MODELING CROWDEDNESS AT PUBLIC TRANSPORT STATIONS DURING SPECIAL EVENTS: A COMPARATIVE STUDY OF ELEVEN CITIES

Wanrong Hu
Chair of Transportation Systems Engineering, Technical University of Munich, Germany
Email: diana.hu@tum.de

Qing-Long Lu, Corresponding Author
Chair of Transportation Systems Engineering, Technical University of Munich, Germany
Email: qinglong.lu@tum.de

Ningkang Yang
Chair of Transportation Systems Engineering, Technical University of Munich, Germany
Email: ningkang.yang@tum.de

Constantinos Antoniou, Ph.D.
Chair of Transportation Systems Engineering, Technical University of Munich, Germany
Email: c.antoniou@tum.de

Word Count: 5538 words + 5 table(s) $\times$ 250 = 6788 words

Submission Date: August 2, 2024

ABSTRACT

Special events are the primary trigger for the abnormal crowdedness patterns present in public transport (PT) systems. Accurately predicting these changes in crowdedness can enhance PT scheduling and crowd management, making it a crucial aspect of public transport demand management (PTDM). This study aims to forecast short-term crowdedness at PT stations during planned special events using publicly available opportunistic data. More specifically, we take football matches as an example of special events, and use Google Popular Times (GPT) as opportunistic data given its extensive spatiotemporal coverage in urban areas. We conduct a comparative analysis involving 15 football clubs from 11 different cities. We develop a graph-based neural network augmented with an attention mechanism and a positional embedding-enhanced temporal convolutional network to predict crowdedness at metro stations. Additionally, we design a specific event indicator to capture the unique temporal characteristics associated with special events. The experimental results demonstrate that our proposed model effectively forecasts crowdedness at metro stations, with the inclusion of the event indicator significantly enhancing prediction accuracy during special events.

Keywords: Crowdedness patterns, special events, opportunistic data, comparative analysis, graph neural network

INTRODUCTION

Special events, whether planned or unplanned, such as the Olympics, football games, concerts, and other large gatherings, play a significant role in our lives and culture. However, these events also cause an abnormal increase in traffic demand, potentially decreasing service levels or even leading to network failures(*1*). Urban rail transport systems are pivotal in addressing the challenges posed by urban sprawl and suburbanization, offering substantial social, economic, and environmental benefits globally.

Effective planning and management are crucial to handle the demand changes triggered by special events. Short-term forecasting of passenger flow, typically predicting trends within a 5-minute to 1-hour window, is integral to urban rail transit management and crowd regulation. Accurate short-term forecasts enable operators to optimize service schedules, plan station crowd regulation measures, and enhance passenger information systems. These forecasts ensure the transportation supply aligns precisely with passenger demand, facilitating emergency response planning. Therefore, evaluating the evolution of crowd patterns and forecasting passenger flow during special events is essential for effectively developing strategies to address these challenges.

Rail passenger forecasting is a vital topic in transportation engineering with wide-ranging applications. It can be broadly divided into long-term forecasting, which usually spans years or months and guides the planning and construction of rail systems, and short-term forecasting, which is the focus of our research. Depending on the data format used, these forecasting problems can be classified into various types, including time series, grid, graph, and hybrids like temporal-spatial data. Traditional statistical models such as autoregressive integrated moving average (ARIMA) (*2*) and its variants, including seasonal autoregressive integrated moving average (SARIMA) (*3*) and generalized autoregressive conditional heteroscedasticity (GARCH) (*4*), have been widely used in traffic demand prediction. More recently, robust principal component analysis (RPCA) (*5*) has also been employed to address some limitations of ARIMA models.

With the advent of AI technology, deep learning models have gained popularity in transportation system research due to their efficiency and accuracy in handling large-scale data and capturing nonlinear relationships. Despite the large number of models developed for traffic flow forecasting in recent years, they can be categorized into convolutional neural network (CNN)-based models, recurrent neural network (RNN)-based models, Graph neural network (GNN)-based models, and various hybrids or variants. Most recent studies in short-term traffic state forecasting show that while statistical models primarily focus on temporal dependencies, deep learning models excel by incorporating both temporal and spatial features (*6*).

Despite the growing interest in deep learning models for short-term rail transit system passenger forecasting, particularly during special events, most studies rely on data from automated fare collection systems, which are not always available. For example, there are no such systems for public transport in most cities in Germany. Our study addresses these gaps using Google Popular Times (GPT) data. Specifically, we collected popularity data on public transport stations across ten cities and 17 metro lines to capture the temporal-spatial characteristics of passenger crowding. We employ a graph-based neural network, the attention with a positional embedding enhanced temporal convolutional network (APT-GCN) to predict crowdedness at metro stations, specifically focusing on the patterns under special events. Our model is trained and tested on this dataset to analyze crowdedness patterns across various public transport networks.

The rest of the paper is structured as follows: following this introduction, we detail the process of data collection and present our model in the Methodology section. Next, we train the

proposed model and discuss testing results. Finally, the paper concludes with a summary of our findings.

DATA COLLECTION AND PROCESSING

In this section, we present a comprehensive overview of the data collection and processing. The dataset is bifurcated into three components: special events data, public transport network data and Google Popular Times (GPT) data.

The special events dataset is first elaborated upon, where we define and describe the specific events considered in this study. These events are contextualized within the public transport network and correlated with transit patterns. The second component pertains to the spatial domain, where the topological relationships of stations along transit lines are collected. Lastly, the temporal domain is addressed. The raw GPT data were collected from 428 public transport stations across 17 transit lines in ten cities, spanning nearly four months. Following the description of the GPT data, we performed data analysis and imputation. We also construct a new dummy feature to encapsulate the special event information.

Special Events Dataset

Special events, such as concerts, sports events, and festivals, significantly impact public transportation usage and passenger flow. We selected football matches as a representative type of special event for our study. Football, being one of the most popular sports globally, often attracts large audiences, enriching people's lives but also exerting significant pressure on public transport systems. Our study focuses on 15 clubs from top European football leagues and the public transport routes passing through their home stadiums. Table 1 provides an overview of the basic information about these clubs and their associated transit lines. Stops represent the stations associated with the stadiums and are typically identified using transportation information provided on the clubs' official websites, where line ID is a unique identifier for each public transport line.

Naturally, the various leagues and cup matches hosted at these stadiums are considered our special events. Within the time frame of our data collection, 122 such events occurred. We use the kickoff time of each match as the event start time and consider two hours post-kickoff as the event end time, accounting for the typical duration of a football match. Table 2 illustrate an example of these events.

Public Transport Network Topology

For the urban rail transit lines involved in the aforementioned special events, we collected their network maps to establish the spatial topological structures. These structures can be categorized into three types: linear, circular, and branching. Stations that are spatially adjacent are defined as those that are connected directly by a single segment of the rail line without any intermediate stops.

Google Popular Time

GPT data properties

GPT data is provided for each Point of Interest (POI). The raw data from GPT includes historical popularity, current popularity, visit duration, and wait time estimates.

- Current Popularity: This indicates the crowd level at a given POI at the current time.
- Historical Popularity: This is based on average popularity over the past several weeks, providing a popularity value for each hour of each day of the week, resulting in 7 *times*

TABLE 1: Overview of Cities, Public Transport Lines, and Stadiums

City	Line ID	Stops	Football Club
Munich	U6	Fröttmanning	Bayern Munich
Berlin	S3	Berlin-Köpenick	1.FC Union Berlin
Dortmund	U45, U46	Westfalenhallen	Borussia Dortmund
Köln	Tram 1	RheinEnergieStadion	FC Köln
Augsburg	Tram 3, Tram 8	Augsburg WWK ARENA	FC Augsburg
Madrid	Line 10	Santiago Bernabéu	Real Madrid
Madrid	Line 7	Estadio Metropolitano	Atletico Madrid
Madrid	Line 12	Los Espartales	Getafe
Madrid	Line 1	Portazgo	Rayo Vallecano
London	Piccadilly	Arsenal	Arsenal
London	District	Fulham Broadway	Chelsea
Newcastle	Yellow	St James	Newcastle United
Marseille	M2	Sainte-Marguerite Dromel	Olympique de Marseille
Copenhagen	M3	Trianglen St.	F.C. Copenhagen
Lisbon	Blue	Colégio Militar/Luz	Benfica

TABLE 2: An Example of Special Events Data

City	Line	Event	Start Time	End Time
Copenhagen	M3	FC Copenhagen-Manchester City	Feb 13, 21:00	Feb 13, 22:55
Lisbon	Blue	Benfica-Toulouse	Feb 15, 21:00	Feb 15, 22:58
Köln	Tram 1	Köln-Bremen	Feb 16, 20:30	Feb 16, 22:27
Madrid	Line 7	Atletico Madrid-Las Palmas	Feb 17, 14:00	Feb 17, 15:51
London	Yellow	Newcastle-Bournemouth	Feb 17, 16:00	Feb 17, 17:58
Lisbon	Blue	Benfica-Vizela	Feb 18, 19:00	Feb 18, 20:58
Marseille	M2	Marseille-Shakhtar	Feb 22, 21:00	Feb 22, 22:59

24 data points. It shows how crowded the location typically is during different times of the day.

- Visit Duration: This data shows how much time customers typically spend at the specific POI, estimated based on patterns of customer visits over the last several weeks.
- Wait Time Estimates: This shows how long a customer would have to wait before receiving service during different times of the day.

We extract current popularity as the primary indicator of station crowding. Additionally, we organize other data points as potential features for future use, providing varies models with more comprehensive information for analysis.

Data quality analysis

We collected GPT data for the aforementioned metro stations from February 13, 2024, to June 4, 2024, resulting in 4617 timestamps. Despite our best efforts, GPT data is not always available at all stations and times. Data collection failures may stem from the inherent unavailability of data or from unavoidable issues like network instability during the collection period. Therefore, we first analyze the collected data to provide a basis for subsequent data processing.

Our analysis is conducted at the level of each subway line and station. Taking Madrid Metro Line 1 as an example, Figure 1 shows the statistics of popularity values at various stations during the data collection period, and Figure 2 illustrates the distribution of data points collected over a week. Given our setting of collecting data from all stations every 30 minutes, there should ideally be 336 data points per week. Stations with abnormal data, poor quality, or significant missing values, such as Atocha station in this instance, are removed from the dataset. Subsequently, we proceed with data imputation to address the missing data issue.

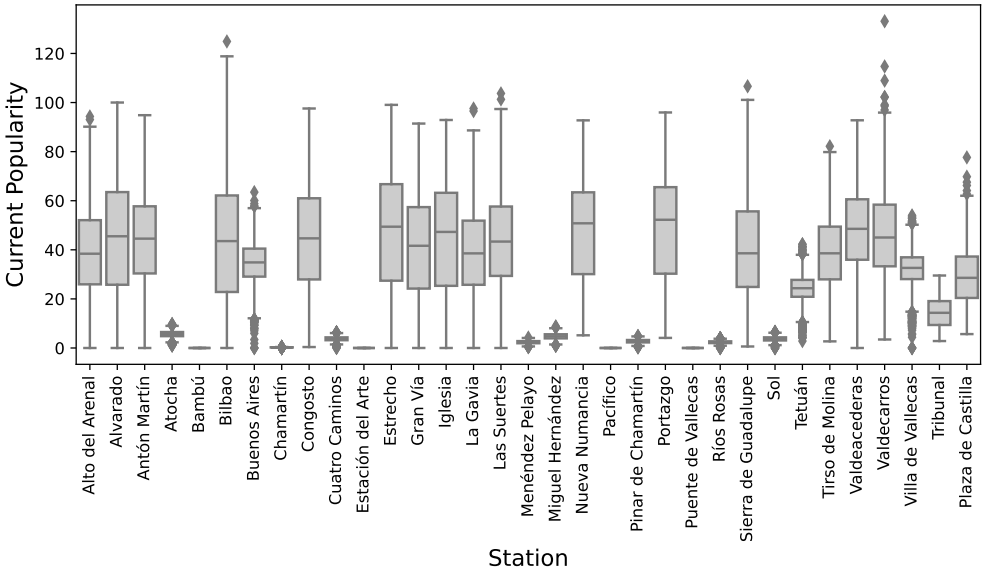


FIGURE 1: GPT popularity value distribution of stations in Madrid Line 1

Data imputation with Temporal Regularized Matrix Factorization (TRMF)

In this section, we introduce the data imputation algorithm that is applied to complete the GPT dataset. Our problem can be defined as a random missing spatial-temporal data problem, where each sensor loses observations completely at random. To address this issue, TRMF is employed. This model incorporates temporal dependencies into the matrix factorization process(*7*). Temporal dependencies are represented by $\mathbf{x}_i$,

$$x_t \approx \sum_{l \in \mathscr{L}} \theta_l \odot x_{t-l} \tag{1}$$

Then,

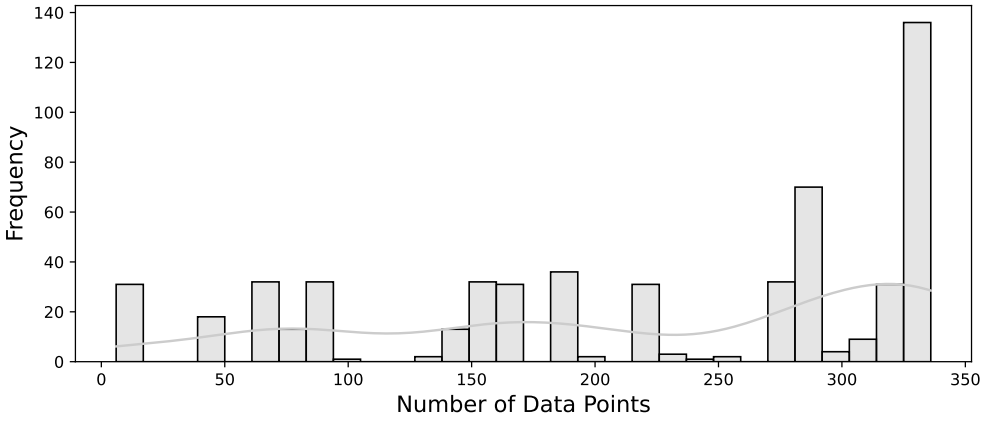


FIGURE 2: Number of collected data points of stations in Madrid Line 1

$$\mathscr{R}_{\mathrm{AR}}(X \mid \mathscr{L}, \Theta, \eta) = \frac{1}{2} \sum_{t=l_d+1}^{f} \left(x_t - \sum_{l \in \mathscr{L}} \theta_l \odot x_{t-l} \right)^{\top} \left(x_t - \sum_{l \in \mathscr{L}} \theta_l \odot x_{t-l} \right) + \frac{\eta}{2} \sum_{t=1}^{f} x_t^{\top} x_t \tag{2}$$

where $\mathscr{L} = \{l_1, l_2, \ldots, l_d\}$ is a lag set, in our task is setting to $\mathscr{L} = \{1, 2, 48\}$, and $\theta_l \in \mathbb{R}^r, \forall l$ is weight to decide autoregressive (AR). Thu the TRMF is to solve,

$$\min_{W,X,\Theta} \frac{1}{2} \sum_{(i,t) \in \Omega} \left(y_{it} - w_i^{\top} x_t \right)^2 + \lambda_w \mathscr{R}_w(W) + \lambda_x \mathscr{R}_{\mathrm{AR}}(X \mid \mathscr{L}, \Theta, \eta) + \lambda_\theta \mathscr{R}_\theta(\Theta) \tag{3}$$

where $\mathscr{R}_w(W) = \frac{1}{2} \sum_{i=1}^{m} w_i^{\top} w_i$ and $\mathscr{R}_\theta(\Theta) = \frac{1}{2} \sum_{l \in \mathscr{L}} \theta_l^{\top} \theta_l$ are regularization terms.

Consequently, a spatial-temporal popularity data with a 30-minute interval based on GPT was obtained by applying data imputation.

In summary, this section sets the groundwork for understanding the data-related approaches employed in our study, providing a detailed account of the dataset's structure, collection, and enhancements, ensuring the reliability of our dataset for subsequent analysis.

METHODOLOGY

Problem Definition

The primary objective of this study is to analyze the crowding patterns at public transport stations during special events and to forecast future crowdedness based on historical data. Crowdedness is quantified using the popularity data derive from Google Popular Times, as previously mentioned. The study is divided into two main tasks. The first task involves a cross-analysis of the crowding patterns along public transport lines in different cities to identify similarities and differences. The second task is treated as a short-term traffic forecasting problem. This involves proposing an attention with positional embedding enhanced temporal convolutional network (APT-GCN) model to predict the future popularity at each station based on a 4-hour historical popularity data.

Definition 1 Public Transport Network *G*. A public transport network is described as an unweighted, undirected graph $G = (V, E)$, representing the inherent typologies of public transport lines. Here, the graph's node set $V = \{v_1, v_2, \ldots, v_N\}$ represents the set of stations, where N is the number of stations. The set of edges E represents the connections between stations, with an edge existing between each pair of adjacent stations.

Definition 2 Adjacency Matrix $\mathbf{A}^{N\times N}$. The adjacency matrix $\mathbf{A} \in \mathbb{R}^{N\times N}$ is a binary matrix that represents the connections between stations in the public transport network G. Each element a_{ij} in the matrix is defined as follows:

$$a_{ij} = \begin{cases} 1 & \text{if there is an edge between station } v_i \text{ and station } v_j, \\ 0 & \text{otherwise.} \end{cases} \tag{4}$$

Definition 3 Feature Matrix $\mathbf{X}^{N\times M}$. Short-term popularity sequence is considered an attribute of network nodes and is represented by the feature matrix $\mathbf{X} \in \mathbb{R}^{N\times S}$, where S denotes the length of the historical time series. At time t, $X_t \in \mathbb{R}^{i\times N}$ represents the popularity at each node, while i corresponds to the dimensionality of the attribute data, accommodating scenarios where there are multiple attributes.

Defination 4 Multi-step popularity forecasting. In the context of multi-step popularity forecasting, the goal is to learn a function f that maps the public transport network G and the feature vector $\mathbf{x} \in \mathbb{R}^{1\times N}$ (in some case, feature matrix $\mathbf{X} \in \mathbb{R}^{i\times N}$ considering the multiple attribute features) to future T-step predictions of popularity for each station. This can be formally described as:

$$(\mathbf{x}_{t+1}, \ldots, \mathbf{x}_{t+T}) = f\left(G, (\mathbf{x}_{t-s}, \ldots, \mathbf{x}_{t-1}, \mathbf{x}_t)\right) \tag{5}$$

where s denotes the length of the historical popularity sequences, and T represents the number of future steps to be predicted.

Overview of the model architecture

In this study, we proposed a spatial-temporal model with a graph attention module enhanced by position embedding mechanism (APT-GCN). Figure 3 illustrate the structure of the APT-GCN model. In the first part, the method to capture temporal-spatial dependencies is based on the Temporal Graph Convolutional Network (TGCN) proposed by (*8*). It begins by taking time series data of popularity, with a length s for each node, and the adjacency matrix $\mathbf{A}$ as input. A graph convolution network (GCN) is then used to capture spatial features from this data. The output of the GCN is subsequently fed into a gated recurrent unit (GRU), which allows for the temporal flow of information across different time snapshots. Following the T-GCN module, the output is further processed by an attention mechanism designed to integrate historical information from non-adjacent stations. This is achieved through a position embedding that enhances the attention module's ability to incorporate information from distant nodes. Finally, the results from the attention module are aggregated along the temporal dimension to produce forecasts for the next **T** time steps. This model effectively combines short-term historical data for each node with information from other nodes in the graph, enabling more accurate predictions of future popularity trends.

Combining spatial-temporal popularity data with special event data

We will add an event label to the popularity data to capture the unique passenger flow characteristics during special events. For each special event, the start and end times have been recorded. Using this information, we will introduce a new feature, 'event,' to the popularity data. The specific indicator is expressed in 6, designating a buffer time of 2 hours before the start and after the end of the event. The event feature for the corresponding station j will be set to 1 if it falls within this time range, and 0 otherwise.

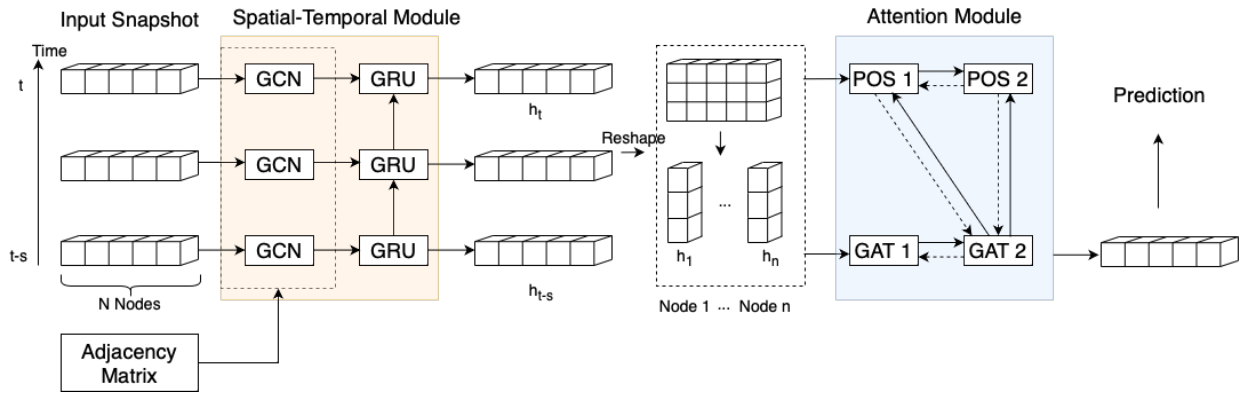


FIGURE 3: An overview of the attention with positional embedding enhanced temporal convolutional (APT-GCN) network.

$$event_{ij} = \begin{cases} 10-i & \text{if } t_{ij} \in (t_{start} - i \times 30\text{minutes}, t_{end} + i \times 30\text{minutes}) \bigwedge v_j \in \mathbf{L}_m, i < 10 \\ 0 & \text{otherwise.} \end{cases} \tag{6}$$

where t_{ij} is the time stamp i of station j, t_{start} and t_{end} represent the start time and end time of event m, $\mathbf{L}+m$ is the public transport line corresponding to event m.

Gathering neighbours' influence via graph convolution layer

Graph Convolutional Networks (GCNs) are a type of semi-supervised model introduced by Kipf and Welling in 2016 (*9*). GCNs enhance traditional CNNs within the domain of graph neural networks. While CNNs can capture local spatial features in Euclidean space, such as those in images, they fall short in domains like transportation where spatial dependencies arise from network topological structures. GCNs address this limitation by performing convolution operations on non-Euclidean data.

The fundamental idea of GCNs is to aggregate features from neighboring nodes and then transform these features. By stacking k layers, a GCN can capture the features of k-order neighbors. Given the adjacency matrix $\mathbf{A}$, which represents non-Euclidean graph data, a GCN typically performs two operations: propagation and transformation. Propagation involves using filters in the Fourier space to capture and aggregate features from first-order neighbors. The aggregated spatial information is then transformed between layers through linear transformations or activation functions. Given the feature matrix $\mathbf{X}$ and the matrix $\tilde{\mathbf{A}}$ representing the network structure, the output of GCN layer $l+1$ is computed as follows:

$$H^{(l+1)} = \sigma\left(\tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}H^{(l)}\theta^{(l)}\right) \tag{7}$$

where $\tilde{H}^{(l)}$ is the output of layer l, $\tilde{\mathbf{A}} = \mathbf{A}+\mathbf{I}$ denotes adjacency matrix adding self-connection, $\mathbf{I} \in \mathbb{R}^{N\times N}$ is an identity matrix. $\tilde{\mathbf{D}}$ is the degree matrix computed by $\tilde{\mathbf{D}} = \sum_j \tilde{\mathbf{A}}_{ij}$. $\theta^{(l)}$ denote the parameter of layer $(l+1)$. $\sigma(\cdot)$ represents the activation function.

Thus, a graph convolutional network can be defined as:

$$H^{(l)} = f(H,\hat{A}) = \sigma\left(\hat{A}H^{(l-1)}W^{(l)}\right) \tag{8}$$

$$Z = f^{(n)}\left(X, \hat{A}\right) = \underbrace{f\left(f\left(\ldots f(X, \hat{A})\ldots\right)\right)}_{n \text{ layers}} \tag{9}$$

$\hat{\mathbf{A}} = \tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}$ which is a pre-process of adjacency matrix. $\mathbf{W}^{(l)} \in \mathbb{R}^{S\times C}$ denoted the weight of layer l, where S is the length of historical popularity series, and C denotes number of units. n is the number of layer, to capture spatial dependencies from n-order neighbours. In our model, n is set to be 2. $\mathbf{X}$ is input feature matrix and $\mathbf{Z} \in \mathbb{R}^{N\times C}$ is the final output of GCN network and C is the number of units from output layer.

Gated recurrent units for temporal domain

When addressing transportation issues—whether they pertain to road traffic, rail transit, or a combination of public transportation types—participants invariably move through a specific network along the time dimension. In the short term, historical data significantly influences the prediction of traffic related variables at a given moment. This relationship has been highlighted in various studies, which emphasizes the importance of incorporating historical data to enhance the accuracy of traffic forecasts (*10*). Therefore, one of the primary objectives in traffic forecasting is to capture temporal dependencies effectively. This goal can be achieved using RNNs, which are designed to handle sequence data and capture dependencies across time. RNNs have been extensively studied for their ability to model temporal sequences and their applications in time-series forecasting.

RNNs come in several variants, with Long Short-Term Memory (LSTM) units and Gated Recurrent Units (GRUs) being among the most notable. LSTM units were introduced by Hochreiter and Schmidhuber in 1997(*11*) to address the vanishing gradient problem in traditional RNNs (*12*). LSTMs use a more complex structure involving input, forget, and output gates to control the flow of information, which helps them maintain long-term dependencies. On the other hand, GRUs, introduced by Cho et al. in 2014, simplify this structure by combining the input and forget gates into a single update gate, making them computationally less demanding (*13*). While both models utilize gating mechanisms to control the flow of information, LSTMs generally involve more complex operations, which can lead to longer training times (*14*). In our study, we evaluated the performance of GRUs and LSTMs on sample data, as well as the combination of these units with single-layer models. Experimental results indicated that while LSTM and the combined model did not significantly improve accuracy under the same conditions, they considerably increased the training time. During our experiment on several lines, the GRU layer achieved an accuracy of 0.9145, while the LSTM layer option slightly improved accuracy to 0.9181, marking a 0.39% increase. However, this improvement in accuracy came at the cost of a 19.16% increase in training time, rising from 2696.31 seconds to 3213.82 seconds. Therefore, considering the trade-offs between accuracy and computational efficiency, we ultimately chose GRUs as the recurrent units to capture temporal information in our model. The GRU cell processes the output $\mathbf{Q} \in \mathbb{R}^{N\times C}$ from graph convolutional network by:

$$Q_t = (1 - U_t)\, Q_{t-1} + U_t C_t \tag{10}$$

$\mathbf{U}_t$ is the update gate to control information from last time step, it along with memory content at current moment $\mathbf{C}_t$ are computed as:

$$U_t = \sigma(W_U[Z_t, Q_{t-1}] + b_U) \tag{11}$$

$$R_t = \sigma(W_R[Z_t, Q_{t-1}] + b_R) \tag{12}$$

$$C_t = \tanh(W_C Z_t + W_C(R_t Z_{t-1}) + b_C) \tag{13}$$

where $\mathbf{Z}_s \in \mathbb{R}^{N\times C}, s \in (0,S)$ is the output of last graph convolutional section. Comparing the original GRU unit, here adding $\mathbf{R}_t \in \mathbb{R}^{N\times C}$ as the reset gate, considering the degree of ignoring information from early time steps.

The output of the GRU, incorporating temporal dynamics, will be passed to the subsequent attention module. Noticeably, the output $\mathbf{Q}$ from the previous module has not aggregated the s time steps along the time sequence, thus still retaining the information in the time series dimension.

Graph attention module for enhancing non-Local dependencies

Crowdedness forecasting in public transport systems, as a subset of general traffic prediction, inherits characteristics from traffic forecasting while presenting unique challenges. Public transport operates on a fixed service schedule, leading to discrete passenger counts at each station and distinctive passenger flow dynamics. This discrete nature becomes particularly pronounced during special events, where sudden surges in passenger volumes can influence multiple subsequent time steps and nodes within the network (*15*).

In previous modules, we utilized graph convolutional networks (GCNs) to capture spatial features within the network. However, a single-layer GCN aggregates information only from immediate neighbors, neglecting the flow of popularity between non-adjacent stations. This limitation is particularly critical in scenarios where rail public traffic passenger transfers between distant stations occur rapidly. For instance, the total trip duration on Munich's U-Bahn Line 6, which comprises 27 stations, is approximately 51 minutes(*16*). Even considering the normal operation schedule is every 10 minutes on this line, this relatively short duration means that passengers can traverse almost the entire line within a single data collection interval, posing a challenge for single or dual layer GCN to capture neighbour node's features. To address this, merely stacking GCN layers to capture higher-order neighbor information is inefficient and computationally expensive, given that the stations of urban public lines are usually beyond 20-30. Moreover, GCNs lack the capability to assign different weights to different nodes, limiting their effectiveness in heterogeneous networks.

To overcome these limitations, we introduce a graph attention module to capture non-local dependencies effectively. The graph attention mechanism allows the model to weight the importance of each node differently, thus providing a more nuanced representation of the network. This module is further enhanced with a position embedding mechanism, which helps in integrating historical information from non-topologically adjacent stations into the current node's prediction. This mechanism is particularly effective for non-homophilic graphs(*17*), enabling us to capture features from stations that are not geographically or topologically proximate but share similar characteristics. This significantly enhances the accuracy of popularity forecasting, especially under special event conditions. During such events, passenger movement along public transport lines tends to follow more predictable patterns, often involving transfers from major interchange stations, such as airports or train stations, to event-specific locations. This consideration is crucial for improving the precision of our forecasts.

The integration of a graph attention module thus not only improves the capture of spatial

dependencies across distant nodes but also ensures that the model remains efficient and effective. This approach is particularly vital for scenarios involving special events, where unexpected spikes in passenger demand necessitate a robust model capable of dynamic and rapid adaptation. This module is based on the Graph Attentional Networks with Positional Embeddings (GAT-POS) proposed by Ma et al. (17). It comprises two graph attention layers and two positional embedding layers that interact with each other, enhancing the model's ability to capture complex spatial relationships within the data. Taking the output $\mathbf{Q}$ from last section and reshaping it into vectors of node features $\mathbf{q}_v \in \mathbb{R}^{S\times C}$ as input, and $\mathbf{q}_v^{'} \in \mathbb{R}^{1\times C}$ as expected output, the attention coefficients is computing by

$$e_{vu}^{k} = a(W_k q_v + U_k p_v \parallel W_k h_u + U_k p_u) \tag{14}$$

where $\mathbf{W}_k$, $\mathbf{U}_k$ and $\mathbf{a}_k$ are the weights in the k-th attention head. $\parallel$ is the concatenation operation. $\mathbf{p}_v$ is the positional embedding for node v computed by,

$$p_v^l = \sigma(W_{emb}^l p_v^{l-1}) \tag{15}$$

$$\mathbf{L}(p_{v v\in N}, G) = \sum_{v\in N} \sum_{u\in \mathscr{N}(v)} (-log\sigma(p_v^T p_u) - Q\cdot \mathbb{E}_{u'\ P_n(v)} log(\sigma(-p_v^T p_{u'}))) \tag{16}$$

where l is the l-th position embeddings layer, and $\mathbf{W}_{emb}^l$ is the learned weight matrix of position embeddings layer l. Equation 16 describes loss function of the unsupervised embedding model.

The attention coefficients $\mathbf{e}_{vu}$ are normalized by softmax function:

$$\alpha_{vu}^k = softmax_u(e_{vu}) = \frac{exp(e_{vu})}{\sum_{j\in\mathbb{N}_v} exp(e_{vj})} \tag{17}$$

Then, a linear transform is applied to gain the output $\mathbf{q}_v^{'}$ by combining neighbours with normalized coefficients,

$$\mathbf{q}_v^{'} = \sigma(\frac{1}{K}\sum_{k=1}^{K}\sum_{u\in\mathbb{N}_v} \alpha_{vu}^k W^k \mathbf{q}_u) \tag{18}$$

Finally, a fully connection layer is used to combined the S time step and make prediction on future T-step. In conclusion, the APT-GCN model effectively captures the spatial-temporal dependencies in public traffic networks and efficiently predicts future popularity. The first module captures the spatial information of topologically adjacent nodes on the public transit lines, as well as the historical information of the time series. The second module utilizes a graph attention network with position embedding mechanisms to obtain information on semantically adjacent stations in crowdedness patterns.

RESULTS

In this section, we employ the collected and processed GPT datasets to analyze the popularity of various urban public transit stations. Using this data, we train the APT-GCN model on each public transport network and present the forecasting results.

Setting Model Parameters

The model is implemented using TensorFlow 2.16.1, running on a Mac Book Pro with an Apple M2 Max chip with 32 GB of memory and 12 cores. The training process encompasses 1000 epochs, utilizing a batch size of 128. The dataset is partitioned into training, validation, and test sets, adhering to a proportion of 7:1:2. The optimization of the model is facilitated by the adaptive moment estimation (Adam) optimizer, with a learning rate set at 0.001. Furthermore, the units of

the hidden layers are configured to 64. The historical time series data length S is 8, and the model performs one-step predictions with the prediction length T set to 1.

Evaluation Metrics

The following error metrics are introduced to evaluate performance on popularity prediction of the APT-GCN model:

- Root Mean Squared Error (RMSE):

$$\text{RMSE} = \sqrt{\frac{1}{M}\sum_{i=1}^{M}(\mathbf{y}_i - \hat{\mathbf{y}}_i)^2} \tag{19}$$

- Mean Absolute Error (MAE):

$$\text{MAE} = \frac{1}{M}\sum_{i=1}^{M}|\mathbf{y}_i - \hat{\mathbf{y}}_i| \tag{20}$$

- Coefficient of Determination (R^2):

$$R^2 = 1 - \frac{\sum_{i=1}^{M}(\mathbf{y}_i - \hat{\mathbf{y}}_i)^2}{\sum_{i=1}^{M}(\mathbf{y}_i - \bar{\mathbf{y}})^2} \tag{21}$$

- Explained Variance Score (Var):

$$\text{Var} = 1 - \frac{\text{Var}(\mathbf{Y} - \hat{\mathbf{Y}})}{\text{Var}(\mathbf{Y})} \tag{22}$$

- Accuracy:

$$\text{Accuracy} = 1 - \frac{\left\|\mathbf{Y} - \hat{\mathbf{Y}}\right\|_F}{\|\mathbf{Y}\|_F} \tag{23}$$

where M is the number of samples, $\mathbf{y}_i$, $\hat{\mathbf{y}}_i$ and $\bar{\mathbf{y}}_i$represent the real, prediction and average popularity vector of network of ith sample. $\mathbf{Y}$ and $\hat{\mathbf{Y}}$ is the set of $\mathbf{y}_i$ and $\hat{\mathbf{y}}_i$. $\|\cdot\|$ represents the Frobenius norm.

Experiment Results

Overview

Table 4 presents the prediction results of the model across the entire public transport network. For each line, the final results of the evaluation metrics are derived from the arithmetic mean of all the stations on that line. The table provides the evaluation results for both regular conditions and special event scenarios. Since multiple events occurred during the data coverage period for each line, the event results presented in this table are also arithmetic means of the results from all events.

TABLE 4: The Evaluation of Crowdedness Forecasting on GPT dataset under Regular and Special Event Scenario

Evaluation Matrix	Scenario	rmse	mae	r2	var	acc
Line 1	Regular	0.0299	0.0226	0.9511	0.9525	0.9132
	Special Event	0.0243	0.0182	0.9554	0.9586	0.9215
Line 7	Regular	0.0484	0.0356	0.9183	0.9184	0.8804
	Special Event	0.0462	0.0351	0.8952	0.9045	0.8843
Line 10	Regular	0.026	0.018	0.924	0.9244	0.8806
	Special Event	0.0254	0.017	0.898	0.9026	0.8546
Line 12	Regular	0.0288	0.0217	0.9603	0.9603	0.9157
	Special Event	0.0254	0.0198	0.9491	0.9545	0.906
Blue	Regular	0.0526	0.0406	0.9109	0.9138	0.861
	Special Event	0.0503	0.0378	0.895	0.9097	0.859
District	Regular	0.0391	0.0301	0.8978	0.898	0.8834
	Special Event	0.0303	0.0232	0.8744	0.8829	0.8989
Piccadilly	Regular	0.0492	0.0383	0.8662	0.8668	0.8707
	Special Event	0.0415	0.0318	0.8734	0.8745	0.8794
M2	Regular	0.0235	0.0151	0.983	0.9831	0.9099
	Special Event	0.0401	0.0236	0.9662	0.9673	0.8744
M3	Regular	0.0416	0.0312	0.915	0.9155	0.889
	Special Event	0.0473	0.0351	0.8677	0.8737	0.8713
S3	Regular	0.0224	0.017	0.9011	0.9019	0.8652
	Special Event	0.0188	0.014	0.9006	0.9143	0.8775
Tram1	Regular	0.016	0.012	0.9579	0.9585	0.919
	Special Event	0.0211	0.0159	0.872	0.8761	0.9061
Tram3	Regular	0.051	0.0376	0.9525	0.9528	0.8693
	Special Event 2	-	-	-	-	-
U6	Regular	0.0465	0.0346	0.8764	0.8765	0.8564
	Special Event	0.032	0.0252	0.8833	0.8975	0.9009
U45	Regular	0.0471	0.0369	0.8148	0.8169	0.8253
	Special Event 2	-	-	-	-	-
U46	Regular	0.0406	0.0277	0.9533	0.9539	0.8724
	Special Event 2	-	-	-	-	-
Yellow	Regular	0.026	0.0195	0.9569	0.9571	0.9094
	Special Event	0.0249	0.0185	0.9408	0.9423	0.9199

For the transport lines Tram 3 in Augsburg, Germany, and U45 and U46 in Dortmund, Germany, no valid data under special events was available during our collection period; therefore, the results for special events are missing for these lines. Despite this, examining the prediction performance for other lines under special event conditions reveals a notable improvement.

Robustness during special events

In our experiments, we initially tested two models. One considers only the popularity feature for each time series and another additionally includes the encoded event feature. The results for Madrid metro Line 1, Line 7 and Munich metro U6 present in Table 5, indicating that while the inclusion of the event feature might lead to a decrease in overall performance across the entire test set, such as with Line 1, Line 7 and U6 where the average accuracy of the second model decreased by 0.94%, 1.03% and 0.94% respectively, it significantly improves performance in special event scenarios by 216.78%, 20.04% and 54.21%. This demonstrates that our model is more robust in capturing crowdedness patterns during special events, enhancing the accuracy of popularity predictions under these conditions. This improvement is visualized in Figure 4, which clearly shows the enhanced prediction accuracy of our model during special events.

TABLE 5: Performance Comparison of Different Models

Line	Scenario	Model	rmse	mae	r2	var	acc
Line 1	Regular	Model 1	0.027	0.0201	0.9603	0.9603	0.9218
		Model 2	0.0299	0.0226	0.9511	0.9525	0.9132
	Special Event	Model 1	0.6417	0.6397	-67.1586	0.5764	0.2909
		Model 2	0.0243	0.0182	0.9554	0.9586	0.9215
U6	Regular	Model 1	0.0439	0.0338	0.89	0.8901	0.8645
		Model 2	0.0465	0.0346	0.8764	0.8765	0.8564
	Special Event	Model 1	0.1927	0.151	-0.0924	-0.0841	0.5842
		Model 2	0.032	0.0252	0.8833	0.8975	0.9009
Line 7	Regular	Model 1	0.0447	0.0331	0.9303	0.931	0.8895
		Model 2	0.0484	0.0356	0.9183	0.9184	0.8804
	Special Event	Model 1	0.1647	0.1235	0.2617	0.3587	0.7367
		Model 2	0.0462	0.0351	0.8952	0.9045	0.8843

Interpretation and visualisation of crowdedness pattern

To further examine the details of the experiment results, Figure 5a visualizes the prediction results for a Germany Bundesliga (first-class) football match on a weekday. Specifically, Figure 5b shows a comparison between the predicted results(**?**) and the ground truth on the training set is demonstrating.

This result represents a typical crowdedness pattern at public transport stations near venues during special events. Popularity increases and peaks approximately two hours before the event begins (as observed in Munich), decreases during the event, and then reaches another peak im-

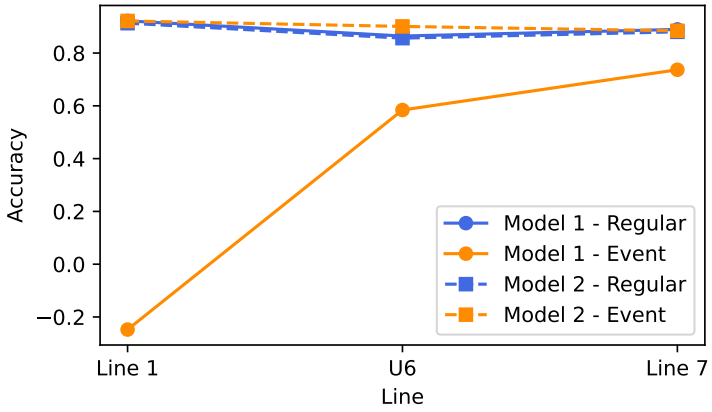
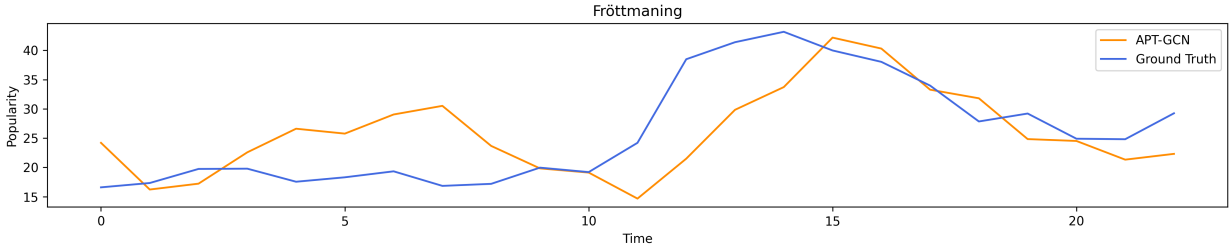
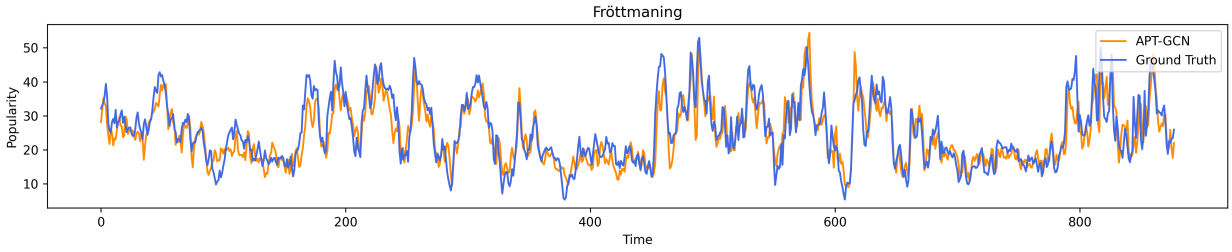


FIGURE 4: Model Performance Analysis



(a) Prediction During Special Event



(b) Prediction on Training Set

FIGURE 5: Forecasting Crowdedness at Munich Metro Line U6 station Fröttmaning

mediately after the event ends as passengers gradually disperse over time. Figure 6 illustrates the popularity comparison between special event periods and average value on weekday and weekend, with Special Event 1 representing an event held on a weekend and Special Event 2 representing an event held on a weekday.

When examining the entire public transport line, it is evident that crowdedness patterns vary significantly across different events, cities, and networks. Figure 7(a) illustrates the distribution of popularity on the Madrid Metro Line 7 during the La Liga match between Atletico Madrid and Betis on March 3, 2024. This match began at 16:15 and ended at 18:15. The blue font corresponds to the station Estadio Metropolitano, which is closest to the stadium. Surrounding stations experienced an increase in popularity before and after the match, whereas Estadio Metropolitano itself did not show a significant increase, potentially due to the station's design.

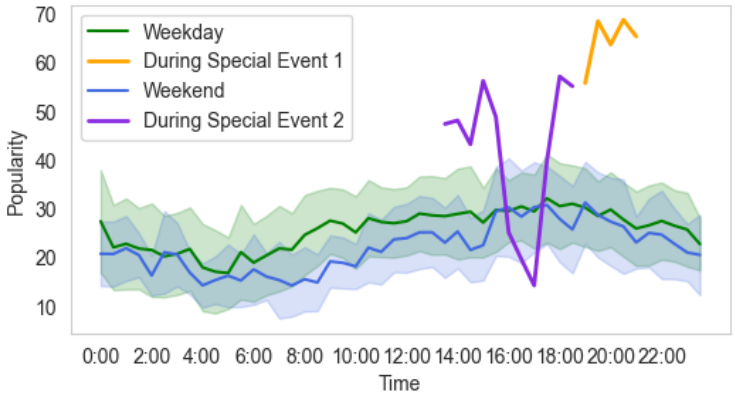


FIGURE 6: Popularity Comparison during Special Event and Regular Times at Munich Metro Line U6 station Fröttmaning

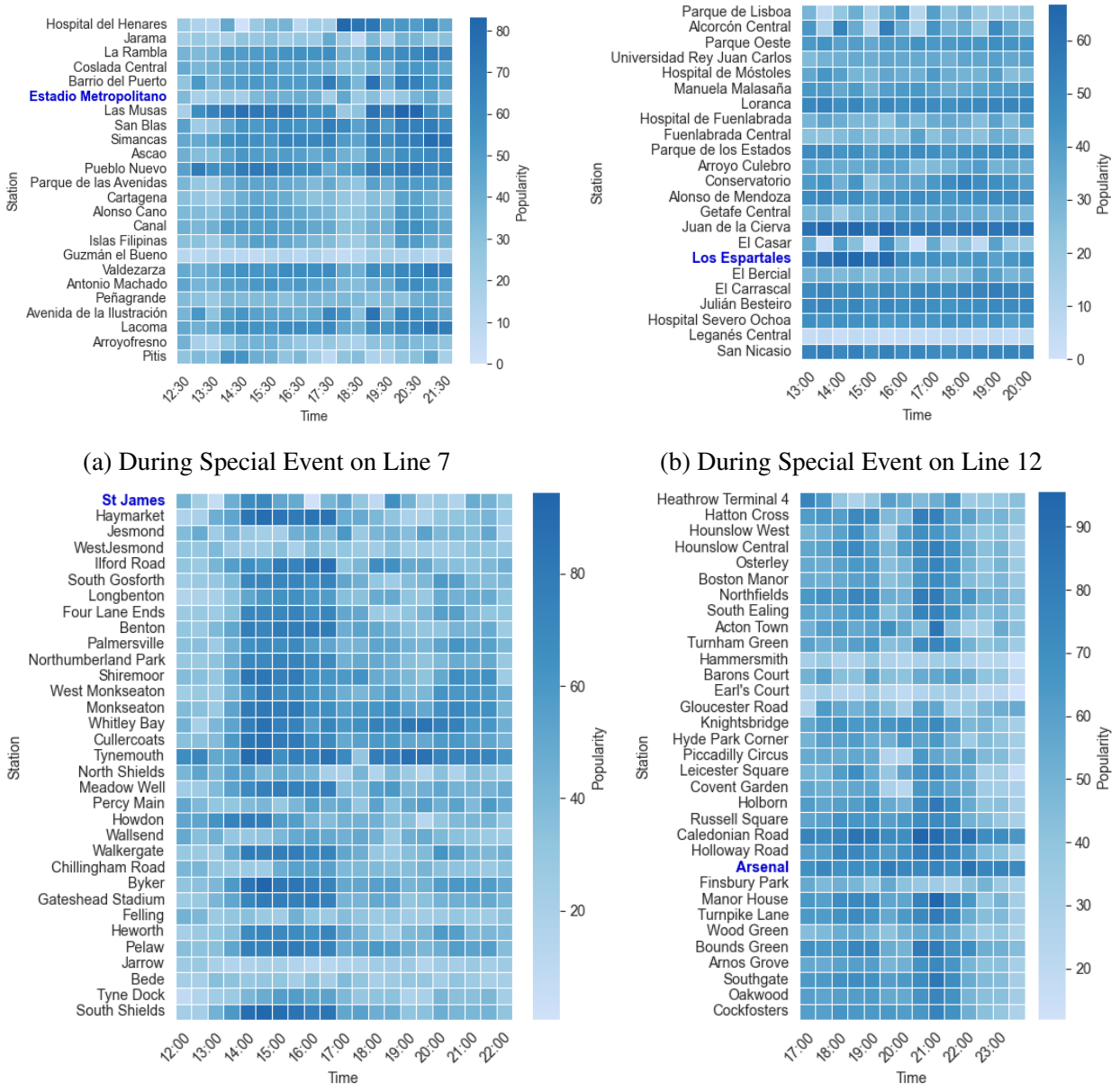
Figure 7(b) represents the match between Getafe and Mallorca on May 26, 2024, on the Madrid Metro Line 12. Los Espartales, the station near the stadium, exhibited crowding during the event. The crowding effect extended to specific upstream and downstream stations on the line.

Figure 7(c) presents data from the Premier League match between Newcastle and Wolves on March 2, 2024. The corresponding line is the Newcastle Metro Yellow Line. According to the Newcastle subway network map (Figure 8), St James, the station near the stadium, is located at one end of this circular line. Other stations on the line showed passenger flow patterns typical of regular times, indicating limited impact from the event.

Figure 7(d) highlights the match between Arsenal and Chelsea on April 23, 2024, on the London Tube Piccadilly Line. The special event significantly influenced popularity at Arsenal and nearby stations, displaying a strong spatiotemporal correlation with the event's timing.

CONCLUSIONS

Our study analyzes the crowdedness patterns at public transport stations during special events across multiple cities and public transport networks by constructing a spatial-temporal graph neural network model. Recognizing that sensor data (e.g., IC card data) is not always available or open-source and digging into the potential of opportunistic data, we have constructed a dataset comprising 428 public transport stations across 17 lines in 10 cities derived from Google Popular Time data. Our model is designed to handle the abnormal passenger flow that special events bring, a significant departure from regular traffic patterns. This unique feature of our model assures its robustness in the face of short-term forecasting challenges. We've also considered the inherent spatial characteristics of the traffic network, integrating the topological structure of rail lines and the adjacency relationships between stations into the temporal dependencies. In our experiments, we initially compared models that considered the event feature with those that did not. We demonstrated that the model incorporating the designated special feature significantly improved performance under special event scenarios without substantially compromising overall accuracy. We showcased the model's predictive performance across all networks and visualized the results, along with the visualization and analysis of popularity change under different scenarios. Our research underscores the importance of including special event data in passenger flow prediction. The APT-GCN model effectively addresses the challenges posed by special events,



(a) During Special Event on Line 7

(b) During Special Event on Line 12

(c) During Special Event on Yellow Line

(d) During Special Event on Piccadilly Line

FIGURE 7: Crowdedness Pattern during Special Events

providing valuable insights for urban rail transit management. Future work can explore additional data sources and refine the model to enhance prediction accuracy and operational efficiency across urban environments.

ACKNOWLEDGEMENTS

This work was supported by the International Graduate School of Science and Engineering (IGSSE) of Technical University of Munich (TUM) (MODA project).

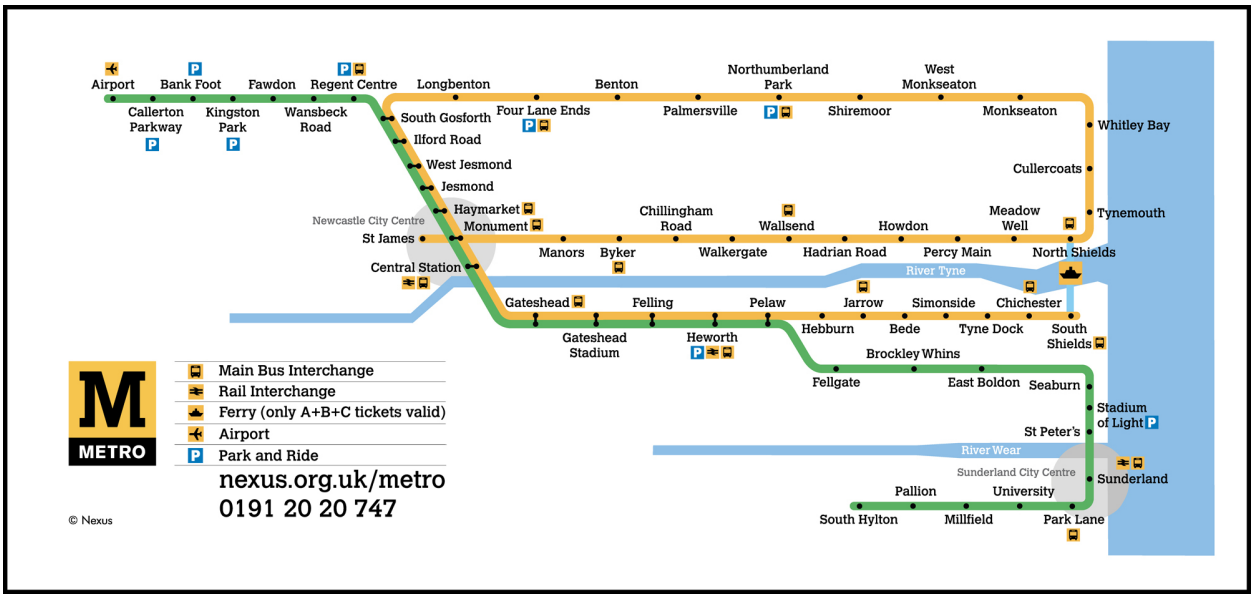


FIGURE 8: Newcastle Public Railway Transport Network Map

REFERENCES

1. Kumar, P. and A. Khani, Evaluating Special Event Transit Demand: A Robust Principal Component Analysis Approach. *IEEE Transactions on Intelligent Transportation Systems*, Vol. 22, No. 12, 2021, pp. 7370–7382.
2. van der Voort, M., M. Dougherty, M. Dougherty, and S. Watson, Combining Kohonen maps with Arima time series models to forecast traffic flow. *Transportation research. Part C: Emerging technologies*, Vol. 4, No. 5, 1996, pp. 307–318.
3. Williams, B. M. and L. A. Hoel, Modeling and Forecasting Vehicular Traffic Flow as a Seasonal ARIMA Process: Theoretical Basis and Empirical Results. *Journal of Transportation Engineering*, Vol. 129, No. 6, 2003, pp. 664–672.
4. Bollerslev, T., Generalized autoregressive conditional heteroskedasticity. *Journal of Econometrics*, Vol. 31, No. 3, 1986, pp. 307–327.
5. Kumar, P. and A. Khani, Evaluating Special Event Transit Demand: A Robust Principal Component Analysis Approach. *IEEE Transactions on Intelligent Transportation Systems*, Vol. 22, No. 12, 2021, pp. 7370–7382.
6. Jin, M., H. Y. Koh, Q. Wen, D. Zambon, C. Alippi, G. I. Webb, I. King, and S. Pan, *A Survey on Graph Neural Networks for Time Series: Forecasting, Classification, Imputation, and Anomaly Detection*, 2023.
7. Yu, H.-F., N. Rao, and I. S. Dhillon, Temporal Regularized Matrix Factorization for High-dimensional Time Series Prediction. In *Advances in Neural Information Processing Systems* (D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, eds.), Curran Associates, Inc., 2016, Vol. 29.
8. Zhao, L., Y. Song, C. Zhang, Y. Liu, P. Wang, T. Lin, M. Deng, and H. Li, T-GCN:

A Temporal Graph Convolutional Network for Traffic Prediction. *IEEE Transactions on Intelligent Transportation Systems*, Vol. 21, No. 9, 2020, pp. 3848–3858.

9. Scarselli, F., M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini, The Graph Neural Network Model. *IEEE Transactions on Neural Networks*, Vol. 20, No. 1, 2009, pp. 61–80.
10. Żochowska, R. and T. Pamuła, Impact of Traffic Flow Rate on the Accuracy of Short-Term Prediction of Origin-Destination Matrix in Urban Transportation Networks. *Remote Sensing*, Vol. 16, No. 7, 2024.
11. Hochreiter, S. and J. Schmidhuber, Long Short-Term Memory. *Neural Computation*, Vol. 9, No. 8, 1997, pp. 1735–1780, _eprint: https://direct.mit.edu/neco/article-pdf/9/8/1735/813796/neco.1997.9.8.1735.pdf.
12. Cho, K., B. van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, *Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation*, 2014.
13. Cho, K., B. van Merrienboer, D. Bahdanau, and Y. Bengio, *On the Properties of Neural Machine Translation: Encoder-Decoder Approaches*, 2014.
14. Chung, J., C. Gulcehre, K. Cho, and Y. Bengio, *Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling*, 2014.
15. Caroleo, B., S. Chiusano, E. Daraio, A. Avignone, E. Gastaldi, M. Paoletti, and M. Arnone, Machine Learning Methods to Forecast Public Transport Demand Based on Smart Card Validations. In *Intelligent Transport Systems* (A. L. Martins, J. C. Ferreira, A. Kocian, U. Tokkozhina, B. I. Helgheim, and S. Bråthen, eds.), Springer Nature Switzerland, Cham, 2024, pp. 194–209.
16. MVV open data, *OpenData*, 2024, snapshot available at: /Users/diana/Zotero/storage/PMDY7QTY/index.html.
17. Ma, L., R. Rabbany, and A. Romero-Soriano, Graph Attention Networks with Positional Embeddings. In *Advances in Knowledge Discovery and Data Mining* (K. Karlapalem, H. Cheng, N. Ramakrishnan, R. K. Agrawal, P. K. Reddy, J. Srivastava, and T. Chakraborty, eds.), Springer International Publishing, Cham, Vol. 12712, 2021, pp. 514–527, series Title: Lecture Notes in Computer Science.