considered self-reachable and hence all diagonal values in BNP are equal to one.

Temporal-Spatial Modeling Module

In the Temporal-Spatial Modeling (TSM) module, temporal-spatial dependencies are captured using two different deep learning algorithms layer: 1) Temporal Gated Dilated Convolution Layer (Temporal-GDCN) and 2) Spatial Graph Attention Convolution Layer (Spatial-GACN). Each TSM module is composed of two Temporal-GDCN layers and two Spatial-GACN layers. Here are the details of the various layers used in this module.

Temporal Gated Dilated Convolution Layer

Figure 3 illustrates how dilated causal convolution slides over inputs by skipping values with a certain step, as a special case of standard 1D convolution. As has been shown, in dilated causal convolution networks, layer depth can be increased exponentially (*34*). This is their advantage over RNNs, as this allows them to handle long-range sequences correctly in a non-recursive manner, facilitating parallel computations and alleviating gradient explosion. By padding the inputs with zeros, the dilated causal convolution preserves the temporal causal order by making predictions on the current time step only involving historical information taken from previous time steps (*34*).

Gate mechanisms are introduced to deal with the vanishing gradient problem and have proven essential to RNN success (*35*). This mechanism smooths out the applied layer's output. Accordingly, a Gated Convolution refers to a convolution layer that uses a gating mechanism. It has been demonstrated that they can effectively control the flow of information through temporal convolution networks (*36*). In order to learn complex temporal dependencies, we employ a Temporal Gated Dilated Convolution (Temporal-GDCN) layer as part of the TSM module. This paper uses a simple Temporal-GDCN contains only an output gate as shown in **Equation 2** (*34*).

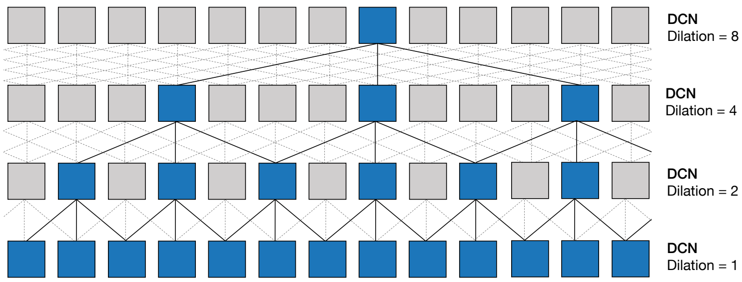

Figure 3 An example of dilated casual convolution with kernel size 3

$$h = g(\Theta_1 * X + a) \odot \sigma(\Theta_2 * X + b) \tag{2}$$

In this equation, X is the input data and Θ_1, Θ_2, a, and b are the model parameters. $\odot$ is also the elementwise product. Moreover, in this layer, $g(\cdot)$ represents the activation function, which was considered a hyperbolic tangent function, and $\sigma(\cdot)$ is the sigmoid function that determines how much information passes on to the next layer.

Spatial Graph Attention Convolution Layer

Attention mechanisms have previously been proven to be effective for tasks such as machine reading (*37*) and learning sentence representations, which use RNNs or convolutional neural networks (*38*). By adding an attention layer to the neural network, the most important parts of the data are given more attention instead of the entire data. This increases the dependability of the neural network while simultaneously focusing on the most important information. As part of the Spatial Graph Attention Convolution Layer (Spatial-GACN) of TSM module, the graph attention layer is juxtaposed just before the graph convolution layer to learn spatial dependencies and aggregate information from nodes by assigning different importance. To make the graph attention layer, the nodes are first transformed using a shared linear transformation (**Equation 3**) parameterized by a weight matrix W.

$$e_{ij} = a(W\vec{h}_i, W\vec{h}_j) \tag{3}$$

The attention is then applied to the nodes by using the function α. This is a single-layer feedforward neural network applied with the LeakyReLU activation function that has 0.2 as negative input slope. To make coefficients easier to compare across nodes, we normalize them using the $softmax$ function, as given in **Equation 4**.

$$\alpha_{ij} = softmax(e_{ij}) = \frac{exp(e_{ij})}{\sum_{k \in \mathcal{N}_i} exp(e_{ik})} \tag{4}$$

Here $\mathcal{N}_i$ is the first neighborhood of node i. Once obtained, the normalized attention coefficients are used to compute a linear combination of the features corresponding to them, to serve as the final output features for every node after applying an ELU activation function (*39*). The final value for each node is represented by **Equation 5**.

$$\vec{h}_i' = ELU(\sum_{j \in \mathcal{N}_i} \alpha_{ij} W \vec{h}_j) \quad (5)$$

After calculation of the output of attention mechanism, it will be used as an input for the graph convolution layer. The primary purpose of a graph convolution layer is to derive locally relevant features from the input data that is organized in the form of a graph. A graph convolution operation for the purpose of extracting spatial features is shown in **Equation 6**.

$$GC_t = (W_{GC} \odot A \odot BNP)\, h_t' \quad (6)$$

where $\odot$ is the Hadamard product operator, and $h_t' \in R^N$ is the vector of passenger flow states of all nodes at time t. The $W_{GC} \in \mathbb{R}^{N \times N}$ is a trainable weight matrix and the $GC \in R^N$ is the extracted spatial graph convolution feature. As A and BNP are both sparse matrices only containing 0 and 1 elements, the result of the convolution is also sparse. In addition, the trained weight W_{GC} has the potential to assess the interaction effect between graph nodes, which would ultimately improve the model's interpretability.

Fusion Layer

As has been shown in **Figure 1**, each of these three recent, daily, and weekly components has been formed with the four TSM module which is repeated consecutively to extract deeper features and make the model more robust. Let $\hat{Y}_R, \hat{Y}_D$ and $\hat{Y}_W$ be the output of these three components. The fusion is done by concatenating them along the feature axis. Then, in order to learn the correlations between the three components as well as the properties of each prediction time step, we use two convolution layers with the ELU activation function as is shown in **Equation 7**,

$$\hat{Y} = W_1 \otimes ELU(W_2 \otimes [\hat{Y}_R \,||\, \hat{Y}_D \,||\, \hat{Y}_W]) \quad (7)$$

In this equation $||$ means concatenation operation and $\otimes$ is the convolution operation. In this study, we use Mean Squared Error (MSE) as a loss function, since it is commonly used for predicting continuous values.

EXPERIMENTS

This section describes the real-world experiment in which the proposed framework is applied. After presenting the characteristics of the dataset, we present the outcomes of our proposed model, as well as the outcomes of other baselines, in order to compare the performance of these models in predicting passenger flows.

Experimental data

The proposed model was applied to a real-world dataset that contains information about passengers of the Société de Transport de Laval (STL), the transportation provider in Laval, Canada. The importance of this dataset stems from the fact that Laval is a rapidly growing city of more than 400,000 residents in the region of Greater Montreal. With the aim of having an order of magnitude in relation to real-world scenarios, we test our model on a relatively complex subnetwork of the Laval bus transportation network, which includes the Route 26, one of the busiest routes in Laval, as well as its ten feeders. **Figure 4** gives an overview of the network modeled in this study.

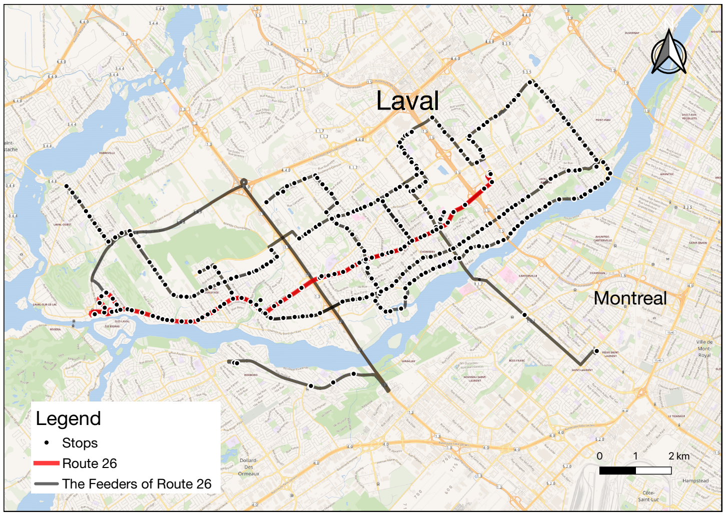

Figure 4 The network of Dataset

Figure 5 provides a more in-depth look at the data by providing an order of magnitude for the number of stops and the number of passengers boarded on each bus route. The X-axis denotes the bus routes in our network. For each route, the orange bar is graduated on the right side of the chart and describes the number of boarded passengers in the whole dataset, whereas the blue bar is graduated on the left chart side and indicates the number of stops on each route. The dataset includes data from the middle of September to the middle of October 2021. It is derived from 11 bus routes with a total of 467 bus stops, and 121,143 boarded passengers.

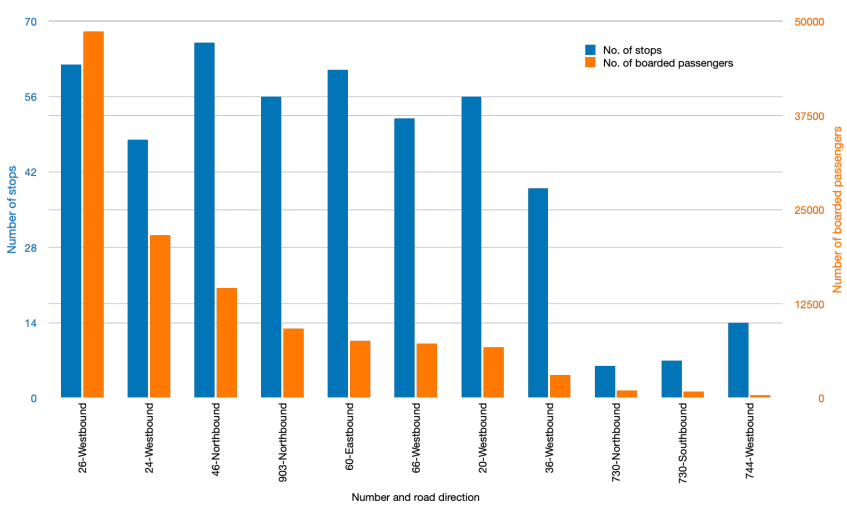

Figure 5 Number of stops and number of boarded passenger per route

The Bus Network Proximity matrix is constructed using the true distance between bus stops for making the distance matrix, and an average bus speed of 18 kilometers per hour according to the Société de Transport de Montréal (*40*). The Min-Max normalization is performed to scale the data into the range of [0, 1] and then we use a sliding window approach to sample the scaled data and reshape them into the required form.

Evaluation

To predict passenger flow at each time step, we used ten previous intervals for the recent segment, four previous intervals for the daily-periodic segment, and two previous intervals for the weekly-periodic segments, respectively. All intervals are equally set to 10 minutes. The performance of the proposed model and the other deep learning models in this study are evaluated by three commonly used metrics: Mean Absolute Error (MAE), Root Mean Squared Error (RMSE) and Percentage Accuracy whose formulae are shown in **Equation 8**, **Equation 9** and **Equation 10** respectively.

$$\mathrm{MAE} = \frac{1}{n}\sum_{k=1}^{n}\left|Y_k - \hat{Y}_k\right| \tag{8}$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{k=1}^{n}\left(Y_k - \hat{Y}_k\right)^2} \tag{9}$$

$$\mathrm{Accuracy} = \frac{Number\ of\ correct\ predictions}{n} \times 100 \tag{10}$$

where n is the total number of observations, Y is the real input data and $\hat{Y}$ is the corresponding predicted data. The model is trained by minimizing the mean squared error and the initial learning rate of 10^{-5}. The Adam algorithm is used as the gradient descent optimizer as it combines the advantages of two other extensions of stochastic gradient descent: Adaptive Gradient Algorithm (AdaGrad) which adaptively scales the learning rate parameter for each dimension to ensure that the training process is neither too slow nor too imprecise (*41*) and Root Mean Square Propagation (RMSProp) which overcomes gradient exploding and vanishing problems (*42*). In the training process, we use 90% of the data, and we choose to perform an early stopping mechanism, resulting in a different number of training epochs for each model.

Experimental Results

To validate the performance of the proposed model, we compare its prediction accuracy with three different models under the same experimental conditions. As can be seen in **Table 1**, the proposed model overwhelmingly outperforms baseline models with regard to MAE and RMSE metrics, which are respectively 0.216 and 0.903. These values correspond to a Percentage Accuracy of 95.7%. This is most likely related to the fact that the model takes into account both temporal and spatial features, the topology of the network, as well as the three types of historical data (recent, daily-periodic and weekly-periodic data), while the other models omit at least one sort of these crucially significant characteristics.

Table 1 Performance comparison with various models

Model	MAE	RMSE	Accuracy	4 bins Accuracy
LSTM	0.485	1.011	75.1%	89.7%
ConvLSTM	0.383	0.883	79.4%	91.2%
Recent STGAC	0.317	1.001	90.5%	97%
Mutli-STGAC without Attention Layer	0.288	0.913	89.2%	96.3%
Mutli-STGAC without BNP matrix	0.312	0.905	87.4%	95.6%
Mutli-STGAC	**0.216**	**0.903**	**95.7%**	**99.3%**

Since the LSTM continues to be the pioneer approach in time series prediction, we compare our model to it. The results demonstrate that the Multi-STGAC model performs better in terms of the evaluation metrics. The LSTM model gets a MAE of 0.485, a RMSE of 1.011 and an accuracy of 75.1%, which are much lower than the proposed model. This likely due to the fact that the LSTM does not take spatial correlations into consideration. Further, for investigating how much the graph hypothesis is helpful, a ConvLSTM model was also implemented to capture both spatial and temporal dependencies but using a Euclidean-spatial structure. Indeed, ConvLSTM yields MAE, RMSE and accuracy values of 0.383, 0.883 and 79.4% respectively, which are worse than the results of the Multi-STGAC model.

Moreover, to highlight the importance of taking into account daily-periodic and weekly-periodic data, a version of the proposed model that neglects the latter types of segments is implemented (Recent STGAC). We notice that with only recent data, the model has the MAE and RMSE values of 0.317 and 1.001 respectively and an accuracy of 90.5% which show it is less efficient than the proposed model. As it turns out, these two types of data (daily and weekly) are very important in predicting short-term passenger flow. In addition, to test our hypothesis regarding the importance of taking vehicle traffic into account, we launch the model without considering the BNP matrix. The corresponding results show that once again, this model is not as effective as the main proposed model. Also, for capturing how attention layer is helpful, we implemented the proposed model without considering the attention layer, and it shows again that in terms of all evaluation metric the Multi-STGAC model is performing much better than it's version without the attention layer.

We have also defined a scenario in which we classify the passenger flows into 4 bins with size of five, and we calculate the percent accuracy for this scenario too. The results summarized in **Table 1** confirm our findings, and our model is the most efficient according to this metric with a value of 99.3%.

CONCLUSION

Planning transit operations heavily relies on predicting passenger flow, which is affected by numerous factors. The use of deep learning models for forecasting passenger flows is becoming increasingly common. A majority of these methods predict future passenger flow by accounting only for the target stop's historical passenger flow without accounting for the impacts of its neighbors, or by treating their spatial and temporal influences equally, which is not valid in practice. In this study, a Multi-Component Spatial-Temporal Graph Attention Convolution (Multi-STGAC) have been presented to predict passenger flow on a bus network. The framework implements a Graph Attention Network and a Graph Convolution Neural Network that are designed specifically to the bus network for spatial modeling. A gated dilated convolution layer was also used for the temporal modeling in order to capture long sequences of time intervals. Additionally, the model contains a fusion layer that considers short-, mid-, and long-term temporal dependencies simultaneously. Furthermore, with real-world data of the Laval, Canada bus network, an analysis of this new model is carried out to compare it with other deep learning models for predicting passenger flow in terms of MAE, RMSE, and accuracy. The proposed model outperforms the LSTM, ConvLSTM, and the different versions of Multi-STGAC that do not take into account one of its assumptions.

For future work, the model's precision and reliability can be improved by having access to additional data sets with a wider time range. Future work would also consider more patterns representing exogenous dependencies, like particular weather conditions and traffic incidents. Other aspects of extending this research would be considering the influence of network parameters on model performance. Moreover, even though the practicability of the model is verified through a bus case study, the applicability of the model for other scenarios like bike flows, and migration flows, could be examined.

ACKNOWLEDGEMENTS

We would like to convey our heartfelt appreciation to BusPas Inc. and Mitacs Accelerate Program for the support they have dedicated to this research. We are also grateful to Société de Transport de Laval (STL) for providing the data.

AUTHOR CONTRIBUTION STATEMENT

The authors confirm their contribution to the paper as follows: study conception and design: Asiye Baghbani, Nizar Bouguila, and Zachary Patterson; data collection: Nizar Bouguila, and Zachary Patterson; analysis and interpretation of results: Mohamed Chaaben, Asiye Baghbani, Nizar Bouguila, and Zachary Patterson; draft manuscript preparation: Mohamed Chaaben, Asiye Baghbani, Nizar Bouguila, and Zachary Patterson. All authors reviewed the results and approved the final version of the manuscript.

REFERENCES

1. Ceder, A., *Public Transit Planning and Operation: Theory, Modeling and Practice*, 2007.
2. Laña, I., J. L. Lobo, E. Capecci, J. Del Ser, and N. Kasabov, Adaptive long-term traffic state estimation with evolving spiking neural networks. *Transportation Research Part C: Emerging Technologies*, Vol. 101, 2019, pp. 126–144.
3. Noekel, K. and G. Gentile, *Modelling Public Transport Passenger Flows in the Era of Intelligent Transport Systems*, 2016.
4. Zhai, H., L. Cui, Y. Nie, X. Xu, and W. Zhang, A Comprehensive Comparative Analysis of the Basic Theory of the Short Term Bus Passenger Flow Prediction. *Transportation*, , No. 2, 2018.
5. Xue and al., Short-Term Bus Passenger Demand Prediction Based on Time Series Model and Interactive Multiple Model Approach. *Transportation*, , No. 2, 2015.
6. Jiang, W. and J. Luo, Graph neural network for traffic forecasting: A survey. *Expert Systems with Applications*, 2022, p. 117921.
7. Zhang, C. H., S. Rui, and Y. Sun, Kalman Filter-Based Short-Term Passenger Flow Forecasting on Bus Stop. *Journal of Transportation Systems Engineering and Information Technology*, Vol. 11, No. 4, 2011, pp. 155–156.
8. Yang, Z., Q. Zhao, S. Zhao, L. Jin, and Y. Mao, Passenger flow volume forecasting method based on public transit intelligent card (IC) survey data. *Transport Research*, , No. 9, 2009, pp. 115–119.
9. GU Yang, HAN Yin, and FANG Xueli, Method of Hub Station Passenger Flow Forecasting Based on ARMA Model. *Journal of Transport Information and Safety*, 2011.
10. Xue, R., D. J. Sun, and S. Chen, Short-term bus passenger demand prediction based on time series model and interactive multiple model approach. *Discrete Dynamics in Nature and Society*, Vol. 2015, 2015.
11. Yang Xinfeng and Liu Lanfen, Short-term passenger flow prediction of bus stops based on support vector machine based on AP clustering. *Journal of Wuhan University of Technology (Transportation Science and Engineering Edition)*, Vol. 40, No. 1, 2016, pp. 36–40.
12. Chen, Q., W. Li, and J. Zhao, The use of LS-SVM for short-term passenger flow prediction. *Transport*, Vol. 26, No. 1, 2011, pp. 5–10.
13. Lv, W., Y. Lv, Q. Ouyang, and Y. Ren, A Bus Passenger Flow Prediction Model Fused with Point-of-Interest Data Based on Extreme Gradient Boosting. *Applied Sciences*, Vol. 12, No. 3, 2022, p. 940.
14. Shen, C., L. Zhu, G. Hua, L. Zhou, and L. Zhang, A Deep Convolutional Neural Network Based Metro Passenger Flow Forecasting System Using a Fusion of Time and Space. In *2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)*, IEEE, 2020, pp. 1–5.
15. Zhang, K., Z. Liu, and L. Zheng, Short-term prediction of passenger demand in multi-zone level: Temporal convolutional neural network with multi-task learning. *IEEE transactions on intelligent transportation systems*, Vol. 21, No. 4, 2019, pp. 1480–1490.
16. Liu, L., R.-C. Chen, and S. Zhu, Passenger flow prediction using weather data for metro systems. In *2018 conference on technologies and applications of artificial intelligence (TAAI)*, IEEE, 2018, pp. 70–73.

17. Al-Molegi, A., M. Jabreel, and B. Ghaleb, STF-RNN: Space Time Features-based Recurrent Neural Network for predicting people next location. In *2016 IEEE Symposium Series on Computational Intelligence (SSCI)*, IEEE, 2016, pp. 1–7.
18. Li, Y., M. Yin, and K. Zhu, Short Term Passenger Flow Forecast of Metro Based on Inbound Passenger Plow and Deep Learning. In *2021 International Conference on Communications, Information System and Computer Engineering (CISCE)*, IEEE, 2021, pp. 777–780.
19. Guo, J., Z. Xie, Y. Qin, L. Jia, and Y. Wang, Short-term abnormal passenger flow prediction based on the fusion of SVR and LSTM. *IEEE Access*, Vol. 7, 2019, pp. 42946–42955.
20. Nagaraj, N., H. L. Gururaj, B. H. Swathi, and Y.-C. Hu, Passenger flow prediction in bus transportation system using deep learning. *Multimedia tools and applications*, Vol. 81, No. 9, 2022, pp. 12519–12542.
21. Wang, J., R. Wang, and X. Zeng, Short-term passenger flow forecasting using CEEMDAN meshed CNN-LSTM-attention model under wireless sensor network. *IET Communications*, 2022.
22. Xiao, W. and H. Xu, A novel bus scheduling model based on passenger flow and bus travel time prediction using the improved cuckoo search algorithm. In *2022 International Conference on Big Data, Information and Computer Network (BDICN)*, IEEE, 2022, pp. 208–212.
23. Zhao, J.-d., J. Shen, and L.-w. Liu, Bus passenger flow classification prediction driven by CNN-GRU model and multi-source data. *J. Traffic Transport. Eng*, Vol. 21, 2021, pp. 265–273.
24. Asif, N. A., Y. Sarker, R. K. Chakrabortty, M. J. Ryan, M. H. Ahamed, D. K. Saha, F. R. Badal, S. K. Das, M. F. Ali, S. I. Moyeen, et al., Graph neural network: A comprehensive review on non-euclidean space. *IEEE Access*, Vol. 9, 2021, pp. 60588–60606.
25. Kong, X., K. Wang, M. Hou, F. Xia, G. Karmakar, and J. Li, Exploring Human Mobility for Multi-Pattern Passenger Prediction: A Graph Learning Framework. *IEEE Transactions on Intelligent Transportation Systems*, 2022.
26. Han, Y., S. Wang, Y. Ren, C. Wang, P. Gao, and G. Chen, Predicting station-level short-term passenger flow in a citywide metro network using spatiotemporal graph convolutional neural networks. *ISPRS International Journal of Geo-Information*, Vol. 8, No. 6, 2019, p. 243.
27. Arabghalizi, T. and X. Jia, A Hybrid Neural Network to Predict Short-term Passenger Flow at Bus Stops, 2021.
28. Ma, D., Y. Guo, and S. Ma, Short-Term Subway Passenger Flow Prediction Based on GCN-BiLSTM. In *IOP Conference Series: Earth and Environmental Science*, IOP Publishing, 2021, Vol. 693, p. 012005.
29. Pan, Z., Y. Liang, W. Wang, Y. Yu, Y. Zheng, and J. Zhang, Urban traffic prediction from spatio-temporal data using deep meta learning. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, 2019, pp. 1720–1730.
30. Han, Y., T. Peng, C. Wang, Z. Zhang, and G. Chen, A hybrid GLM model for predicting citywide spatio-temporal metro passenger flow. *ISPRS International Journal of Geo-Information*, Vol. 10, No. 4, 2021, p. 222.
31. Li, L., J. Xu, S. T. Ng, J. Zhang, S. Zhou, and Y. Yang, Attention-based Graph Neural

Network Enabled Method to Predict Short-term Metro Passenger Flow. In *2020 5th International Conference on Universal Village (UV)*, IEEE, 2020, pp. 1–6.

32. Chen, T., J. Fang, M. Xu, Y. Tong, and W. Chen, Prediction of Public Bus Passenger Flow Using Spatial–Temporal Hybrid Model of Deep Learning. *Journal of Transportation Engineering, Part A: Systems*, Vol. 148, No. 4, 2022, p. 04022007.
33. Ye, J., J. Zhao, K. Ye, and C. Xu, How to build a graph-based deep learning architecture in traffic domain: A survey. *IEEE Transactions on Intelligent Transportation Systems*, 2020.
34. Wu, Z., S. Pan, G. Long, J. Jiang, and C. Zhang, Graph wavenet for deep spatial-temporal graph modeling. *arXiv preprint arXiv:1906.00121*, 2019.
35. Bengio, Y., P. Simard, and P. Frasconi, Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks*, Vol. 5, No. 2, 1994, pp. 157–166.
36. Dauphin, Y. N., A. Fan, M. Auli, and D. Grangier, Language modeling with gated convolutional networks. In *International conference on machine learning*, PMLR, 2017, pp. 933–941.
37. Cheng, J., L. Dong, and M. Lapata, Long short-term memory-networks for machine reading. *arXiv preprint arXiv:1601.06733*, 2016.
38. Lin, Z., M. Feng, C. N. d. Santos, M. Yu, B. Xiang, B. Zhou, and Y. Bengio, A structured self-attentive sentence embedding. *arXiv preprint arXiv:1703.03130*, 2017.
39. Velickovic, P., G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, Graph attention networks. *stat*, Vol. 1050, 2017, p. 20.
40. STM, *Société de transport de Montréal, The bus network and the schedules enlightened*. `https://www.stm.info/en/info/networks/bus-network-and-schedules-enlightened`, 2022.
41. Lydia, A. and S. Francis, Adagrad—an optimizer for stochastic gradient descent. *Int. J. Inf. Comput. Sci*, Vol. 6, No. 5, 2019, pp. 566–568.
42. Cui, Z., K. Henrickson, R. Ke, and Y. Wang, Traffic graph convolutional recurrent neural network: A deep learning framework for network-scale traffic learning and forecasting. *IEEE Transactions on Intelligent Transportation Systems*, Vol. 21, No. 11, 2019, pp. 4883–4894.